

# Arka plan Modellemesi ve Optik Akış ile Hareket Tespiti

## Araştırma Makalesi/Research Article

 İbrahim DELİBAŞOĞLU

Yazılım Mühendisliği Bölümü, Bilgisayar ve Bilişim Bilimleri Fakültesi, Sakarya Üniversitesi, Sakarya, Türkiye

[ibrahimdelibasoglu@sakarya.edu.tr](mailto:ibrahimdelibasoglu@sakarya.edu.tr)

(Geliş/Received:25.12.2020; Kabul/Accepted:05.05.2021)

DOI: 10.17671/gazibtd.846961

**Özet**— Hareket tespiti çeşitli bilgisayarlı görme problemlerinde ön-işlem aşamasında kullanılmaktadır ve üzerinde çalışılan önemli konulardan birisidir. Hareket tanıma için önerilen yöntemler çoğunlukla arka plan modellemesi ve çıkarımına dayanmaktadır. Bu çalışmada, arka plan modelleme ve optik akış vektörlerinin kullanıldığı bir yöntem önerilmiştir. Optik akış tahmin edilmesinde Farneback yöntemi kullanılmıştır. Önerilen yöntemde optik akış, arka plan çıkarımı aşamasında her bir piksel için uygulanacak eşik değeri belirlemek için kullanılmıştır. Deneysel sonuçlar arka plan çıkarımı yaparken optik akış bilgisini kullanmanın sabit eşik değer uygulamaya nazaran performansı arttırdığını göstermiştir. Önerilen yöntem CDNET-2014 veri kümesinden farklı altküme görüntüleri üzerinde değerlendirilmiş ve literatürdeki yöntemler karşısında iyi sonuçlar elde edilmiştir.  $F_1$  performans kriterinde literatürdeki yöntemlere çok yakın sonuçlar elde edilirken, hatalı tespit edilen veya ıskalanan hareketli piksel oranını veren PWC metriğinde ise en iyi ortalama performansa ulaşıldığı gözlemlenmiştir.

**Anahtar Kelimeler**— hareket tespiti, hareketli nesne tespiti, optik akış

## Motion Detection with Background Modelling and Optical Flow

**Abstract**— Motion detection is a challenging task and may be used as a pre-processing step in different computer vision tasks. Methods proposed for motion detection are mostly based on background modelling and subtraction. In this study, a method is proposed with background modelling and optical flow vectors. Farneback method is applied to estimate optical flow vectors. Optical flow is used to determine the threshold value of each pixel applied in background subtraction step of proposed method. The experimental results show that proposed approach using optical flow in background subtraction improves the performance according to a static threshold. The proposed method is evaluated on different subset images of CDNET-2014 dataset and has a reasonable performance against methods in the literature. Proposed method has similar performance in the  $F_1$  metric compared to the methods in the literature, but it is observed that it has the best average performance in the PWC metric, which gives the ratio of wrongly detected or missed moving pixels.

**Keywords**— motion detection, moving object detection, optical flow

### 1. GİRİŞ (INTRODUCTION)

Hareketli nesnelerin tanınması problemi, gerçek zamanlı nesne takip etme, olay analizi ve güvenlik uygulamaları gibi alanlarda ihtiyaç duyulan bir bilgisayarlı görme konusudur. Son yıllarda gerçekleştirilen bilgisayarlı görme çalışmaları baz alındığında üzerinde yoğun bir şekilde çalışılan bir problem olduğu göze çarpmaktadır [1]. Örneğin; güvenlik amaçlı gerçekleştirilen bir uygulamada

sabit bir kameradan alınan görüntüler işlenerek hareket tespiti yapılmış ve tasarlanan kamera destekli güvenlik sistemi ile maliyet, zaman ve emek kaybının önlenmesi amaçlanmıştır [2]. Hareketli nesne tespitinde amaç, görüntünün ön plan ve arka plan olarak sınıflandırılmasıdır. Yapılacak sınıflandırma görüntü alınan kameranın hareket durumu, ortam ışıklandırması, arka plana ait dinamik değişimler gibi etkenlere göre karmaşık olmaktadır. Görüntünün alındığı kameralar statik

veya hareketli olabilmektedir. Hareketli kameralar belirli periyotlarla hareket eden veya serbest bir şekilde hareket eden kameralar olarak sınıflandırılabilir. Bunlara örnek olarak; bir gölgeyi çekim açısını değiştirerek tarayan güvenlik kameraları (PTZ- pan-tilt-zoom) ve drone kameraları gösterilebilir.

