

# Parçacık Filtresi Tabanlı Dikkat Odaklamalı Görsel Hedef Takibi

İlter KILIÇASLAN

Orta Doğu Teknik Üniversitesi  
Elektrik-Elektronik Mühendisliği Bölümü  
ikilicaslan@aseelsan.com.tr

Aydan M. ERKMEN

Orta Doğu Teknik Üniversitesi  
Elektrik-Elektronik Mühendisliği Bölümü  
aydan@metu.edu.tr

## Özet

Bu bildiride, Humanoidlerde kullanılabilecek şekilde, bir objenin, Parçacık Filtresi kullanılarak, hareketli odaklanma penceresi içerisinde takip edilmesine ilişkin bir yöntem sunulmuştur. Takip sırasında birincil hedef özellik olarak hareketli objenin rengi temel alınmıştır. Hareketli odaklanmaya ait boyutların, maksimum odak pencere hızının ve hedef rengin parametrik birer girdi olarak tanımlandığı bu çalışma sonucunda, odaklanmanın hareketli objeyi çevrimiçi ve gerçek zaman olarak başarılı bir biçimde takip edilebildiği görülmüştür. Çalışmamızın amacı, belirtilen görsel dikkat odaklanmasını ses duyuları ile birleştirerek Humanoid kafasına insansı hareket kabiliyeti sağlamaktır. Bildirimimiz dâhilinde, görsel dikkat odaklanması alt sisteminde kullanılan yöntemimiz, örnekler üzerinden performans analizleri ile sunulmaktadır.

**Anahtar Kelimeler:** Parçacık Filtresi, Hareketli Hedef Takibi, Görsel Hedef Takibi, Dikkat Odaklaması.

## 1. Giriş

Görsel hedef takibi, verilen görüntü içerisinde hedefin yerinin bulunmasına ilişkin yöntemdir. Hedef takibinin başlıca uygulama alanları arasında güvenlik & gözetleme [1], robot yeri tespiti [2], radar ile hedef tespiti-takibi [3], tıbbi görüntüleme [4] ve görsel iletişim [5] yer almaktadır. İlgili uygulama alanlarında farklı yöntemlerle uygulanan görsel hedef takibi, özellikle hayati önem taşıyan uygulamalarda gün geçtikçe daha kritik bir rol almaktadır.

Görsel hedef takibinin önemli bir rolü olan diğer bir uygulama alanı da “Humanoid”lerdeki dikkat odaklanmasıdır. Görsel yeteneğe sahip Humanoidlerde dikkat algısının oluşturulabilmesi için dikkati üzerine çekmesi istenen insan veya nesnenin uygun şekilde takip edilebilmesi gerekmektedir. Humanoidlerde dikkat odaklanmasının sağlanabilmesi adına, insan beyninin

dikkat olgusuyla arasındaki ilişkiyi modellemek gibi biyolojik tabanlı çalışmalar yapılmıştır [6],[7].

Bayes Filtresi kavramını temel alan ancak kesin sonucun hesaplanmasının maliyet-etkin olmadığı durumlarda, yaklaşık sonucun hesaplanması anlayışına dayanan Parçacık Filtresi, simülasyona dayalı bir model tahmin tekniğidir [8].

Görsel hedef takibi konusu kapsamında Parçacık Filtresi, birden fazla hedefin tespit edilmesi, birden fazla hedefin birbirinden ayrılması, çoklu hedeflerin takibi, eşzamanlı konum belirleme ve harita oluşturma (SLAM) gibi birçok alt başlıkta kullanılmaktadır [2],[9],[10].

Ayrıca Humanoid hareketlerinin veya insan niyetlerinin tahmin edilmesine yönelik olarak Parçacık Filtresi yönteminin kullanılabileceğini destekleyen çalışmalar da literatürde yer almaktadır [11],[12].

Çalışmamızın amacı, Humanoidlerdeki dikkat odaklanmasını ses duyuları ile birleştirerek robot kafasına insansı hareket kabiliyeti sağlamaktır. Çalışma bünyesinde, hâlihazırda var olan çalışmalardan farklı olarak; takibi öngörülen objeye dikkatin odaklanması görevinin görsel girdi olmadığı durumlarda ses duyularıyla, görsel girdinin olduğu durumlarda ise ses duyuları ve görsel farkındalığın bir arada kullanılarak sağlanması hedeflenmektedir. Böylece Humanoid takip performansının ve güvenilirliğinin artırılması hedeflenmektedir.

Bildirimimizde ise yukarıda genel yapısı belirtilen sistemimizin çevrimiçi ve gerçek zamanlı görsel dikkat odaklanması ele alınmaktadır. Çalışmamız dâhilinde odaklanma yetenekli görsel hedef takibi alt sistemimiz, Parçacık Filtresi tabanlı gerçekleştirilmiştir. İlgili Parçacık Filtresi hareketli objenin rengini hedef alacak şekilde tanımlanmıştır. Literatürde, bakış kontrolü [14] ve dikkat çekici objelerin tespiti konularında [15] yapılan çalışmalar, obje takibi sırasında Humanoid kafasının uygun şekilde hareket ettirilebilmesi ve objenin takip edilebilmesi için

fiziksel sınırlamalardan doğan bir zaman problemi olduğu vurgulanmaktadır. Çalışmamız dahilinde yapılandırılan Parçacık Filtresi tabanlı uygulamamıza ait kabiliyet ile hareketli objenin takibine ilişkin gerçek zamanlı takip bilgisinin oluşturulabilmektedir. Böylece kontrol öngörülen Humanoid kafasının hareketi için gereken süre artırılabilir.

Oluşturulan tasarım, odaklanma performansının ölçülebilmesi amacıyla farklı analizlere imkân tanıyacak şekilde tasarlanmış olup; hedef alınan renk, pencerenin başlangıç konumu, pencerenin eni ve boyu, pencerenin maksimum hızı, Parçacık Filtresi'ndeki parçacık sayısı, parçacıklara ait hız ve gürültü seviyeleri gibi birçok girdi parametrik olarak tanımlanmıştır.

