İstanbul Ticaret Üniversitesi Fen Bilimleri Dergisi Yıl: 9 Sayı: 18 Güz 2010 s. 45-63

NESNE TAKİP YÖNTEMLERİNİN SINIFLANDIRILMASI

Muhammed Fatih TALU*

Geliş: 16/06/2010 Kabul: 05/11/2010

ÖZET

Bu çalışmanın amacı, video imgelerindeki hareketli nesnelerin takibini gerçekleştirebilen nesne takip yöntemlerini belirli kıstaslar kullanarak farklı kategoriler altında sınıflandırmak ve bu alandaki yeni yönelimleri tanımlamaktır. Nesne takibi, video kaydı içerisinde hareket etmekte olan nesnelerin hareket yörüngelerini tahmin etme problemidir. Nesne hareketlerinin ve şekillerinin karmaşıklığı, nesne görünümünün engellenebilmesi, ortamdaki ışık miktarının değişimi, imgeler arasındaki veri bağı probleminin çözülme ihtiyacı ve gerçek zamanlı uygulama gereksinimleri, nesne takip problemini zorlaştırmaktadır. Literatürde, bahsedilen nesne takip problemine tatmin edici çözümler sunabilen birçok yöntem bulunmaktadır. Fakat, takip edilen nesne özelliklerinin ve ortam şartlarının farklı oluşu önerilen yöntemlerin kategorize edilebilmesini engellemiştir. Bu çalışmada, dokuz farklı kıstas belirlenerek herbir kıstas altında önerilen takip yöntemlerinin davranışları gözlemlenerek yöntemler sınıflandırılmıştır.

Anahtar Kelimeler: Nesne Takibi, Kalman Filtre, Parçacık Filtre, Çekirdek Takibi, Siluet Takibi.

CLASSIFICATION OF OBJECT TRACKING METHODS

ABSTRACT

The goal of this article is to review the state of the art tracking methods which can track moving objects in video files, classify them into different categories and identify new trends in this studying field. Object tracking is a trajectory estimation problem of moving objects in video. Due to complex object moving, complex object shape, occlusion of all or a portion of the object appearance, changing the amount of ambient light and real time application requirements, the estimation problem of object trajectory is challenging problem. In literature, there are many methods which are able to provide satisfactory solutions for object tracking problem. However, due to be complex of object shape and movement and different of environmental conditions, the proposed methods have been prevented to be categorized. In this study, we classify the proposed tracking methods by taking into account each of the determined nine different criteria.

Keywords: Object Tracking, Kalman Filter, Particle Filter, Kernel Tracking, and Silhouette Tracking.

** Fırat Üniversitesi, Enformatik Bölümü, 23119, Elazığ, Türkiye. fatihtalu@gmail.com..*

1. GİRİŞ

Nesne takibi, "bilgisayarlı görme" alanı içerisinde önemli bir göreve sahiptir. En basit tanımıyla, nesne takibi, video kaydı içerisinde hareket etmekte olan bir nesnenin hareket yörüngesini tahmin etme problemidir. Diğer bir deyişle, bir nesne takip edicinin (object tracker), ardışık video imgeleri (görüntüleri) içerisinde hareket etmekte olan nesneleri algılayarak her bir nesneye eşsiz bir etiket verme işlemidir (http://en.wikipedia.org/wiki/Video_tracking.[10,10,2009,WEB]). Hızlı veri işleme özelliğine sahip bilgisayarların artması, yüksek kaliteye sahip ucuz kameraların üretilmesi ve otomatik video analizi için gerekli olan ihtiyacın giderek artması, nesne takibi algoritmalarına karşı olan ilgiyi oldukça arttırmıştır.

Nesne takibi ile ilgili literatürde doyurucu sayıda yaklaşım önerilmiştir. Önerilen bu yaklaşımların temel farkları aşağıdaki sorulara verilen yanıtlar yardımıyla ortaya çıkmaktadır.

- Nesne takibi için hangi nesne sunumu daha uygundur?
- Takip işleminde nesnelere ait hangi özellikler kullanılmalıdır?
- Belirlenen nesne özellikleri nasıl modellenir?

Bu sorulara verilen yanıtlar, nesne takip işleminin gerçekleştiği ortama bağlı olmakla birlikte tasarlanacak algoritma yapılarının da belirlenmesini sağlar (Yilmaz vd., 2006).

2. NESNE TAKİP YÖNTEMLERİ

Nesne takip yöntemleri iki temel görevi içermektedir (Forsyth vd., 2005): 1) takip edilecek nesnenin güncel imge içerisindeki konumunun tespit edilmesi (object detection, lifting); 2) imgeler boyunca konumları belirlenen nesneler arasındaki veri bağı ilişkisinin kurulması (data association). Bu iki görev bağımsız gerçekleştirilebileceği gibi bir arada da çalıştırılabilir. Bağımsız çalıştırıldığında, ilk olarak nesne yakalama algoritmaları yardımıyla ardışık imgelerdeki hareketli nesneler yakalanır. Daha sonra, bu nesneler arasında var olan veri bağı ilişkileri elde edilir. Görev birlikte çalıştırıldığında ise, güncel nesne bilgilerini hesaplayabilmek için güncel gözlem bilgisi ve öncesel nesne bilgisi birlikte değerlendirilir.