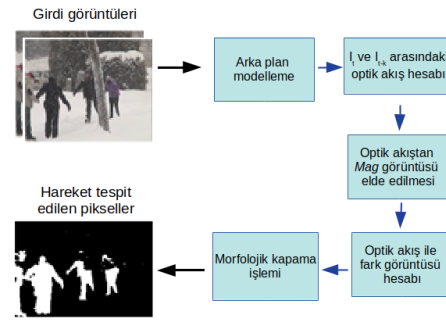
Hareketli nesne tanıma uygulamalarında amaç doğruluğunun yüksek olmasının yanı sıra uygulamanın gerçek zamanlı çalışabilmesidir. Literatürde gerçekleştirilen çalışmalar incelendiğinde, art arda gelen imgelerin çıkarılması, arka plan modellemesi ve optik akış temelli yöntemler kullanıldığı görülmektedir. Art arda gelen imgelerin çıkarılması yöntemi her ne kadar hızlı çalışsa ve arka plan değişimlerine hızlı uyum sağlayabilse de başarı oranı çok düşüktür [3,4]. Optik akış yöntemlerinde ise klasik yoğun optik akış bulan yöntemler ile derin öğrenme temelli yöntemler kullanılmaktadır [5]. FlowNet2 mimarisi bu kapsamda yaygın kullanılan mimarilerden birisidir [6]. Derin öğrenme temelli optik akış tahmini yöntemlerinin dezavantajı yüksek çözünürlüklü görüntülerde hesaplama maliyetinin yüksek olması, dolayısıyla gerçek zamanlı uygulamalar için elverişli olmamasıdır. Dolayısıyla, hesaplama maliyeti-performans dengesi açısından en verimli yöntemin arka plan modellemesi temelli yöntemler olduğu söylenebilir. Bu yöntemlerde belirli geçmişteki görüntüler kullanılarak arka plana ait bir model çıkarılır [7]. Arka plan modeli oluşturulmasında, klasik görüntü işleme teknikleri [8], istatistiksel yöntemler [9-11] ve sinir ağları [12] literatürde gerçekleştirilen çalışmalarda kullanılmıştır. Kwang ve arkadaşları tarafından gerçekleştirilen çalışmada [9], her bir pikselin yaş, standart sapma ve ortalama değerlerinin tutulduğu iki ayrı arka plan modeli oluşturulmuştur. Kamera hareketinden kaynaklanan yanlış sınıflamaları önlemek için, tanımlanan bir komşuluk penceresi parametresine göre her bir pikselin komşuları ile karşılaştırıldığı bir yöntem uygulanmıştır. Bu yöntemde seçilen komşuluk değeri yöntemin çalışma hızını ve doğruluğunu etkilemektedir. Önerilen bu yöntemin GMM [10] yönteminden en önemli farkı, hareketli pikselleri arka plan ile karıştırma noktasında daha başarılı olmasıdır. Literatürde hareketli kameralar için iyi sonuçlar veren bir başka yöntem ise ViBe [13] diye isimlendirilen yöntemdir. Bu yöntem ile GMM yöntemine nazaran daha iyi sonuçlar elde edilebildiği farklı veri kümeleri üzerinde gösterilmiştir. Son zamanlarda gerçekleştirilen farklı çalışmalarda [14, 15] ise yine birden çok arka plan modeli oluşturan ve kamera hareketinin etkisini minimuma indirmek için piksel komşulukları ile kontrol işlemlerinin uygulandığı yöntemler önerilmiştir. Ayrıca hareketli pikseller tespit edildikten sonra uygulanan bölütleme metotlarıyla sonuçlar iyileştirilmektedir. Bu yöntemlerde uygulanan komşuluk kontrolü işlemleri işlem gücü gereksinimini özellikle yüksek çözünürlüklü görüntüler için önemli miktarda arttırmakta ve ancak GPU desteğiyle çalıştırılarak gerçek zamana yakın hızlarda çalışabilmektedir.