Kullanılan yöntem, bildirinin 2. bölümünde kapsamlı olarak anlatılmaktadır. Bu bölümün alt başlıkları, oluşturulan yazılımın temel bloklarını, Parçacık Filtresi'ne dayalı yaklaşımımız ile algoritmamızı ve pencere hareketine ilişkin fiziksel modellememizi sunmaktadır. Bildirinin 3. Bölümü performans testlerini kapsamlı bir biçimde irdelemektedir. Bu bölüm ayrıca sonuç gözlemlerimizi aktarmaktadır.

## 2. Yöntem

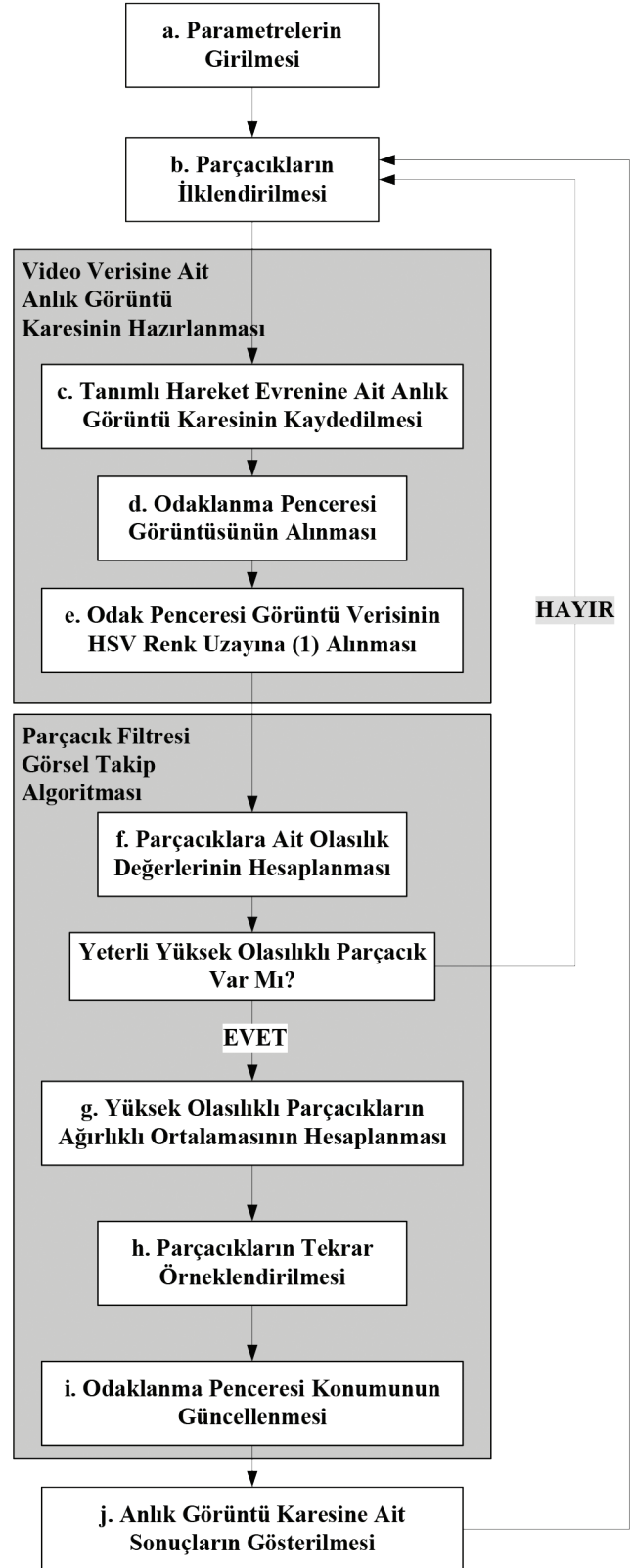
### 2.1 Sistemin Genel Yapısı

Odaklanma yetenekli görsel hedef takibi sistemimizin genel yapısı, geliştirilen sistem donanımı ve sistem yazılımı olarak iki ayrı parçada incelenebilir.

Bu alt sistemimize ait donanım, belirlenen bir ortam içerisindeki nesnenin hareketini kaydeden bir kamera ve takip yönteminin üzerinde çalıştırıldığı bir bilgisayardan oluşmaktadır.

Alt sisteme ait yazılım ise belirtilen bilgisayar üzerinde koşturulan ve Parçacık Filtresi yapısı başta olmak üzere bu bildiri kapsamında bilgileri verilen yöntemlerin gerçekleştirildiği uygulama yazılımıdır. İlgili yazılım MATLAB programı kullanılarak yazılmıştır.

Geliştirilen alt sisteme ait blok şema aşağıda verilmiştir.



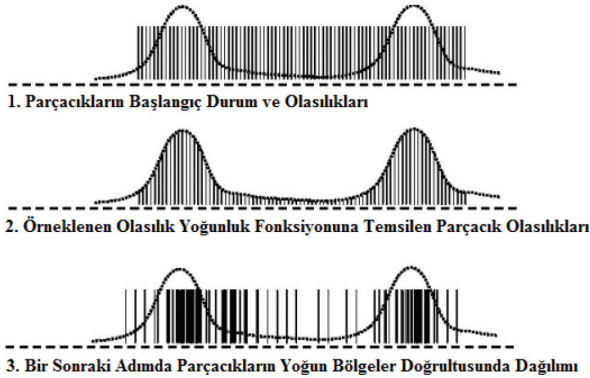
<sup>1</sup> HSV (Hue, Saturation, Value) Renk Uzayı – [Renk Özü, Doygunluk, Değer] Renk Uzayı

## 2.2 Sistem Algoritması ve Parçacık Filtresi'ne Yaklaşım

Bu bölüm dâhilinde, bütünlüğün sağlanması adına, öncelikle Parçacık Filtresi ve literatürdeki tanımlanmasına ilişkin bilgiler verilmiş, daha sonra da sistem blok şemasına ait bölümler, Parçacık Filtresi kullanarak geliştirdiğimiz yaklaşımımızla birlikte sunulmuştur. Sistemin yapısının detaylı anlatımı sırasında, anlaşılabilirliğin artırılabilmesi amacıyla örnek bir uygulama kullanılmıştır.