Bahsedilen her iki nesne takip yöntemi de takip edilecek nesneyi şekil ve görünüm modelleri ile ifade eder. Nesne şekli, yapabileceği hareketi sınırlar. Örneğin; takip edilecek nesne sadece bir nokta ile ifade edilirse, o zaman nesne hareketlerini ifade etmek için basit bir dönüştürme modeli yeterli olabilir. Oysa nesnenin elips gibi geometrik bir şekle sahip olduğu bir durumda, parametrik hareket modelleri (affine veya projective dönüşüm modelleri) kullanılabilir. Bu sunum şekilleri yardımıyla geometrik şekli ve sınırları net olan nesnelerin hareketleri modellenebilir. Sınırları net olmayan nesneler içinse siluet veya kenar bilgisi, daha tanımlayıcı bir sunum sağlayabilir. Ayrıca, bu tür nesnelerin hareketlerini modellemek için parametrik veya parametrik olmayan modeller kullanılabilir (Sheskin, 2003).

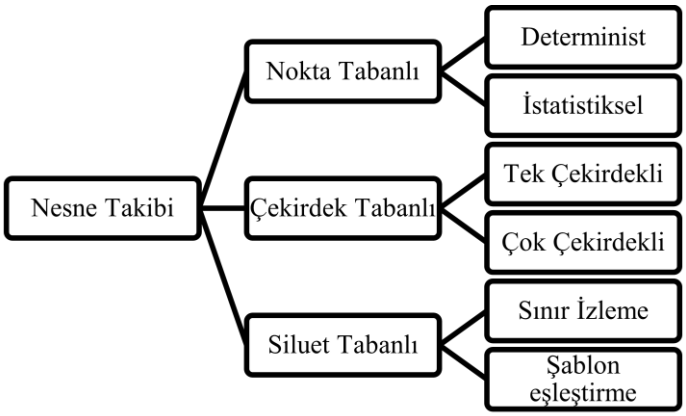
Nesne hareketleri veya görünümleri üzerinde belirli sınırlamalar yapılarak nesne takip problemi basitleştirilebilir. Hemen tüm nesne takip algoritmaları, nesne hareketlerinin ani değişimlere değil de yumuşak hareketlere sahip olduğunu varsayar. Ayrıca, nesne hareketlerinin sabit hız veya ivmeye sahip olduğu düşünülürse nesne takip işlemi oldukça basitleşir. Buna ilaveten, nesnelerin sayıları, büyüklükleri, görünümleri veya sahip oldukları fiziksel şekillerin önceden belirtilmesiyle de, nesne takip algoritmalarının karmaşıklığı azaltılır.

Nesne takibinde karşılaşılan temel zorluklar aşağıda listelendiği gibidir (Bettencourt ve Somers, 2007):

- 3 boyutlu gerçek dünya verilerinin 2 boyutlu görüntü alanına yansıtılmasıyla meydana gelen bilgi kaybı
- Görüntü üzerinde meydana gelen gürültüler
- Nesne hareketlerinin karmaşık oluşu
- Nesnelerin ayırt edilebilir fiziksel bir yapıya sahip olamayışları
- Nesne görünümünün bir kısmının veya tamamının engellenmesi (occlusion)
- Karmaşık nesne şekilleri
- Ortamdaki ışık miktarının değişimi
- Gerçek zamanlı uygulamaların gereksinimleri

Şekil 1'de nesne takip yöntemleri takip türüne göre sınıflandırılarak gösterilmiştir. Buna göre üç temel nesne takip yöntemi bulunmaktadır: 1) nokta tabanlı; 2) çekirdek tabanlı; 3) siluet tabanlı. Bu yöntemlerle ilgili literatürde yer alan çalışmalar Tablo 1'de listelendiği gibi yöntemlerin kullanıldığı uygulama alanları da aşağıdaki gibi listelenebilir:

- Hareket tabanlı tanımlama (Örn: Hareket yörüngesi bilinen bir tümörünün akciğer üzerindeki konumun tespiti (Berbeco vd., 2005)).
- Gizlice izleme (Örn: Şüpheli aktivitelerin veya istenmeyen hareketlerin tespit edilebilmesi için sabit bir ekran görüntüsünün izlenmesi (Haritaoglu vd., 2000)).



Şekil 1. Nesne takibi yöntemlerinin sınıflandırılması (Yilmaz vd., 2006)

Tablo 1. Nesne takip yöntemleri ile ilgili literatürde bulunan örnek çalışmalar

Kategoriler	Örnek Çalışmalar
Nokta Tabanlı Nesne Takibi	
• Deterministlik Yöntemler	Haritaoglu vd., 2000. Salari ve Sethi, 1990. Veenman vd., 2001.
• İstatistik Yöntemler	Xue vd., 2008. Small vd., 2008. Zhao vd.,2008. Dong ve Desouza, 2009. Chang vd., 2008, Leven ve Lanterman, 2009. Oh vd., 2009.
Çekirdek Tabanlı Nesne Takibi	
• Tek Boyutlu	Yilmaz, 2007. Comaniciu vd., 2003. Comaniciu ve Meer, 2004.
• Çok Boyutlu	Fieguth ve Terzofoulos,1997. Fan vd., 2005
Siluet Tabanlı Nesne Takibi	
• Sınır Değerlendirme	Cootes vd.,1998. Cootes vd., 1995. Cordea vd., 2008. Lee vd., 2007.
• Şekil Eşleştirme	Munder vd., 2008. Nascimento ve Marques, 2007. Zhu vd., 2007.

- Video indeksleme (Örn: Çoklu ortam veritabanında bulunan videoların, içerdiği nesneler ve bu nesnelerin hareketleri şeklinde kodlanarak sisteme kaydedilmesi, nesne adları veya hareketleri kullanılarak video kayıtlarına hızlı bir şekilde ulaşılabilmesi).
- İnsan bilgisayar etkileşimi (Örn: İnsan vücut azaları takip edilerek insanın yürüdüğü veya koştuğu sonucunun üretilmesi (Dong ve DeSouza, 2009)).
- Trafik izleme (Örn: Trafik akışını düzenlemek için gerçek zamanlı trafik bilgilerinin toplanması.
- Araç yolculuğu (Aracın şoförsüz seyahat edebilmesi veya yol çizgilerini takip edebilmesi).