Bu çalışmada arka plan modellemesi ve optik akış vektörlerinin bir arada kullanıldığı bir yöntem önerilmiştir. Uygulama C++, CUDA ve OpenCV kütüphaneleri

kullanılarak geliştirilmiştir. Önerilen yöntem, literatürde kabul gören GMM [11] ve ViBe [13] yöntemleri ile kıyaslanmıştır. Önerilen yöntemin detayları ve performans kıyaslamaları bölüm 2 ve 3’de paylaşılmıştır. Son bölümde ise sonuçlar yorumlanmış, eksik yönler ve bir sonraki çalışma kapsamında yapılabilecek iyileştirmeler üzerinde durulmuştur.

## 2. YÖNTEM (METHOD)

Önerilen yöntemde arka plan modeli oluşturulduktan sonra,  $t$  anındaki görüntünün  $t-k$  anındaki bir görüntü ile arasındaki optik akış vektörleri hesaplanmıştır. Bu optik akış vektörleri, arka plan görüntüsünden fark görüntüsü hesaplanması aşamasında kullanılmıştır. Önerilen yöntemde uygulanan aşamalar Şekil 1’de görsel olarak gösterilmiş, detayları 2.1 ve 2.2 alt başlıkları altında açıklanmıştır.



Şekil 1. Önerilen yöntemin akış şeması (Flow chart of the proposed method)

### 2.1. Arka plan Modellemesi (Background Modelling)

Statik kameralarda ışık veya arka plan hareketi durumlarına, hareketli kameralar için ise sürekli ve daha hızlı değişen arka plana uyum sağlamak için arka plan modellemesi ihtiyacı ortaya çıkmaktadır. Arka plan modellenmesinde amaç, art arda gelen video görüntülerinden statik arka planı çıkarmaktır. Arka plan eğer başarılı olarak çıkarılabilirse, elde edilen anlık görüntü arka plan modelinden çıkarılarak değişim olan yerler (hareketli bölgeler) tespit edilebilir. Bunun için görüntüdeki her bir pikselin geçmiş bilgisi tutularak, ortalama ve standart sapmaları hesaplanır. GMM yönteminde her bir piksel  $K$  adet Gauss fonksiyonunun karışımı ile ifade edilir. Zivkovic tarafından önerilen yöntemde, her bir piksel için karar aşamasında kullanılan GMM parametrelerinin güncellenmesi ve öğrenme hızının adaptif olarak değiştirilmesi amaçlanmıştır [10,11]. Böylece arka plan değişimlerine daha hızlı bir şekilde uyum sağlanabilmektedir.

Bu çalışma kapsamında tek bir arka plan modeli oluşturulmuş, her bir pikselin ortalaması geçmiş değerlerine göre hesaplanmıştır. Arka plan hesaplanması için denklem (1)’de gösterilen formül kullanılmaktadır. Denklemde  $p_i$   $I$  görüntüsündeki bir pikseli,  $M_i(p_i)$  ilgili

pikselin  $t$  anındaki ortalamasını,  $\alpha$  öğrenme hızını ifade etmektedir.

$$\alpha = \frac{1}{age} \quad (1)$$

$$M_t(p_i) = (1 - \alpha(p_i))M_{t-1}(p_i) + \alpha(p_i) I_t(p_i) \quad (2)$$

Öğrenme hızı belirlenirken her bir pikselin yaşı ( $age$ ) bilgisi göz önüne alınmıştır. Buna göre; arka plan değerleri hesaplanırken 2. görüntüdeki bir pikselin etkisi 0,5 iken, 3. görüntüde bu oran 0,33 4. görüntüde ise 0,25 olmaktadır ve giderek azalmaktadır.  $age$  için maksimum değer 500 olarak seçilmiştir, dolayısıyla  $t=500$  anından itibaren  $\alpha=0,002$  olmaktadır. Veri kümesinden örnek bir görüntü ve geçmiş görüntülerin ortalamasından elde edilen arka plan görüntüsü Şekil 2'de gösterilmiştir. Arka plan modellemesi neticesinde öğrenme hızına bağlı olarak hareketli bölgelerin elimine edildiği statik arka plan elde edilmektedir. Piksel değerleri öğrenme hızına göre güncellenirken, hareketli nesne durağan duruma geçtiğinde belirli bir süre daha hareketli olarak algılanabilir. Veya büyük bir nesne hareket ederken çok yüksek bir öğrenme hızı seçildiğinde nesnenin iç kısımları hatalı bir şekilde arka plan olarak değerlendirilebilir. Arka plan elde edildikten sonra hareketli piksellerin bulunması için arka plan çıkarımı işlemi uygulanmaktadır.