Literatürde sıkça kullanılan Parçacık Filtresi'nin temel fikri, herhangi bir olasılık yoğunluk fonksiyonunun bir parçacıklar kümesiyle temsil edilebilmesidir [13]. Tahmin edilmeye çalışılan olasılık yoğunluk fonksiyonun parametrelerinin kesin bir şekilde hesaplamayı hedefleyen Bayes Filtresi yaklaşımından farklı olarak, Parçacık Filtresi olasılık yoğunluk fonksiyonunun parçacıklarla temsilini öngörmekte ve kesin sonuç yerine yaklaşık bir sonucun bulunmasını hedeflemektedir. Bu durum Parçacık Filtresi'nin kullanıldığı durumlarda, verilen olasılık yoğunluk fonksiyonunun karakteristiğinden bağımsız olarak modellenebilmesi avantajını sağlamaktadır [13].

Parçacık Filtresi'nin bu avantajına karşın, sisteme dâhil edilen her parçacığın getirdiği hesaplama yükü bir dezavantaj teşkil etmektedir. Ancak gelişen teknolojinin ve hızlanan bilgisayarların sayesinde Parçacık Filtresi'nin hesaplama maliyetine ilişkin dezavantaj gün geçtikçe azalmaktadır [3]. Literatürde yer alan Parçacık Filtresi'nde, filtre dâhilinde kullanılan parçacıkların, uygulandıkları olasılık yoğunluk fonksiyonunda örneklendikleri olasılık değerleri göz önüne alınır. Bu değerlere bakılarak her adımda olasılığı daha yüksek olan parçacıkların yoğunlaştığı bölgelerde daha fazla sayıda parçacık ile örnekleme yapılması hedeflenir [13].



Şekil-1: Parçacık Filtresi Yöntemi Gösterimi (13)

Dolayısıyla Parçacık Filtresi'ne ait genel yöntem aşağıda verilen 3 adımda irdelenebilmektedir [13]:

- Olasılık Tahmini (Öngörü)
- Parçacıkların Güncellemesi
- Parçacıkların Tekrar Örnekleme

Parçacık Filtresi'nin tekrar örneklenmesi aşamasında yüksek olasılığa sahip parçacıkların varlığını sürdürüp, düşük olasılığa sahip parçacıkların göz ardı edilmesi durumunda "Parçacık Tükenmesi" problemi ortaya çıkmaktadır [13]. Gerekli önlem alınmadığında yeterli sayıda adımın ardından uygulamada kullanılacak parçacığın kalmaması söz konusudur.

Geliştirdiğimiz odaklanma yetenekli görsel hedef takibi sistemimizde de, Parçacık Filtresi'nin dağıtık olasılıksal gösterim özelliği uygulanmış olup, Parçacık Filtresi yapısı doğası gereği odaklanma parçacıkların yakalanan nesne bölümlerine dağılımları olarak yaklaşım çerçevesinde kullanılmıştır. Bu yaklaşım, sistem algoritması içerisinde parçalar halinde uygulanmış ve detaylandırılmıştır.

### a. Sistem Parametrelerinin Girilmesi

Sistem başlatılmadan önce aşağıdaki parametreler bir sefere mahsus olmak üzere ayarlanmaktadır:

- Hedef Renk: Hedef Renk parametresi olarak sisteme HSV renk uzayında bir rengi temsil edecek şekilde Hue (Renk Özü), Saturation (Doygunluk), Value (Değer) değerleri sisteme girdi olarak verilir. Verilen örnekte HSV değeri olarak örnekte kullanılan kırmızı objenin HSV değerine bakılmış ve hedef renk bilgisi bu doğrultuda kalibre edilmiştir.
- Odaklanma Penceresi Boyutları: İlgili en ve boy bilgileri sisteme girdi olarak verilir. 640x480 piksel boyutlarında olan örnek görüntülerde pencere genişliği 160, boyu 120 piksel olarak ayarlanmıştır.
- Odaklanma Penceresi Başlangıç Konumu: Odaklanmanın başlangıç konumu sisteme girdi olarak verilir. Alt sistemimizde herhangi ses duyumu olmaması sebebiyle, görüntü algısının olmadığı durumda odaklanma yapılmamaktadır. Bu nedenle verilen örnekte odaklanma penceresi görüntünün tam ortasında olacak şekilde ilklendirilmiştir.
- Filtredeki Parçacık Sayısı: Uygulanan Parçacık Filtresi'nde kullanılması istenen parçacık sayısı sisteme girdi olarak verilir. Verilen örnekte parçacık sayısı, odaklanma penceresi kapsamına denk gelen toplam piksel sayısının %5'i olacak şekilde seçilmiştir. Örnekte yer alan hareketli odaklanma penceresi içerisinde toplam 19200 piksel bulunmaktadır (160x120). Dolayısıyla toplam parçacık sayısı 960'tır.

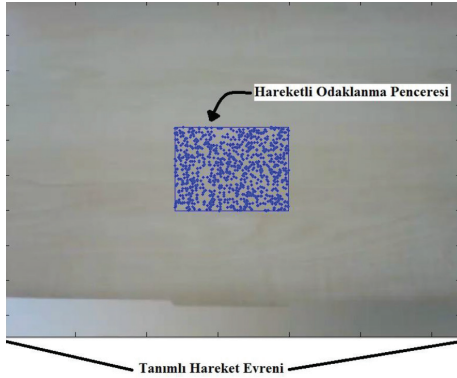
- Parçacık “VURDU” Olasılık Eşik Değeri: Kullanım yeri ve sebebi bildiri dâhilinde belirtilen bu parametre, sisteme girdi olarak verilmelidir. Verilen örnekte bu değer 0.3 olarak seçilmiştir.

- Vuran Parçacık Sayısı Eşik Değeri: Kullanım yeri ve sebebi bildiri dâhilinde belirtilen bu parametre, sisteme girdi olarak verilmelidir. Verilen örnekte bu değer 5 parçacık olarak seçilmiştir.

- Parçacık Konum ve Hız Gürültü Değeri: Parçacıklara ait hareket modelinin bir parçası olan ve kullanım sebebi bildiri dâhilinde belirtilen bu parametreler, sisteme girdi olarak verilir. Verilen örnekte bu değerler konum için 20 piksel, hız için ise 5 piksel/görüntü karesi olarak seçilmiştir.