Bu bölümün alt başlıklarında modern nesne takip yöntemleri hakkında detaylı bilgi verilmektedir.

2.1. Nokta tabanlı nesne takibi

Bu takip yönteminde takip edilen her bir nesne tek bir nokta ile ifade edilir. Bu yöntem ile güncel imgede takip edilecek her bir nesneye bir nokta aktarıldıktan sonra bu noktalar ile önceki imgede tespit edilen noktalar arasındaki veri bağı ilişkisinin doğru bir şekilde oluşturulması beklenir. Bu problemin çözümü için şu iki adım sırayla gerçekleştirilir: 1) güncel imgede nesne yakalama (her biri nokta ile ifade edilir); 2) bu noktalar ile önceki imgede tespit edilen noktalar arasındaki nokta benzerlik değerlerinin hesaplanması. Nokta benzerliğinin hesaplanması işlemi, özellikle nesne görünümünün kaybolması, yanlış nesne yakalanması, nesnenin imgeye ilk girişi veya çıkışı gibi durumlarda karmaşık bir hale dönüşür. Bu alandaki yöntemler genellikle iki alt kategoride incelenir (Yilmaz, 2006): Deterministlik ve istatistiksel. Aşağıda iki yöntemle ilgili detaylı bilgi verilemektedir.

2.1.1. Deterministlik Yöntemler

Deterministlik yöntemler, ardışık imgelerde bulunan nesnelerin birbirlerine bağlanma maliyetlerini tanımlar. Nesneler arasındaki bağlanma maliyetinin tanımlanması için nesne hareketleri üzerinde aşağıda belirtilen sınırlamalar göz önüne alınır (Yilmaz, 2006):

- *Yakınlık:* Bir imgeden diğer bir imgeye geçerken nesne pozisyonlarının önemli ölçüde değişmeyeceğini ima eder.
- *Maksimum hız:* Nesnelerin hızları üzerinde bir üst sınır değeri tanımlanır ve sadece nesnelerin etrafında dairesel bir komşuluk içerisinde kalan muhtemel nesne adaylarının benzerlik değerleri göz önüne alınır.
- *Küçük hız değişimi (yumuşak hareket):* Nesnenin hız yönünün önemli bir ölçüde değişmeyeceğini ima eder.
- *Genel hareket:* Küçük bir komşuluktaki nesnelerin hızlarının benzer olması için sınırlama uygulanır. Bu sınırlama çoklu noktalar ile sunulan nesneler için uygundur.
- *Katılık:* 3-boyutlu dünyadaki nesneler genellikle katı bir biçime sahiptir. Bundan dolayı güncel bir nesne üzerindeki herhangi iki nokta arasındaki mesafenin değişmeyeceği düşünülür.

Bahsedilen sınırlamalar sadece deterministlik yöntemlerde değil aynı zamanda istatistiksel yöntemlerde de kullanılabilir.

2.1.2. İstatistiksel Yöntemler

Video algılayıcılarından elde edilen ölçümler her zaman gürültüye sahiptir. Ayrıca, nesne hareketleri birtakım istenmeyen etkilere maruz kalır. İstatistiksel yöntemler, nesnenin durum (pozisyon, hız ve ivmesinin) tahmini boyunca ölçüm değerlerini ve model belirsizliklerini göz önüne alarak nesne takibi problemini çözmeye çalışır. Bu yöntemler pozisyon, hız ve ivme gibi nesne özelliklerini modellemek için durum uzay yaklaşımını kullanırlar (Doucet ve Johansen, 2008). Ölçümler genellikle imgelerdeki nesne pozisyonlarını içerir. Bu noktadan sonra istatistiksel yöntemlerin

imge içerisindeki nesnelere ait durum tahminini nasıl formülleştirdiğini ve hangi çözümleri sunduğunu detaylı bir şekilde incelenir. Buna göre problemin tanımı için Kalman filtreleme yöntemi ve parçacık filtreleme yöntemi kullanılır. (Dong ve Desouza, 2009).

İmge üzerinde hareket eden bir nesnenin istatistiksel olarak durum tahmin problemi şu şekilde formüle edilebilir (Xue vd., 2008). Takip edilecek nesnenin durum bilgisi (örneğin pozisyon) $\boldsymbol{X^t}: \boldsymbol{t} = \mathbf{1}, \mathbf{2}, \ldots$ şeklinde bir dizi olarak tanımlanır. Zaman boyunca durum üzerindeki dinamik değişim denklem (1) ile ifade edilir:

$$X^t = f(X^{t-1}) + W^t \quad [1]$$

$W^t: t = 1,2 \ldots$ beyaz gürültüdür. Ölçüm verisi ile durum değişkeni arasındaki ilişki ise denklem (2) ile ifade edilir.

$$Z^t = h(X^t) + N^t \quad [2]$$

N^t'de beyaz gürültüdür ve W^t den bağımsızdır. İstatistiksel yöntemlere dayalı nesne takip edicilerin temel amacı, t anına kadarki tüm ölçüm değerlerini göz önüne alarak X^t durum değişkenini tahmin etmektir. Bir başka diyişle, sonrasal olasılık yoğunluk fonksiyonunu (posterior probability density function) $p(X^t|Z^{1,\ldots,t})$ elde etmektir. Teorik olarak en uygun çözüm, problemi iki adımda çözen tekrarlamalı Bayes filtresi yöntemini kullanmaktır. Bu adımlar tahmin ve düzeltme adımıdır. Tahmin adımı, dinamik bir eşitlik kullanır ve güncel durumun $t-1$ anındaki öncesel olasılık yoğunluk fonksiyonunu (prior probability density function) $p(X^t|Z^{1,\ldots,t-1})$ hesaplanır. Daha sonra sonrasal yoğunluk fonksiyonunun hesaplanabilmesi için güncel ölçümün maksimum olabilirlik fonksiyonunu $p(Z^t|X^t)$ kullanılır. Buna göre sonrasal yoğunluk fonksiyonu denklem (3)'de gösterildiği gibi hesaplanır.