Şekil 2. Örnek bir görüntü (a) ve arka plan görüntüsü (b)  
(Sample frame (a) and background image (b))

## 2.2. Optik Akış ve Arka plan Çıkarımı (Optical Flow and Background Subtraction)

Arka plan elde edildikten sonra, her bir görüntü arka plan görüntüsünden çıkarılarak fark görüntüsü hesaplanır. Fark görüntüsü üzerinden eşik değer alma işlemi uygulanarak,  $T$  değerinden fazla değişim olan pikseller hareketli piksel olarak değerlendirilir. Gerçekleştirilen çalışmada ilk olarak tüm pikseller için aynı ( $T=50$ ) eşik değeri kullanılmıştır. Literatürdeki çalışmalarda özellikle hareketli kameralarda her bir piksel için farklı eşik değerler ve farklı öğrenme hızları kullanıldığı görülmektedir. Bu çalışma kapsamında sabit kameralar için önerilen yöntemde, literatürdeki çalışmalardan farklı olarak her bir piksel için öğrenme hızı belirlenirken optik akış vektörlerinden elde edilen piksel değişim oranları da hesaba katılmıştır. Optik akış hesaplamasında, OpenCV kütüphanesinde CUDA implementasyonu bulunan Farneback yöntemi kullanılmıştır [16]. Optik akış hesaplanırken iki girdi görüntüsü gerekmektedir. Şekil 3'te veri kümesinden örnek bir görüntü ve sağda  $t$  ile  $t-k$  anlarındaki her iki

görüntü kullanılarak elde edilmiş  $Mag$  görüntüsü gösterilmiştir.



Şekil 3. Örnek görüntü ve optik akış  
(Sample image and optical flow)

Çalışma kapsamında optik akış hesaplanırken  $k=8$  değeri kullanılmıştır. Optik akış ile her bir pikselin  $x$  ve  $y$  yönlerinde ne kadar değişim gösterdiği bilgisi elde edilmektedir. Önerilen yöntemde optik akış vektörlerinden denklem (3)'de gösterilen formülle  $magnitude$  ( $Mag$ ) görüntüsü hesaplanmaktadır.

$$Mag = \sqrt{dxdx + dydy} \quad (3)$$

Denklemde  $dx$  her bir pikselin  $x$  yönündeki değişimi,  $dy$  ise  $y$  yönündeki değişimini göstermektedir.  $Mag$  görüntüsü elde edildikten sonra, belirli bir eşik değerden ( $T_{mag} = 15$ ) büyük olan yerlerde hareket olduğu kabul edilmiştir.  $Mag$  görüntüsünde  $T_{mag}$  değerinden büyük değişim gösteren pikseller 1-2 değerleri arasına normalize edilerek,  $Mag_n$  matrisi hesaplanır (4). Ardından fark görüntüsü  $D$  hesaplanarak (5),  $Mag_n$  matrisi ile çarpılır (6). Böylece optik akış neticesinde fazla değişim olduğu tahmin edilen piksellerin, fark görüntüsündeki değişim miktarlarının artırılması amaçlanmaktadır.

$$Mag_n(p_i) = \text{norm}(Mag(p_i)) \text{ if } Mag(p_i) > T_{mag} \quad (4)$$

$$D_t(p_i) = I_t(p_i) - M_t(p_i) \quad (5)$$

$$C_t(p_i) = Mag_n(p_i) \times M_t(p_i) \quad (6)$$

Elde edilen  $C$  görüntüsü, fark görüntüsü ( $D$ ) üzerine optik akış bilgisi ile hareket olduğu tahmin edilen piksellerin etkisinin artırıldığı bir fark görüntüsüdür.  $C$  görüntüsüne eşik değer alma işlemi uygulanması suretiyle hareketli pikseller elde edilmektedir (7).

$$F_t(p_i) = \begin{cases} 255, & C_t(p_i) \geq T \\ 0, & C_t(p_i) < T \end{cases} \quad (7)$$

Hareketli piksel maskesi ( $F$ ) bulunduktan sonra son olarak, morfolojik kapama işlemi uygulanarak hareketli olarak saptanan bölgelerin içerisindeki boşluklar doldurulmaya çalışılmıştır. Önerilen yöntemin her bir aşamasında uygulanan işlemler Tablo 1'de paylaşılmıştır.