- Maksimum Odaklanma Penceresi Hızı: Alt sistemin kullanılabileceği bir Humanoid’e ait odaklanma hareketinin ancak belirli fiziksel sınırlar dâhilinde mümkün olabileceği göz önüne alınarak odaklanma penceresinin hareket kabiliyeti sınırlandırılmıştır. Bu sebeple odaklanma penceresinin maksimum hızı sisteme girdi olarak verilir. Verilen örnekte bu değer her iki eksen de (dikey ve yatay) 20 piksel/görüntü karesi olarak ayarlanmıştır.

Yukarıda belirtilen verilere göre başlatılmış olan sisteme ait görsel aşağıda sunulmuştur:



Şekil-2: Sistem Başlangıcı Gösterimi

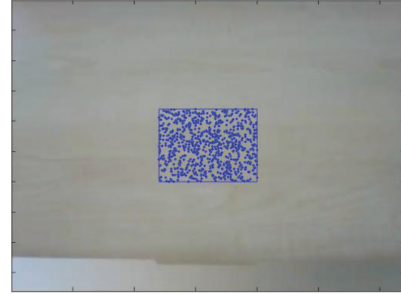
Bildirinin kalan kısmında yukarıda verilen parametrelere ilişkin kısaltmalar kullanılmış olup, bu kısaltmaların listesi aşağıda verilmiştir:

| Parametre                             | Kısaltma |
|---------------------------------------|----------|
| Hedef Renk                            | HR       |
| Odaklanma Penceresi                   | OP       |
| Filtredeki Parçacık Sayısı            | PS       |
| Parçacık “VURDU” Olasılık Eşik Değeri | PVOED    |

|                                   |       |
|-----------------------------------|-------|
| Vuran Parçacık Sayısı Eşik Değeri | VPSED |
| Parçacık Konum Gürültü Değeri     | PKGD  |
| Parçacık Hız Gürültü Değeri       | PHGD  |
| Maksimum Odaklanma Penceresi Hızı | MOPH  |

## b. Parçacıkların İlkendirilmesi

Hareketli objenin bir kamera ile görüntüsünün alındığı sistemimizde, başlangıç olarak nesnenin konumu, hızı ve hareket yönüne dair hiçbir bilgi önceden sisteme verilmemektedir. Sistem ilk çalıştırıldığında başlangıç tahmini tamamen rastgele yapılmaktadır (Şekil-3).



Şekil-3: Rastgele İlkendirilen Parçacıkların Gösterimi

Parçacıklar, OP’nin içerisine uniform dağılım kullanılarak dağıtılmıştır. Her bir parçacığın hem dikey hem de yatay eksen de denk geleceği piksele ilişkin olasılık bütün pikseller için eşit değerdedir.

## c. Tanımlı Hareket Evrenine Ait Anlık Görüntü Karesinin Kaydedilmesi

Algoritmamız verilen videonun her bir karesi için blok şemada verilen adımları döngüsel bir biçimde işleme almaktadır. Verilen örnekteki videoda bulunan görüntü karesi sayısı 288’dir. Bir başka deyişle blok şema içerisinde verilen ve C harfinden J harfine kadar gösterilen tüm bloklar verilen örnek kapsamında 288 kere çalıştırılmıştır. İşleme alınmadan önce her bir görüntü karesi bilgisi sisteme alınmaktadır. Buna ilişkin örnek görüntü Şekil-3’te verilmiştir.

Bu adımda alınan görüntü bilgisi RGB (Kırmızı-Yeşil-Mavi) renk uzayında olacak şekilde kaydedilmektedir. Verilen örnekteki Tanımlı Hareket Evreni’ne ait anlık görüntü karesinin boyutları 640 x 480 pikseldir. Kaydedilen bilgi her bir piksel için Red (Kırmızı), Green (Yeşil) ve Blue (Mavi) değeri olmak üzere 3’er adet olacak şekilde tutulmaktadır.

#### d. Odaklanma Penceresi Görüntüsünün Alınması

Alt sistemimizin yapısı gereği, sadece Odaklanma Penceresi (OP) içine düşen görüntü bilgisi görülen alan olarak tanımlanmaktadır. Bu sebeple girdi olarak verilen videonun her karesinde, tanımlı hareket evreni görüntüsünün alınmasının ardından, OP konumu bilgisine göre OP içerisine denk gelen görüntü parçası alınmakta ve sadece OP içerisinde kalan görüntü bölümü sisteme girdi olarak verilmektedir.

Bir önceki adımda alınan görüntü bilgisinin RGB renk uzayında olması sebebiyle, bu adımda alınan OP görüntüsüne ait kayıt da RGB renk uzayındadır.

Verilen örnekteki OP'ye ait görüntü parçasının boyutları 160 x 120 pikseldir. Kaydedilen bilgi her bir piksel için Red (Kırmızı), Green (Yeşil) ve Blue (Mavi) değeri olmak üzere 3'er adet olacak şekilde tutulmaktadır.

#### e. Odak Penceresi Görüntü Verisinin HSV Renk Uzayına1 Alınması

Bu adımda, bir önceki adımda çıkartılan OP görüntüsü renk uzayı RGB renk uzayından HSV renk uzayına çevirilmektedir. Bu çevirimin amacı, HSV renk uzayının insanların görsel algısına daha yakın özellikte oluşudur.

Belirtilen çevirim işlemi, MATLAB içerisinde tanımlı olan RGB'den HSV'ye çeviri altyapısı kullanılarak gerçekleştirilmiştir.

Verilen örnekte çevirim sonucu, OP'ye ait görüntü parçasının boyutları tekrar 160 x 120 pikseldir. Kaydedilen bilgi her bir piksel için Hue (Renk Özü), Saturation (Doygunluk) ve Value (Değer) değeri olmak üzere 3'er adet olacak şekilde tutulmaktadır.

#### f. Parçacıklara Ait Olasılık Değerlerinin Hesaplanması

Bu aşamada, alt sistemimiz dâhilinde veri olarak OP'ye içerisine denk gelen piksellerin HSV değerleri ile Parçacık Filtresi'ne dâhil edilen parçacıkların konum bilgileri bulunmaktadır. Bu iki bilgi kullanılarak, OP içerisindeki pikseller örneklenmekte ve her parçacığa ait HSV değeri elde edilmektedir.