$$p(X^t|Z^{1,\ldots,t}) = \frac{p(X^t|Z^{1,\ldots,t-1})\, p(Y^t|X^t)}{p(Y^t|Z^{1,\ldots,t-1})} \quad [3]$$

Denklem (3)'deki $p(Y^t|Z^{1,\ldots,t-1})$ normalizasyon katsayısıdır. Ekranda sadece tek nesnenin olması durumunda, nesnenin durum bilgisi bahsedilen iki adım kullanılarak rahatlıkla tahmin edilebilir. Diğer taraftan, ekranda birden fazla nesnenin olması durumunda, elde edilen ölçümler ile ilgili nesneler arasında gerekli bağlantıların kurulma ihtiyacı ortaya çıkar. Bu nedenle çoklu nesnelerin takibi problemi için veri bağı ve durum tahmini problemlerinin birleştirilmiş bir çözümüne ihtiyaç duyulur.

Tek nesnenin olduğu bir durumda, eğer f^t ve h^t fonksiyonları doğrusal ve nesnenin başlangıç durumu X^1 ve gürültü değeri Gaussian dağılımına sahipse, o zaman en uygun çözüm Kalman filtre tarafından elde edilebilir. Eğer nesnenin başlangıç durumu ve sistem gürültüsü Gaussian dağılımına sahip değilse o zaman en uygun çözüm Parçacık filtreleme yöntemiyle elde edilir.

Kalman Filtreleme Yöntemi

Kalman filtreler, durum vektörü bir Gaussian dağılımına sahip lineer sistemlerin durum tahminlerini gerçekleştirmek için kullanılır (Leven ve Lanterman, 2009). Tahmin ve düzeltme gibi iki adımdan oluşur. Tahmin adımı, değişkenlerin yeni durumu tahminini yapabilmek için durum modelini kullanır. Şöyle ki:

$$\bar{X}^t = DX^{t-1} + W \quad [4]$$

$$\bar{\Sigma}^{\mathrm{t}} = \mathrm{D}\Sigma^{\mathrm{t}-1}\mathrm{D}^{\mathrm{T}} + \mathrm{Q}^{\mathrm{t}} \quad [5]$$

$\bar{X}^t$ ve $\bar{\Sigma}^t$, t anındaki durum ve kovaryans tahminleridir. D, t ile $t-1$ anındaki durum değişkenleri arasındaki ilişkiyi tanımlayan durum dönüşüm matrisidir. Q, W gürültüsünün kovaryansıdır. Benzer bir şekilde, düzeltme adımı güncel gözlem değerini Z^t nesne durumunu güncellemek için kullanır.

$$K^t = \bar{\Sigma}^t M^t \left[\mathrm{M}\bar{\Sigma}^{\mathrm{t}}\mathrm{M}^{\mathrm{t}} + \mathrm{R}^{\mathrm{t}}\right]^{-1} \quad [6]$$

$$\mathrm{X}^{\mathrm{t}} = \bar{\mathrm{X}}^{\mathrm{t}} + \mathrm{K}^{\mathrm{t}} \underbrace{\left[\mathrm{Z}^{\mathrm{t}} - \mathrm{M}\bar{\mathrm{X}}^{\mathrm{t}}\right]}_{\mathrm{v}} \quad [7]$$

$$\Sigma^{\mathrm{t}} = \bar{\Sigma}^{\mathrm{t}} - \mathrm{K}^{\mathrm{t}}\mathrm{M}\bar{\Sigma}^{\mathrm{t}} \quad [8]$$

Yukarıdaki denklemlerde v yenilik olarak adlandırılır ve M ölçme matrisidir. K, durum modellerinin yayılımı için kullanılan kalman kazancıdır. Dikkat edilmesi gereken nokta şudur ki; güncellenen X^t durumu hala bir Gaussian dağılımıdır. Bu durumda f^t ve h^t fonksiyonları doğrusal değildir ve Taylor seri açılımları kullanılarak fonksiyonlar doğrusallaştırılabilir. Bahsedilen bu filtreleme tekniği literatürde *genişletilmiş kalman filtre* (extended kalman fitler) olarak bilinir.

Parçacık Filtreleme Yöntemi

Kalman filtresinin bir dezavantajı, durum değişkenlerinin Gaussian dağılımına sahip olma zorunluluğudur. Böylelikle, kalman filtresi gaussian dağılıma sahip olmayan sistemler için zayıf bir tahmin edicidir (Xue vd., 2008). Bu dezavantaj parçacık filtreleme yöntemiyle çözülebilir (Small vd., 2008). Parçacık filtrede, t anındaki şartsal durum yoğunluğu π ağırlığına sahip N tane parçacık içeren örnek küme ile ifade edilir. Her bir parçacık $\left\{s_t^{(n)}: n = 1, \dots, N\right\}$ ve her bir parçacığın ağırlığı ise $\pi_t^{(n)}$ (örnekleme olasılığı) olarak gösterilir. Ağırlıklar parçacığın önemi veya onun gözlenme frekansı olarak tanımlanabilir. Her bir $\left(s_t^{(n)}, \pi_t^{(n)}\right)$'nin hesaplama maliyetinin azaltılması için birikmiş (katlanmış) bir ağırlık $c^{(n)}$ de aynı zamanda hafızaya yüklenir ($c^{(N)} = 1$). t anındaki yeni örnekler $t-1$ anındaki $S_{t-1} = \left\{\left(s_{t-1}^{(n)}, \pi_{t-1}^{(n)}, c_{t-1}^{(n)}\right): n = 1, \dots, N\right\}$ örnekleme şemasıyla elde edilir. En genel