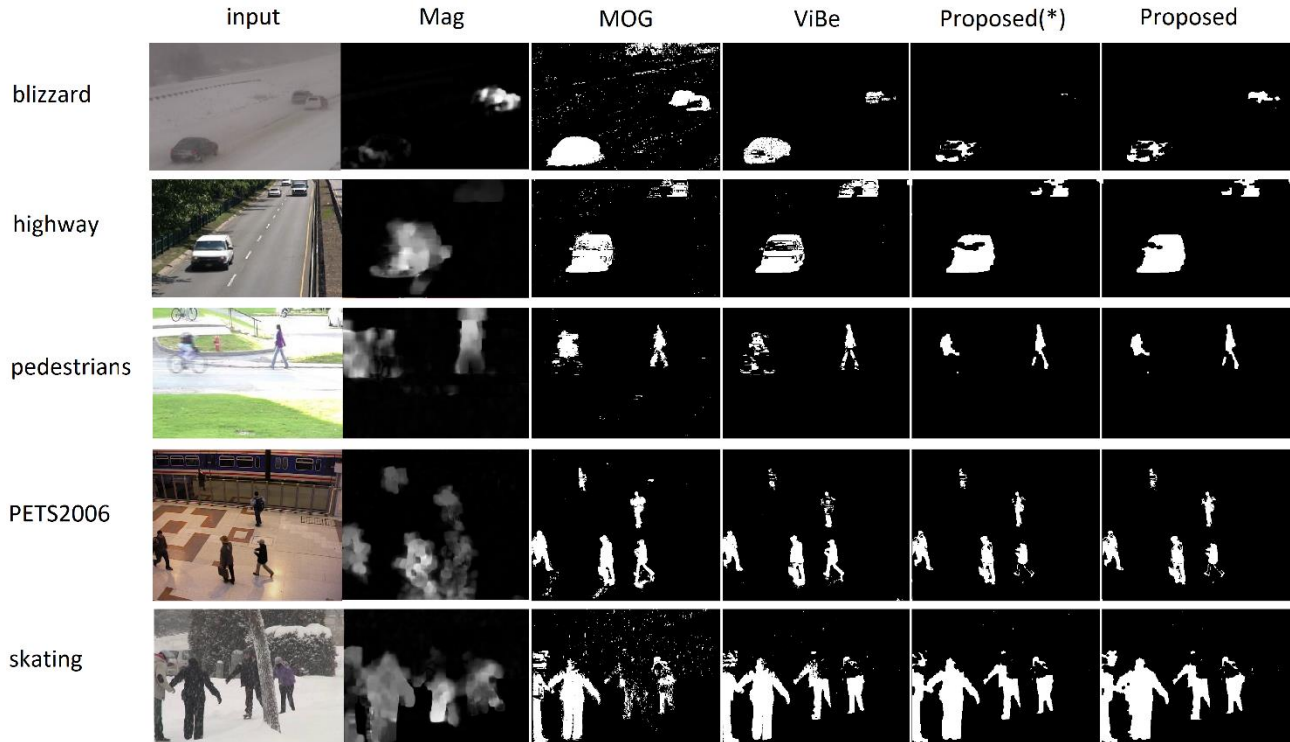
Tablo 1. Algoritma adımları  
(Algorithm steps)

Adım-1	Arka plan oluşturulması: Denklem (2)
Adım-2	$I_t$ ve $I_{t-k}$ ile optik akış tahmini
Adım-3	Mag görüntüsünün elde edilmesi: Denklem (3)
Adım-4	Fark görüntüsünün hesaplanması: Denklem (4,5,6)
Adım-5	Eşik değer alma: Denklem (7) ve morfolojik kapama işlemi

### 3. VERİ SETİ VE DENEYSEL SONUÇLAR (DATASET AND EXPERIMENTAL RESULTS)

Çalışmada hareket tespiti algoritmalarının kıyaslanmasında yaygın olarak kullanılan *Change Detection* (CDNET-2014) veri seti kullanılmıştır [17].