Elde edilen parçacık HSV değerleri ve sisteme girdi olarak verilen Hedef Renk (HR)'e ait HSV değerleri kullanılarak, her parçacık için birer olasılık değeri hesaplanır.

Parçacık Numarası:  $k$

Anlık görüntü kare numarası:  $t$

Bir sonraki görüntü kare numarası:  $t + 1$

$k$  numaralı parçacığa ait:

$h_k$ : Hue Değeri

$s_k$ : Saturation Değeri

$v_k$ : Value Değeri

Sisteme girilen hedef renge ait:

$h_H$ : Hue Değeri

$s_H$ : Saturation Değeri

$v_H$ : Value Değeri

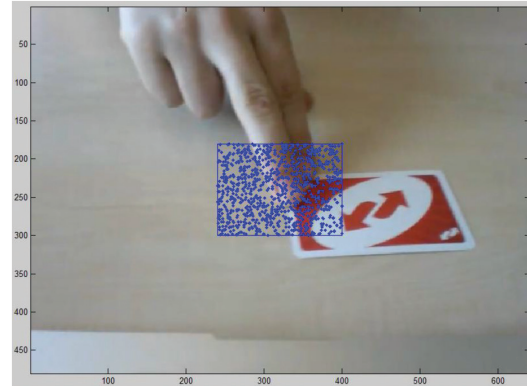
Parçacığa ait olasılık değeri:

$$w_k^t = \sqrt[4]{(h_k - h_H)^4 + (s_k - s_H)^2} \quad (1.1)$$

$$w_k^t = (w_k^t)^{-1} \quad (1.2)$$

Denklem-1: Parçacık Olasılıklarının Hesaplanması

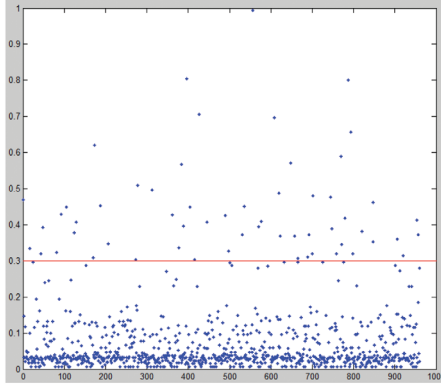
Hesaplama sırasında,  $k$  numaralı parçacığa ait olan ve hedef olarak verilen Hue ve Saturation değerleri arasında doğrusal bir uzaklık tanımlanmıştır. İlgili parçacığa ait olasılık ise bu uzaklığın tersi olarak tanımlanmıştır. Verilen örnekteki video görüntüsünün Şekil-4'te verilen bir karesi için belirtilen olasılık hesaplaması Şekil-5'te sunulmuştur.



Şekil-4: Örnek Görüntü Karesi ve Olasılıkları Hesaplanacak Parçacıkların Gösterimi

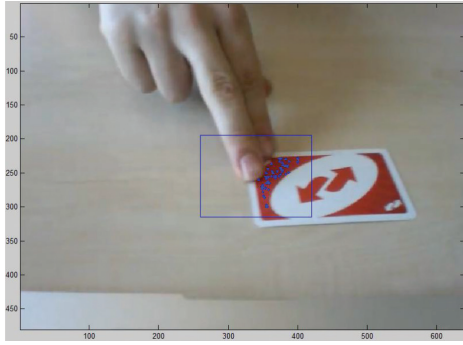
Olasılık hesaplamasının ardından, belirli bir eşik değeri seçilmiş ve bu değerden daha yüksek olasılığa sahip parçacıklar "VURDU" olarak tanımlanmıştır. Bu eşik değeri, Parçacık "VURDU" Olasılık Eşik Değeri (PVOED) olarak isimlendirilmiş ve verilen örnekte 0.3 olarak seçilmiştir (Şekil-5). "VURDU" tanımının oluşturulması sonucu, hedef rengin görülmediği durumlarda, dikkat odaklanmasının olmaması amaçlanmıştır. Ek olarak "VURDU" olarak hesaplanan parçacıklar ve konum bilgileri, "i. Odak Penceresi Konumunun Güncellenmesi"

bölümünde detayları verildiği şekilde odaklanma penceresinin güncellenmesinde kullanılmaktadır.



Şekil-5: Parçacıklara Ait Olasılık Değerleri

Şekil-5'te verilen örnek görüntü karesi için seçilen eşik değeri kırmızı bir doğru ile gösterilmektedir. Şekil-4'te verilen parçacıklar arasından "VURDU" olarak hesaplanan parçacıkların gösterimi Şekil-6'da verilmiştir.



Şekil-6: "VURDU" Parçacıklarının Gösterimi

Denklem-1 içerisinde yapılan işlem sonucu, 0.3 değerinin videodaki tüm kareler için tutarlı bir PVOED olduğu görülmüştür. PVOED değerinin değişiminin dikkat odaklanmasına olan etkisi "3. Deney Sonuçları ve Değerlendirme" bölümünde sunulmuştur.

Her adımda yapılan tahmin ile birlikte olasılık hesaplama işlemi, yaklaşımımız dâhilinde Parçacık Filtresi'nin "Olasılık Tahmini" adımının uygulanmasıdır.

#### g. Yüksek Olasılıklı Parçacıkların Ağırlıklı Ortalamasının Hesaplanması

Bu adımda, Şekil-6'da gösterilen ve "VURDU" olarak atanan parçacıkların ağırlıklı ortalamasının hesaplanması işlemi gerçekleştirilmektedir. Bu işlemin amacı, parçacıkların hesaplanan olasılık değerleri temel alınarak bir merkezin hesaplanmasıdır. Bu merkez bir bakıma "VURDU" olarak belirtilen parçacıkların ağırlık merkezi

olarak da düşünülebilir. Parçacıkların ağırlıklı ortalaması sadece belirli sayıda "VURDU" olarak atanan parçacık varsa yapılmaktadır. Bu sınırlamanın sebebi, herhangi bir parçacığa ait olasılık değerinin görüntüdeki gürültülerden kaynaklı olarak yüksek çıkması durumunda, odaklama penceresinin gürültüye denk gelen parçacığa doğru kaydırılmasını engellemektir. Bu amaçla oluşturulan eşik değeri, Vuran Parçacık Sayısı Eşik Değeri (VPSED) olarak isimlendirilmiş ve verilen örnekte 5 olarak seçilmiştir. Böylece vuran parçacık sayısı 5'ten az ise parçacıkların ağırlıklı ortalaması hesaplanmamakta, parçacıkların rastgele konumlandırılmasına devam edilmekte ve odaklanma penceresi konumu güncellenmemektedir. Parçacıkların ağırlıklı ortalaması aşağıdaki denklemler kullanılarak hesaplanmaktadır.