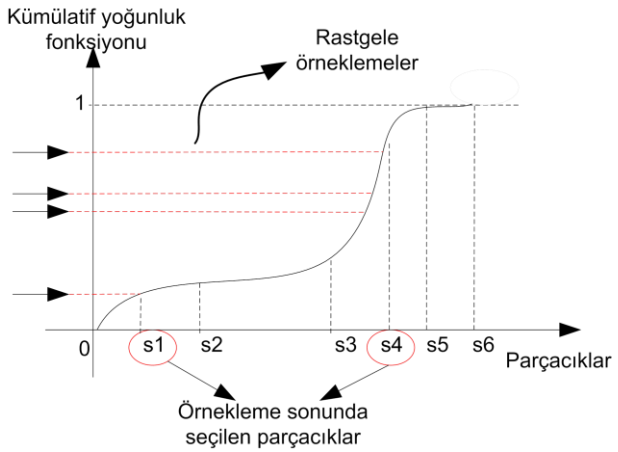
örnekleme şeması aşağıda bahsedildiği gibi *önem örnekleme* şemasıdır. Bu şema, *seçme, tahmin* ve *düzeltme* olmak üzere üç temel adımın tekrarlı bir şekilde gerçekleşmesi prensibine dayanır (Doucet ve Johansen). Bu adımlar şu şekilde gerçekleşir:

(1) *Seçme adımı:* S_{t-1} den N tane rastgele $\hat{s}_t^{(n)}$ örnek seçilir. Seçme işlemi yapılırken $r \in [0,1]$ olmak kaydıyla rastgele bir değişken üretilir ve $c_{t-1}^{(j)} > r$ ve $\hat{s}_t^{(n)} = s_{t-1}^{(j)}$ şartlarını sağlayan en küçük j indeksine sahip örnekler seçilir. Bu adımdaki gerçekleşen operasyon Şekil 2'de resimlenmiştir.

(2) *Tahmin adımı:* Seçilen her bir $\hat{s}_t^{(n)}$ örneği için $s_t^{(n)} = f\left(\hat{s}_t^{(n)}, W_t^{(n)}\right)$ ile yeni bir örnek üretilir. Burada $W_t^{(n)}$ sıfır ortalamaya sahip bir Gaussian hatasıdır ve f negatif olmayan bir fonksiyondur ($f(s) = s$).

(3) *Düzeltme adımı:* $s_t^{(n)}$ örneklerine ait $\pi_t^{(n)}$ ağırlıkları z_t ölçümleri kullanılarak hesaplanır. Bunun için $\pi_t^{(n)} = p\left(z_t \middle| x_t = s_t^{(n)}\right)$ eşitliği kullanılır. Burada $p(.)$, Gaussian dağılımı ile modellenebilir.

Elde edilen yeni S_t örnekleri $\varepsilon_t = \sum_{n=1}^{N} \pi_t^{(n)} f\left(s_t^{(n)}, W\right)$ eşitliğinde kullanılarak yeni nesne pozisyonları hesaplanabilir. Parçacık filtre tabanlı nesne takip edicilerin başlangıç değerleri, örnekleme dizisini kullanan sistemlerin eğitimiyle veya ilk ölçümler ($s_0^{(n)} \sim X_0$) kullanılarak gerçekleşir. Sistem ilk ölçümler kullanılarak başlatılırsa, her bir örneğin ağırlık değeri $\pi_0^{(n)} = 1/N$ şeklinde eşit olarak dağıtılır. Ayrıca en iyi parçacık örneklerinin korunması için düşük ağırlıklı olanların elenmesi gerekir. Bunun için ek bir örnekleme algoritmasına ihtiyaç duyulur. Burada dikkat edilmesi gereken nokta, sonrasal yoğunluk fonksiyonunun Gaussian olmak zorunda olmayışıdır.



Şekil 2. Parçacık filtreleme yönteminin seçme adımı

2.2. Çekirdek tabanlı nesne takibi

Çekirdek tabanlı nesne izleyiciler, parametrik olmayan tahmin ediciler gurubu içerisinde yer almaktadır. Parametrik olmayan sistemlerde sabit bir fonksiyon yapısı söz konusu değildir ve bir tahmin gerçekleştirileceği zaman dağılıma ait tüm veri değerleri göz önünde bulundurulur. Bunun yanında parametrik sistemlerde sabit bir fonksiyon yapısı ve sabit parametre değerleri bulunmaktadır (Doucet ve Johansen, 2008).

Çekirdek tabanlı nesne takibi yönteminin amacı, takip edilecek nesneyi basit bir geometrik şekil içerisine alarak şekil içerisinde kalan görünüm bilgisinin olasılık yoğunluk dağılımını elde etmek ve bu dağılımı ardışık video imgeleri boyunca takip edebilmektir (Martinez ve Martinez, 2002). Olasılık yoğunluk dağılımı, takip edilecek nesnenin her pikselinin diğer pikseller üzerindeki etkisini ifade eder (Ushakov, 2001 ve http://school.maths.uwa.edu.au/~duongt/seminars/intro2kde. 03/02/2008.WEB). Bu etki, çekirdek yoğunluk fonksiyonu kullanılarak yumuşatılır.