CDNET-2014 veri kümesinden *baseline* ve *badWeather* alt kümelerinden örnek görüntüler kullanılarak performans kıyaslaması gerçekleştirilmiştir. Seçilen görüntülerde hareket olarak değerlendirilmeyen gölge alanlar, ağaç hareketi ve kar yağışı gibi zorlayıcı tipte görüntüler yer almaktadır. Önerilen yöntemde, *baseline* alt kümesindeki görüntülerde yer alan gölgeleri elimine etmek için Sanin ve arkadaşları tarafından gerçekleştirilen çalışmada [18] tanımlanmış *Chromacity* ve *LR-texture* yöntemleri kullanılmıştır. Yapılan denemeler sonucunda, kapalı alan görüntüleri içeren *baseline/PETS2006* verisi için *chromacity* yöntemi, diğer görüntülerde ise *LR-texture* yöntemi en iyi sonucu verdiği için ilgili yöntemler gölge alanları elimine etmek için bu yöntemler kullanılmıştır.



Şekil 4. Veri kümesindeki görüntüler ile farklı yöntemler kullanılarak elde edilen hareketli alanlar  
(Motion masks obtained from different methods by using images in dataset)

#### 3.1. Performans Metrikleri (Performance Metrics)

Önerilen yöntem literatürdeki MOG [11] ve ViBe [13] yöntemleri ile kıyaslanmıştır. Algoritma performansları denklem (8) ve (9) ile gösterilen  $P$ ,  $R$ ,  $F_1$  ve  $PWC$  metrikleri ile değerlendirilmiştir.

$$P = \frac{TP}{TP+FP} \quad R = \frac{TP}{TP+FN} \quad F_1 = \frac{2PR}{P+R} \quad (8)$$

$$PWC = \frac{100(FP+FN)}{TP+FP+TN+FN} \quad (9)$$

$P$  bulunan hareketli piksellerin ne kadarının doğru olduğunu (kesinlik),  $R$  ise hareketli piksellerin ne kadarının tespit edilebildiğini (duyarlılık) göstermektedir.  $F_1$  değeri,  $P$  ve  $R$  değerlerinin harmonik ortalamasıdır.  $PWC$  değeri

ise, yanlış hesaplanan hareketli veya hareketli olmayan piksel sayılarının yüzdesini ifade etmektedir.

#### 3.2. Deneysel Sonuçlar (Experimental Results)

Sayısal sonuçlar Tablo 2 ve Tablo 3'te, görsel sonuçlar ise Şekil 4'te paylaşılmıştır. Tabloda her bir görüntü kümesi için en iyi iki performans metriği değerleri kalın rakamlarla gösterilmiştir. *Proposed(\*)* ile optik akış kullanılmadan direkt sabit eşik değer alma ile elde edilen sonuçlar ifade edilmiştir. *highway* görüntülerinde en iyi  $F_1$  skor değeri önerilen yöntemle elde edilmiş, diğer görüntülerde ViBe ve GMM yöntemlerinin en iyi  $F_1$  skor değerlerine sahip olduğu gözlemlenmiştir. Tablo 1'de *baseline* alt kümesinin ortalama  $F_1$  değerleri incelendiğinde GMM ve ViBe yöntemleriyle 0,7861 ve 0,7954 doğruluk değerlerine ulaşılmıştır. Önerilen yöntemde, optik akış vektörleri kullanılmadığında ortalama 0,7893  $F_1$  değerine ulaşılırken,



optik akış bilgisi ile birlikte 0,8018  $F_1$  değeri ile en yüksek ortalama  $F_1$  değerine ulaşmıştır.

Tablo 2 ve Tablo 3'teki sonuçlar incelendiğinde, önerilen yöntemde arka plan çıkarımı esnasında optik akış vektörlerinin kullanılmasıyla birlikte tüm görüntülerde performansın arttığı gözlemlenmiştir. Görüntü kümeleri ayrı ayrı incelendiğinde ise, kalın rakamlarla ifade edilen en iyi metrik değerleri göz önüne alındığında önerilen yöntemle en iyi sonuçların veya en iyi sonuca yakın performans değerlerinin elde edildiği görülmüştür. Önerilen yöntemle, toplam hata oranını gösteren PWC metriğinde çoğunlukla daha düşük değerler elde edildiği görülmüştür.