*Toplam parçacık sayısı (PS): n*

*Parçacık numarası: k*

*Anlık görüntü kare numarası: t*

*Bir sonraki görüntü kare numarası: t + 1*

*t anında k numaralı parçacık;*

*İlişkin olasılık değeri:  $w_k^t$*

*Yataydaki konumu:  $x_k^t$*

*Dikeydeki konumu:  $y_k^t$*

*Ağırlıklı ortalamanın yataydaki konumu:  $x_M^t$*

*Ağırlıklı ortalamanın dikeydeki konumu:  $y_M^t$*

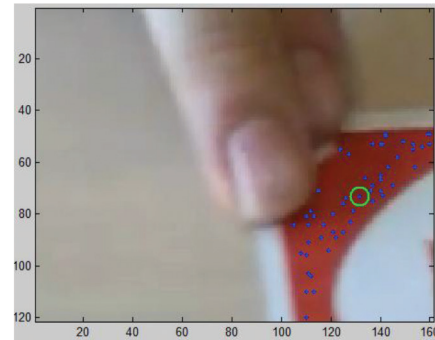
$$x_M^t = \frac{\sum_{k=1}^n x_k^t \cdot w_k^t}{n} \quad (2.1)$$

$$y_M^t = \frac{\sum_{k=1}^n y_k^t \cdot w_k^t}{n} \quad (2.2)$$

*t anında ağırlık ortalaması:*

$$M^t = [x_M^t \ y_M^t] \quad (2.3)$$

Denklem-2: Parçacıkların Ağırlıklı Ortalamasının Hesaplanması



Şekil-7: Parçacıkların Ağırlıklı Ortalamasının Hesaplanması Sonucu Gösterimi

## h. Parçacıkların Tekrar Örneklendirilmesi

Bu bölümde, parçacıkların bir sonraki görüntü karesi öncesinde tekrar örneklendirilmesi işlemi gerçekleştirilmektedir. Tekrar örneklendirme işlemi sadece, ağırlıklı ortalama değeri hesaplandıysa yapılmaktadır. Bir başka deyişle, eğer vuran parçacık sayısı tanımlanan VPSED'den az ise ağırlıklı ortalama hesaplaması ve parçacıkların tekrar örneklendirilmesi yapılmamaktadır.

Yapılan olasılık ve ağırlıklı ortalama hesaplanmasının ardından, alınan en son verilere göre hareketli objenin ağırlıklı ortalaması yakınlarında olduğu tahmin edilmektedir. Bu doğrultuda, parçacıkların hesaplanan konumu civarına hareketlendirilmesi gerekmektedir.

Sistemimizde parçacıklar hareket kabiliyetine sahip şekilde modellenmiş olup her adımda parçacık konumları ve hızları bir önceki konumlarına ve hızlarına göre güncellenebilmektedir. Belirtilen modelleme aşağıdaki denklemle tanımlanmıştır.

$t$  anında  $k$  numaralı parçacık;

İlişkin olasılık değeri:  $w_k^t$

Yataydaki konumu:  ${}_X x_k^t$

Dikeydeki konumu:  ${}_X y_k^t$

Yataydaki hızı:  ${}_V x_k^t$

Dikeydeki hızı:  ${}_V y_k^t$

Parçacık konumu modellemesi:

$${}_X x_k^{t+1} = {}_X x_k^t + {}_V x_k^t \quad (3.1)$$

$${}_X y_k^{t+1} = {}_X y_k^t + {}_V y_k^t \quad (3.2)$$

Denklem-3: Parçacıkların Hareket Modeli

Parçacıkların konumlarının değişebilmesi için Denklem 3.1 ve 3.2'de ve olarak belirtilen hız değerlerinin belirlenmesi gerekmektedir. Bu değerlerin hesaplandığı formüller aşağıdaki denklemde verilmiştir.

$X$  yönünde Konum Gürültüsü (Gauss):  ${}_X x_N$

$X$  yönünde Hız Gürültüsü (Gauss):  ${}_V x_N$

$Y$  yönünde Konum Gürültüsü (Gauss):  ${}_X y_N$

$Y$  yönünde Hız Gürültüsü (Gauss):  ${}_V y_N$

Merkezi  $\mu$ , varyansı  $\sigma$  olan Gauss Fonksiyonu kullanılarak türetilen rastgele değer:  $D(\mu, \sigma)$

Parçacık Konum Gürültü Değeri (PKGD):  ${}_X n$

Parçacık Hız Gürültü Değeri (PHGD):  ${}_V n$

$${}_X x_N = D(0, {}_X n) \quad (4.1)$$

$${}_X y_N = D(0, {}_X n) \quad (4.2)$$

$${}_V x_N = D(0, {}_V n) \quad (4.3)$$

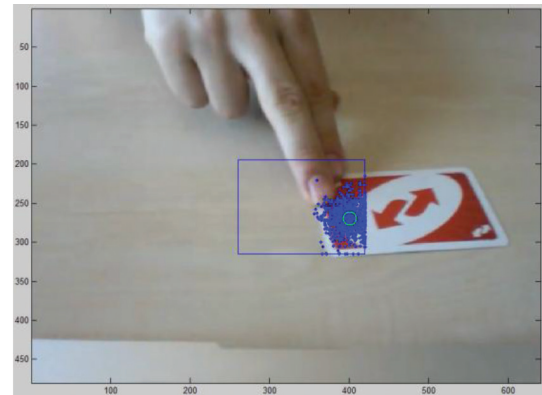
$${}_V y_N = D(0, {}_V n) \quad (4.4)$$

$${}_V x_k^t = \begin{cases} {}_X x_N + {}_V x_N & , \text{"VURDU"} \\ {}_X x_N + {}_V x_N + x_M^t - {}_X x_k^t & , \text{diğer} \end{cases} \quad (4.5)$$

$${}_V y_k^t = \begin{cases} {}_X y_N + {}_V y_N & , \text{"VURDU"} \\ {}_X y_N + {}_V y_N + y_M^t - {}_X y_k^t & , \text{diğer} \end{cases} \quad (4.6)$$