Çekirdek tabanlı nesne takibi işlemini anlayabilmek için öncelikle bir nesnenin görünümüne ait histogram sunumunu ve böyle bir sununum sahip olduğu dezavantajları bilmek gerekir. Takip edilecek nesneye ait görünüm bilgisinin histogramı oluşturulmak istenildiği zaman öncelikle görünüm veri kümesinin kaç eşit parçaya ("bin" olarak adlandırılır) bölünmesi gerektiği ve bu parçaların başlangıç ve bitiş noktalarının hangi değerler olacağı belirlenmek zorundadır. Bu zorunluluklar histogram sunumunun cazibeliğini azaltır ve bu sınırlamaların olmadığı çekirdek yoğunluk fonksiyonlarının kullanılmasına neden olur (http://school.maths.uwa.edu.au/~duongt/seminars/intro2kde. 03/02/2008.WEB).

Histogramdaki her bir parçanın başlangıç ve bitiş noktalarına olan bağımlılığını kaldırmak için çekirdek yoğunluk tahmin edicileri her bir veri noktasını merkez kabul ederek veriyi belirlenen çekirdek fonksiyonundan geçirir ve her bir veri noktası için bir yoğunluk değeri elde edilir. Böylelikle çekirdek fonksiyonunun yumuşak veya sertliği yapılan tahmininin yumuşak veya sert olmasına neden olur. Bu sayede histogram sunumu kullanıldığı zaman elde edilen dezavantajlar ortadan kalkmış olur (http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/ AV04 05/MISHRA/kde.html. 03/02/2008).

Tablo 2'de literatürde sıklıkla kullanılan çekirdek fonksiyonlarının bir listesi verilmiştir. Bu çizelgede kullanılan $K(.)$ çekirdek fonksiyonunun u giriş parametresi denklem (9)'de ifade edildiği gibi hesaplanır.

Tablo 2. Çekirdek fonksiyonları

Çekirdek Fonksiyon Adı	Çekirdek Fonksiyonu $\left(\boldsymbol{K}(\boldsymbol{u})\right)$
Uniform	$(1/2) * I(\lvert u\rvert < 1)$
Triangle	$(1 - \lvert u\rvert)I(\lvert u\rvert < 1)$
Epanechnikov	$(3/4) * (1 - u^2)I(\lvert u\rvert < 1)$
Quartic	$(15/16) * (1 - u^2)^2 I(\lvert u\rvert < 1)$
Triweight	$(35/32) * (1 - u^2)^3 I(\lvert u\rvert < 1)$
Gaussian	$(1/\sqrt{2\pi}) * exp(-0.5u^2)$
Cosinus	$(\pi/4) * cos(u\pi/2)I(\lvert u\rvert < 1)$

$$u = \frac{x-\mu}{\sigma} \quad [9]$$

Formüldeki μ ve σ simgeleri ile $K(.)$ çekirdek fonksiyonunun merkez değerini ve bant genişliğini ifade etmektedir. Literatürde en sık kullanılan çekirdek fonksiyonu Gaussian olarak bilinir (http://homepages.inf.ed.ac.uk/rbf/CVonline/ LOCAL_COPIES/AV0405/MISHRA/kde.html) . Buna göre, sıfır merkezi değerine ve değişik bant genişliklerine sahip Gaussian çekirdek fonksiyonları Şekil 3'de resimlenmiştir.

Çekirdek tabanlı nesne takibi işlemi tek boyutlu ve çok boyutlu veriler üzerinde gerçekleştirilebilir. Buna göre, takip edilecek nesnenin görünüm bilgisi denklem (10)'da ifade edildiği gibidir.

$$X_i = [X_{i1}, X_{i2}, \dots, X_{id}]^T, \quad i = 1, \dots, n \quad [10]$$

2.2.1. Tek boyutlu çekirdek yoğunluk tahmini

Tek boyutlu X_i veri kümesinin olasılık yoğunluk dağılımını hesaplanmak için Denklem 3'de belirtilen formül kullanılır. Buna göre, x veri elemanının X dizisi üzerindeki katkısı (olasılık yoğunluk değeri veya yoğunluk tahmini) şu şekilde hesaplanır:

$$f(x) = \frac{1}{n}\sum_{i=1}^{n} K\left(\frac{x-X_i}{h}\right) \quad [11]$$

Bu formülde $K(.)$ çekirdek fonksiyonu x merkezi noktasına ve h bant genişliğine sahiptir. Burada $\int K(t)dt = 1$ eşitliğinin sağlanma zorunluluğu vardır.

2.2.2. Çok boyutlu çekirdek yoğunluk tahmini

Çok boyutlu bir veri kümesinin olasılık yoğunluk dağılımı Denklem (4) kullanılarak hesaplanır.

$$\hat{F}_h(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h^d}K\left(\frac{x-X_i}{h}\right) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h^d}K\left(\frac{x_1-X_{1d}}{h},\dots,\frac{x_d-X_{id}}{h}\right) \quad [12]$$

Burada kullanılan bant genişlikleri $h = (h_1,\dots,h_d)^T$ açık bir şeklinde yazılabilir. Buna göre denklem (12)'nin güncel hali denklem (13) ile ifade edilebilir.

$$\hat{f}_h(x) = \frac{1}{n}\sum_{i=1}^{n}\frac{1}{h_1\dots h_d}K\left(\frac{x_1-X_{i1}}{h_1},\dots,\frac{x_d-X_{id}}{h_d}\right) \quad [13]$$

Uygun bant genişliği değerini elde etmek için sık kullanılan genel bir yöntem denklem (14 ve 15)'de belirtilir. Bu eşitlikler yardımıyla dağılımın AMISE (Asymptotic Mean Integrated Squared Error) değeri minimum seviyeye indirilir. Bu değer takip edilecek nesneye ait görüntü verisinden elde edilir. Bant genişliğinin belirlenmesinde AMISE değerinin kullanılması, veri kümesindeki tüm karakteristik özelliklerin korunması anlamına gelir.

$$h_{optimal} = \arg_{min}\mathrm{AMISE} \quad [14]$$

$$\mathrm{AMISE}(h) = \int_{-\infty}^{\infty} E\left(\hat{f}_h(y) - f_h(y)\right)^2 dy \quad [15]$$

Dağılımın AMISE değerinin hesaplanmasında kullanılan f fonksiyonu bilinmeyen yoğunluğu ifade eder ve $\hat{f}$ simgesiyle de f fonksiyonunun n. örneğine dayalı tahmin bilgisi ifade edilmektedir. Bununla birlikte E simgesi beklenen değeri ifade eder.