Tablo 2. *baseline* alt kümesi için performans kıyaslaması  
(Performance comparison for *baseline* subset)

Yöntemler	highway			
	PWC	P	R	$F_1$
GMM	2,6117	0,8239	0,7111	0,7634
ViBe	<b>2,2618</b>	<b>0,900</b>	0,6955	<b>0,7846</b>
Proposed* (without Mag)	<b>2,365</b>	0,8824	0,6932	0,7764
Proposed	0,2368	<b>0,8847</b>	<b>0,7157</b>	<b>0,7931</b>
PETS2006				
GMM	<b>0,4391</b>	0,9318	<b>0,7143</b>	<b>0,8087</b>
ViBe	0,6326	0,9039	0,5741	0,7072
Proposed* (without Mag)	0,5700	<b>0,9529</b>	0,5904	0,7291
Proposed	<b>0,5433</b>	<b>0,9531</b>	<b>0,6119</b>	<b>0,7453</b>
pedestrians				
GMM	0,4689	0,7122	<b>0,8774</b>	0,7862
ViBe	<b>0,1842</b>	0,9678	<b>0,8404</b>	<b>0,8996</b>
Proposed* (without Mag)	0,2406	<b>0,9836</b>	0,7680	0,8625
Proposed	<b>0,2343</b>	<b>0,9799</b>	0,7775	<b>0,8670</b>

Tablo 3. *badWeather* alt kümesi için performans kıyaslaması  
(Performance comparison for *badWeather* subset)

Yöntemler	skating			
	PWC	P	R	$F_1$
GMM	2,5581	0,7376	0,7519	0,7411
ViBe	<b>1,019</b>	<b>0,9866</b>	0,8049	0,8865
Proposed* (without Mag)	<b>0,9728</b>	<b>0,9883</b>	<b>0,8129</b>	<b>0,8921</b>
Proposed	1,0268	0,9436	<b>0,8428</b>	<b>0,8903</b>
blizzard				
GMM	<b>0,5566</b>	0,7598	<b>0,7641</b>	<b>0,7619</b>
ViBe	0,7486	0,9935	0,3602	0,5287
Proposed* (without Mag)	0,9482	<b>0,9973</b>	0,1871	0,3152
Proposed	<b>0,7452</b>	<b>0,9938</b>	0,3630	<b>0,5318</b>

Tablo 3'te ise kötü hava şartları görüntüleri ile elde edilen metrikler gösterilmiştir. Önerilen yöntem *skating* görüntülerinde en iyi performansı sergilerken, *blizzard* görüntülerinde GMM'nin gerisinde kalmıştır. Şekil 4'teki görsel sonuçlara bakıldığında, kar yağışından dolayı özellikle beyaz araçların geçişleri esnasında  $T=50$  eşik değerinin yüksek olduğu gözlemlenmiştir. Bu sebeple,

optik akış dikkate alınmadığında çok daha kötü performans elde edilirken, optik akışla birlikte ciddi manada bir performans artışı sağlandığı görülmüştür. *blizzard* görüntüsünde optik akışın etkisiyle sağ üstteki araçların bulunabildiği Şekil 4'te görülmektedir. Eşik değerinin yüksek seçilmesinin sebebi, tüm görüntülerde aynı eşik değeri kullanabilmek ve gürültüyü düşürmektir. Eşik değeri düşük seçildiğinde, küçük arka plan hareketleri bile hareket olarak algılanabilmektedir. Örneğin *skating* görüntü sonuçlarına bakıldığında, GMM ile son derece gürültülü sonuçlar elde edilirken önerilen yöntemde gürültü minimum düzeydedir.

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

Gerçekleştirilen çalışmada tek bir arka plan modeli oluşturularak, optik akış bilgisiyle birlikte arka plan çıkarımı işlemi neticesinde hareketli pikseller tanınmıştır. Tek model kullanılması ve sabit öğrenme hızından dolayı yöntemde aslında değişen arka plana uyum sağlama konusunda yetersiz kalmaktadır. Bu sebeple, arka plan çıkarımı neticesinde hareketli pikseller bulunurken kullanılan eşik değeri yüksek tutulmuş ve yanlış ön plan bulunmasının önüne geçilmeye çalışılmıştır. Eşik değerinin yüksek olması, hareketli piksellerin birçoğunun yakalanamamasına sebep olmaktadır. Önerilen yöntemde, optik akış vektörlerinin kullanılması ile fark görüntüsünde hareket olduğu tahmin edilen bölgelerin piksel değerleri iki katına kadar artırılmıştır. Böylece, optik akış yüksek olan pikseller için bir nevi adaptif eşik değeri kullanılmıştır.