Denklem-4: Parçacık Hızlarının Güncellenmesi ve Parçacıkların Tekrar Örneklendirilmesi

Yukarıda verilen denklemin uygulanmasının ardından parçacıklar tekrar örneklendirilme konumlarına yönlendirilmişlerdir. Verilen örnek için parçacıkların konumlarına ait gösterim Şekil-8'de sunulmuştur.



Şekil-8: Parçacıkların Tekrar Örneklendirilmesi

Denklem-4 kapsamında verilen güncellemelerle, literatürdeki "Parçacık Tükenmesi" problemi de giderilmektedir. Belli bir adımda "VURDU" olarak tanımlanmayan parçacıkların göz ardı edilmesi yerine bir sonraki adımda, hesaplanan ağırlıklı ortalama

konumuna doğru hareketlendirilmesi sağlanmaktadır. Böylece her adımda toplam parçacık sayısı sabit tutularak, parçacıkların belli bir adım sayısından sonra tükenmesi engellenmektedir.

### i. Odaklanma Penceresi Konumunun Güncellenmesi Örneklendirilmesi

Görsel dikkat odaklanmasının sağlanabilmesi amacıyla, odaklanma penceresi hareketli bir şekilde modellenmiş olup, bu modellemeye ilişkin bilgiler Denklem-5 kapsamında tanımlanmıştır.

$t$  anında odaklanma penceresi merkezinin;

Yataydaki konumu:  $x_{OP}^t$

Dikeydeki konumu:  $y_{OP}^t$

Yataydaki hızı:  $v_{x_{OP}}^t$

Dikeydeki hızı:  $v_{y_{OP}}^t$

Odaklanma Penceresi hareket modellemesi:

$$x_{OP}^{t+1} = x_{OP}^t + v_{x_{OP}}^t \quad (5.1)$$

$$y_{OP}^{t+1} = y_{OP}^t + v_{y_{OP}}^t \quad (5.2)$$

Denklem-5: Odaklanma Penceresi Hareket Modeli

Odaklanma penceresi konumunun değişebilmesi için Denklem 5.1 ve 5.2'de ve olarak belirtilen hız değerlerinin belirlenmesi gerekmektedir. Bu değerlerin hesaplandığı formüller aşağıdaki denklemde verilmiştir.

Maksimum Odaklanma Penceresi Hızı (MOPH):  $max v_{OP}$

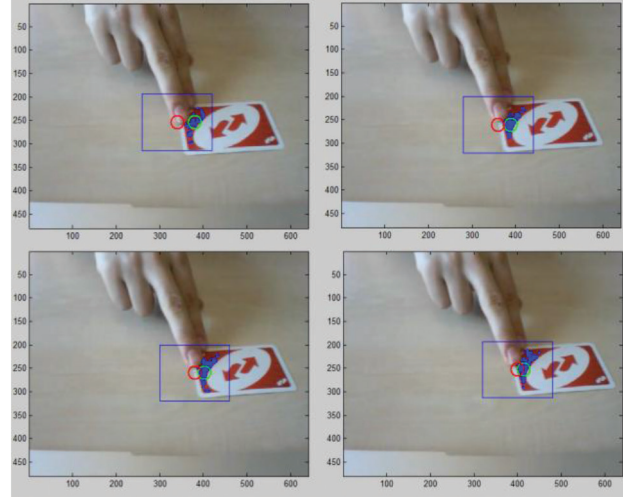
$$v_{x_{OP}}^t = \begin{cases} x_M^t - x_{OP}^t, & |x_M^t - x_{OP}^t| \leq max v_{OP} \\ max v_{OP}, & |x_M^t - x_{OP}^t| > max v_{OP} \end{cases} \quad (6.5)$$

$$v_{y_{OP}}^t = \begin{cases} y_M^t - y_{OP}^t, & |y_M^t - y_{OP}^t| \leq max v_{OP} \\ max v_{OP}, & |y_M^t - y_{OP}^t| > max v_{OP} \end{cases} \quad (6.6)$$

Denklem-6: Odaklanma Penceresi Hızının Hesaplanması

Yukarıda verilen denklemin uygulanmasının ardından odaklanma penceresi görsel hedef takibini gerçekleştirmektedir. Verilen örnek için odaklanma penceresinin hareketine ilişkin gösterim Şekil-9'da sunulmuştur.

Şekil-9'da kırmızı daire ile odaklanma penceresinin merkezi, yeşil daire ile Şekil-7'de gösterilen ağırlıklı ortalama konumu gösterilmiştir. Birbirini takip eden görüntü karelerinde, odaklanma merkezinin gittikçe ağırlıklı ortalama konumuna yaklaştığı görülmüştür.



Şekil-9: Odaklanma Penceresinin Hareketi Gösterimi

### j. Anlık Görüntü Karesine Ait Sonuçların Gösterilmesi

Bu adımda, kullanılan algoritma sonucu hesaplanan değerler ve parçacıklar görüntü üzerinde gösterilmektedir. Blok şemada sunulan döngü kapsamında görüntü karelerinin ardı ardına gösterilmesi ile akıcı bir görüntü çıktısı elde edilmektedir.