2.3. Siluet Takibi

Takip edilmesi istenilen nesneler basit geometrik şekiller ile tanımlanamayan el, baş, omuz gibi karmaşık geometrik şekillere sahip olabilir. Siluet tabanlı yöntemler bu nesneler için doğru bir şekil tanımlayıcısı sağlarlar. Siluet tabanlı nesne takip edicilerinin asıl hedefi, önceki imgeler kullanılarak üretilen nesne modelini güncel imge içerisinde bulmaktır (Raykar ve Duraiswami, 2006). Bu modeller nesnenin renk histogramı, nesne kenarını veya sınır şeklini kullanır. Siluet tabanlı nesne takip edicileri "şekil karşılaştırıcılar" ve "sınır takip ediciler" olarak adlandırılan iki alt kategori altında incelemek mümkündür. Şekil karşılaştırmalı yaklaşım, güncel imge içerisinde nesne siluetini arar. Diğer taraftan sınır izlemeli yaklaşım ise, bazı enerji fonksiyonlarının minimizasyonunu veya durum uzay modellerini kullanarak nesnenin güncel imgedeki yeni yerini belirlemeye çalışır.

2.3.1. Şekil karşılaştırıcılar

Şekil karşılaştırıcı yöntemlerin çalışma prensibi şablon karşılaştırmalı nesne takip yöntemleri ile benzerdir. Bu yaklaşımda, nesne siluetinin güncel imgeden bir sonraki imgeye aktarıldığı varsayılır. Güncel imgedeki nesne silueti ve onunla ilişkili nesne modeli bir sonraki imgede aranır. Arama işlemi, bir önceki imgede elde edilen nesne silueti ve modeli ile güncel imgede yakalanan nesneler arasındaki benzerlik değerleri hesaplanarak gerçekleştirilir. Bu nedenle, silueti net olmayan (katı olmayan) nesneler doğru ve istikrarlı bir şekilde takip edilemez. Genellikle nesne modeli için kenar haritası kullanılır. Nesne modelinin her imgede nesneyi tam ifade edebilmesi için görünüş değişikliklerini göz önüne alarak yeniden hesaplanması gerekir. Bu güncelleme işlemi, özellikle katı olmayan nesne hareketlerindeki ışık değişimi ve bakış noktası gibi nesne takip problemlerinin üstesinden gelebilmek için gereklidir (Munder vd., 2008).

2.3.2. Sınır takibi

Şekil karşılaştırma yöntemlerinin aksine sınır takibi yöntemleri güncel imgedeki sınır çizgilerini başlangıç kabul ederek bir sonraki imgede yeni nesne sınırlarını arar. Sınır takibi algoritmalarının uygun sonuç vermesi için güncel imgedeki nesne ile bir sonraki imgede bulunan nesne alanlarının kesişimi olmak zorundadır. Sınır değerlendirme algoritmaları iki alt kategoride incelenebilir. İlk yaklaşım, sınır biçim ve hareketini modellemek için durum uzay modellerini kullanır. İkinci yaklaşım, gradient azalma gibi minimizasyon tekniklerini kullanarak sınır enerjisini minimize eder (Cordea vd., 2008).

Durum uzay modelli kullanarak nesne takibi

Nesneye ait durum vektörü, nesne sınırının biçim ve hareketi şeklinde tanımlanır. Nesne sınırına ait sonrasal olasılık yoğunluk değerinin maksimum olması için sınıra ait durum vektörü her bir imgede güncellenir. Sonrasal yoğunluk, öncesel yoğunluğa ve güncel olabilirliğe bağlıdır. Güncel olabilirlik değeri, nesne sınırları ile gözlenen kenar değerleri arasındaki mesafe kullanılarak hesaplanabilir (Bertalmio vd., 2000).

Sınır enerji fonksiyonunun direk minimizasyonu yöntemiyle nesne takibi

Sınır değerlendirme kontekstinde, bu bölümde işlenen sınır takip ediciler ile bir önceki bölümde incelenen bölütleme yöntemleri arasında bir paralellik bulunmaktadır. Nesne bölütleme ve nesne takip yöntemleri sınır enerji fonksiyonunu greedy veya gradient azaltma yöntemlerini kullanarak minimize eder. Sınır enerjisi iki farklı yöntem ile tanımlanabilir (Lee, 2007): 1) geçici gradient(optik akış) yöntemi; 2) nesne ve arka plan alanından üretilen görünüm istatistikleri yöntemi. Geçici görüntü gradientini kullanarak sınır takip yöntemi, optik akış hesaplaması üzerine yapılan kapsamlı bir çalışmayla ile motive edilir. Optik akış sınırlamaları denklem (16) ile belirtilen sabit parlaklık sınırlaması kullanılarak üretilir:

$$I^{t+1}(x,y) - I^{t}(x-u, y-v) = 0 \quad [16]$$

Burada I görüntü, t zaman, (u,v) ise x ve y yönündeki optik akış vektörleridir. Optik akış yöntemine alternatif olarak, bir imgeden diğerine geçerken nesnenin iç ve dış alanları hakkında hesaplanan istatistiksel tutarlılık değeri kullanılarak nesne takibi gerçekleştirilebilir. Bu yaklaşım sınırın güncel imgede başlangıç değerlerinin belirlenmesi gerekliliğine ihtiyaç duyar.