Bununla birlikte, optik akış vektörlerini hesaplamada kullanılan Farneback yönteminin birçok durumda hareketli bölgeleri olduğundan çok daha fazla bulduğu veya nispeten küçük hareketleri ıskaladığı görülmüştür. Bu kapsamda gerçekleştirilecek sonraki çalışmalarda optik akış için farklı yöntemlerin denemesi de planlanmaktadır. Ayrıca öğrenme hızının da adaptif bir şekilde belirlenmesi ile daha fazla değişen arka planlara karşı daha etkili olabileceği düşünülebilir. Üzerinde iyileştirme yapılacak bir başka problem ise Şekil 4'te ilk iki satırdaki görüntülerde açık şekilde görülen, bulunan hareketli bölgelerin içerisinde boşluklar olması sorunudur. Hareket tespit edilen bölgenin üzerinde sonradan gerçekleştirilecek ekstra bölütleme işlemleriyle kopuk kopuk tespit edilen hareketli pikseller birleştirilebilir. Yürütülen çalışmanın bu kapsamda genişletilerek geliştirilmesi düşünülmektedir.

#### KAYNAKLAR (REFERENCES)

- [1] C. Marie-Neige, T. Bouwmans. "Moving Objects Detection with a Moving Camera: A Comprehensive Review", *arXiv preprint arXiv:2001.05238*, 2020.
- [2] H. Çakır, H. K. Babacan, "Hareketi Algılayan Kamera Destekli Güvenlik", *Bilişim Teknolojileri Dergisi*, 4(2), 2011.
- [3] R. Collins, A. J. Lipton, T. Kanade, H. Fujiyoshi, D. Duggins, Y. Tsin, D. Tolliver, N. Enomoto, O. Hasegawa, P. Burt, et al., **A system for video surveillance and monitoring**, VSAM final report, 1–68, 2000.

- [4] L. Zhao, Q. Tong, H. Wang., "Study on moving-object-detection arithmetic based on w4 theory", **2011 2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)**, IEEE, 4387–4390, 2011.
- [5] H. Junjie, et al., "Optical flow based real-time moving object detection in unconstrained scenes", *arXiv preprint arXiv:1807.04890*, 2018.
- [6] I. Eddy, et al., "FlowNet 2.0: Evolution of optical flow estimation with deep networks.", **IEEE conference on computer vision and pattern recognition**, 2462-2470, 2017.
- [7] T. Bouwmans, "Traditional Approaches in Background Modeling for Video Surveillance.", **Handbook Background Modeling and Foreground Detection for Video Surveillance**, Taylor and Francis Group, T. Bouwmans, B. Hoferlin, F. Porikli, A. Vacavant, 2014.
- [8] A. Gianni, et al., "EFIC: edge based foreground background segmentation and interior classification for dynamic camera viewpoints", **International Conference on Advanced Concepts for Intelligent Vision Systems**. Springer, Cham, 2015.
- [9] M. Yi, Kwang, et al., "Detection of moving objects with non-stationary cameras in 5.8 ms: Bringing motion detection to your mobile device", **IEEE Conference on Computer Vision and Pattern Recognition Workshops**. 2013.
- [10] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction.", **17th International Conference on Pattern Recognition**, IEEE, 2004.
- [11] Z. Zivkovic, F. Van Der Heijden, "Efficient adaptive density estimation per image pixel for the task of background subtraction", *Pattern recognition letters*, 27(7), 773-780, 2006.
- [11] M. De Gregorio, M. Giordano, "WiSARDrp for Change Detection in Video Sequences", **European Symposium on Artificial Neural Network (ESANN)**, 453-458, 2017.
- [13] B. Olivier, M. Van Droogenbroeck, "ViBe: A universal background subtraction algorithm for video sequences", **IEEE Transactions on Image processing**, 20(6), 1709-1724, 2010.
- [14] Y. Kimin, J. Lim, J. Y. Choi, "Scene conditional background update for moving object detection in a moving camera", *Pattern Recognition Letters*, 88, 57-63.
- [15] Y. Yang, L. Kurnianggoro, K. Jo, "Moving object detection for a moving camera based on global motion compensation and adaptive background model", *International Journal of Control, Automation and Systems*, 17(7), 1866-1874, 2019.
- [16] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion", **Scandinavian conference on Image analysis**, Springer, Berlin, Heidelberg, 363-370, 2003.
- [17] Internet: <http://changedetection.net/>, 25.12.2020.
- [18] A. Sanin, C. Sanderson, B. C. Lovell, "Shadow detection: A survey and comparative evaluation of recent methods", *Pattern recognition*, 45(4), 1684-1695, 2012.