### 3. Deney Sonuçları ve Değerlendirme

Yapılan deneyde 640x480 piksel boyutlarında toplam 288 görüntü karesi içeren bir video kullanılmıştır. Video süresi 10.6 saniye, videonun saniyedeki görüntü karesi sayısı ise 27.16 karedir. Yapılan denemelerde gerçek zamanlı bir görüntü çıktısının alınabildiği görülmüş ve çıktı olarak oluşturulan görüntünün saniyedeki görüntü karesi sayısının 11'in altına hiç inmediği gözlenmiştir. Bu istatistiklere ulaşıldığı durumda sistem parametreleri aşağıda verilmiştir:

| Parametre | Tipik Değer |
|-----------|-------------|
| PS        | 960 (%5)    |
| PVOED     | 0.3         |
| VPSED     | 5           |
| PKGD      | 20          |
| PHGD      | 5           |
| MOPH      | 20          |

Sistem parametreleri değiştirilerek yapılan birçok denemenin ardından güvenilir görsel takibin sağlanabilmesi için sınır değerler belirlenmiştir. Belirlemeler sırasında aynı anda sadece bir değer

değiştirilmiş, değiştirilen değer dışında kalan parametreler yukarıdaki tipik değerlerinde sabit alınmıştır.

Belirlenen değerler ve ilgili yorumlar aşağıda sunulmuştur:

| Parametre | Mutlak Sınırlar         |
|-----------|-------------------------|
| PS        | 826 (%4.3) – 3840 (%25) |
| PVOED     | 0.15 – 0.40             |
| VPSED     | 3 - 7                   |
| PKGD      | 5 – 35                  |
| PHGD      | 0 - 15                  |
| MOPH      | 7 - 25                  |

- Parçacık Filtresi kapsamında kullanılan parçacık sayısının belirli bir düzeyin (826 adet) altına inmesi durumunda takip için yeterli sayıda vuran parçacık atamasının yapılamadığı durumlar olduğu ve bu durumun görsel takibin performansını etkilediği görülmüştür. Benzer şekilde filtredeki parçacık sayısının belirli bir düzeyin (3840 adet) üzerine çıkarılması hesaplama sürelerinin uzamasına ve performansın kötü yönde etkilenmesine sebep olmaktadır.
- Parçacık "VURDU" Olasılık Eşik Değeri'nin belirli bir seviyenin (0.15) altına indirilmesi odaklanmanın hedef rengi görmediği durumlarda bile hedef renk görmüş-çesine değişmesine sebep olduğu gözlenmiştir. Benzer şekilde bu seviyenin çok yükseltilmesi durumunda (0.40'tan büyük) hedef rengin görülmesine rağmen tespitin yapılamadığı durumlar olduğu görülmüştür.
- Vuran Parçacık Sayısı Eşik Değeri'nin belirli bir seviyenin (3 adet) altında veya belirli bir seviyenin (7 adet) üstünde olması durumunda PVOED parametresine benzer bir şekilde hedefin görülmesi/görülmemesi kararında bozulma olduğu görülmüştür.
- Parçacık Konum ve Hız Gürültü Değerleri'nin belirli bir seviyenin altında oluşu, hareketli hedefin hızlı olması durumunda hedefin yakalanamaması yönünde problemler oluşturduğu görülmüştür. Bu değerlerin aşırı miktarda artırılması durumunda, dağınık bir odaklama oluşmakta ve görsel takip performansı kötü yönde etkilenmektedir.
- Maksimum Odaklanma Penceresi Hızı değerinin belirli bir seviyenin (7 piksel/kare) altında olması durumunda hareketli hedefe odaklanmanın aksadığı durumların oluşabildiği gözlenmiştir. Bu değerlerin belirli bir seviyenin (25 piksel/kare) üzerine çıkarılması, odaklanma penceresinin hareketinin dengesiz bir şekilde hareket etmesine ve görsel takip performansının negatif yönde etkilenmesine sebep olmaktadır.

#### 4. Kaynaklar

- (1) Govaers, F., ve Wieneke, M.. "Rao-Blackwellized Particle Filter for Security Surveillance."
- (2) Hue, C., J-P. Le C., ve Perez, P.. "Tracking multiple objects with particle filtering." Aerospace and Electronic Systems, IEEE Transactions on 38.3 (2002): 791-812.
- (3) Ristic, B., Arulampalam, S. ve Gordon, N.. "Beyond the Kalman filter: Particle filters for tracking applications." Artech House Publishers, 2004.
- (4) Florin, C., Paragios, N. ve Williams, J.. "Particle filters, a quasi-monte carlo solution for segmentation of coronaries." Medical Image Computing and Computer-Assisted Intervention-MICCAI 2005. Springer Berlin Heidelberg, 2005. 246-253.
- (5) Black, M.J. ve Jepson, A.D.. "A probabilistic framework for matching temporal trajectories: Condensation-based recognition of gestures and expressions." Computer Vision ECCV'98. Springer Berlin Heidelberg, 1998. 909-924.
- (6) Vijayakumar, S., et al. "Overt visual attention for a humanoid robot." Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on. Vol. 4. IEEE, 2001.
- (7) Itti, L.. "Real-time high-performance attention focusing in outdoors color video streams." Electronic Imaging 2002. International Society for Optics and Photonics, 2002.
- (8) Arulampalam, M.S., et al. "A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking." Signal Processing, IEEE Transactions on 50.2 (2002): 174-188.
- (9) Rekleitis, I.M. "A particle filter tutorial for mobile robot localization." Centre for Intelligent Machines, McGill University 3480 (2004).
- (10) Campillo, F.. "Simultaneous localization and mapping (SLAM) via particle filtering."
- (11) Miyazato, K., et al. "Real-time estimation of human visual attention with dynamic Bayesian network and MCMC-based particle filter." Multimedia and Expo, 2009. ICME 2009. IEEE International Conference on. IEEE, 2009.
- (12) Glas, D.F., et al. "Laser tracking of human body motion using adaptive shape modeling." Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on. IEEE, 2007.
- (13) Kaijen, H., Plinval-Salgues, H. ve Miller, J.. "Particle Filters and Their Applications" Cognitice Robotics, Nisan 2005
- (14) Cuevas, E., Zaldivar, D. ve Rojas, R. "Neurofuzzy Prediction for Gaze Control" Canadian Journal of Electrical and Computer Engineering 34 (1) , art. no. 5291203 , pp. 15-20, 2009
- (15) Kuhn, B., Schauerte, B., Kroschel, K., Stiefelhagen, R.. "Multimodal saliency-based attention: A lazy robot's approach" IEEE International Conference on Intelligent Robots and Systems , art. no. 6385515 , pp. 807-814, 2012