3. NESNE TAKİP YÖNTEMLERİNİN KARŞILAŞTIRILMASI

Nesne takip yöntemleri hakkında önerilen yaklaşımların karşılaştırmalı mukayese edilmesi yaklaşımlar hakkında hızlı bir şekilde bilgi edinilmesini sağlayabilir. Bu nedenle bu bölümde aynı gurup içerisinde yer alan nesne takip yöntemlerinin belirli

Tablo 3. Karşılaştırmada kullanılan kıstaslar

No	Kıstasın Tanımı
1	Aynı anda birden fazla nesne takibi gerçekleştirilebiliyor mu?
2	Dinamik arka plan modeline sahip mi?
3	Görüntü ekranına nesne giriş çıkışına izin veriliyor mu?
4	Eğitim aşamasına ihtiyacı ortadan kaldırılmış mı?
5	Başlangıç aşamasında nesne pozisyonlarının belirlenme ihtiyacı ortadan kaldırılmış mı?
6	Kapatma olayı gerçekleştiği zaman nesne takibi istenilen şekilde devam edebiliyor mu?
7	Ardışık imgeler arasındaki nesne bağı problemi için uygun bir çözüm sağlanabiliyor mu?
8	Gürültü, ışık miktarının değişimi, karmaşık nesne şekli ve hareketi gibi durumlarda takip edici nesneyi takip edebiliyormu?
9	Gerçek zamanlı uygulama gereksinimleri karşılanabiliyor mu?

kıstaslar kullanılarak karşılaştırılması yapılmıştır. Nitelikli bir karşılaştırmanın yapılabilmesi için önerilen yöntemler arasındaki farkları net bir şekilde ortaya çıkarabilen (ayırt edici özelliklere sahip) kıstasların belirlenmesi gerekir.

Önerilecek yeni bir nesne takip yönteminin mevcut yöntemlere karşı üstünlük ve eksikliklerinin daha net anlaşılabilmesi amacıyla Tablo 3'de gösterilen birkaç kıstas belirlenmiştir. Bu kıstasların belirlenmesinde Tablo 1'deki uygulamalardan yararlanılmıştır. Her bir uygulamanın bir diğerinden üstünlüğü, avantaj ve dezavantajı Tablo 3'deki kıstasların belirlenmesinde önemli bir etken olmuştur.

Tablo 3'deki 1 nolu kıstas ile tek nesne takip ediciler ile çoklu nesne takip ediciler bir birinden ayrılmaktadır. 2 nolu kıstas ile takip esnasında arka plan bilgisinin sabit olarakmı yoksa dinamik olarakmı belirlendiği anlaşılmaktadır. 3 nolu kıstas ile takip edilen nesne sayısının sınırlımı yoksa bu noktada herhangi bir sınırlamanın olmadığı anlaşılmaktdır. 4 nolu kıstas ile takip edilecek nesnenin renksel ve şekilsel bilgisinin öğrenilmesi amacıyla bir eğitim aşamasına ihtiyaç duyulup duyulmadığı anlaşılmaktadır. Bu aşamada yapay sinir ağları gibi teknikler kullanılacağı gibi aktif şekil modelleri gibi yöntemlerde kullanılabilir. 5 nolu kıstas yardımıyla takip edicinin nesneyi otomatik algılama özelliğinin olup olmadığı belirlenir. 6 nolu kıstas, nesne görünümlerinin engellenmesi problemi olarak bilinen "kapatma" problemine karşılık takip edicinin yeteneğini belirler. 7 nolu kıstas, nesne bağı probleminin takip edici tarafından nasıl çözüldüğünü ortaya çıkarır. 8 nolu kıstas, klasik takip problemleri olarak bilinen ışık miktarının değişimi, gürültü gibi problemlere karşı takip edicinin davranışının belirlenmesini sağlar. 9 nolu kıstas ile gerçek zamanlı uygulamaların "hız" ve "maliyet" gibi gereksinimlerine karşı takip edicinin yanıtının ne olduğunun anlaşılmasını sağlar.

Tablo 4'de, literatürde yer alan nokta tabanlı, çekirdek tabanlı ve siluet tabanlı nesne takip yöntemlerinin belirlenen kıstaslara verdikleri yanıtlar gösterilmektedir. Verilen yanıtlar olumlu veya olumsuz şeklinde ifade edileek her bir yöntemin belirlenen 9 kıstas üzerindeki yanıtı listelenmiştir.

Mevcut takip yöntemleri ile ilgili Tablo 4'de verilen karşılaştırma sonuçlarının incelenmesi, nesne takip algoritmaları ile ilgili kullanıcıya hangi ortam ve şartlarda hangi algoritmayı kullanması gerektiği hakkında bilgi sağlayabilir. Böylelikle ortama ve şartlara bağlı olarak en uygun nesne takip algoritmasının seçimi gerçekleştirilerek takip performansının arttırılması sağlanabilir.

Tablo 4. Nesne takip yöntemlerinin karşılaştırma sonuçları. İlgili algoritmanın kıstaslara verdiği yanıtlar, √ (olumlu) ve × (olumsuz) sembolleri ile ifade edilmektedir.

Algoritma	**KISTASLAR**								
Nokta tabanlı	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**
Zhao vd. 2008	√	×	×	√	√	√	√	√	×
Haritaoglu vd. 2000	√	×	√	×	√	√	√	√	√
Xue J. vd. 2008	√	√	√	×	√	√	√	√	√
Dong ve DeSouza, 2009	×	√	×	√	×	×	√	√	×
Veenman vd. 2001	×	×	×	×	×	×	√	√	√
Çekirdek tabanlı									
Yilmaz, 2007	×	√	×	√	×	×	√	√	√
Comaniu, 2003	×	×	×	√	×	×	√	√	√
Fan vd., 2005	√	√	×	√	×	×	√	√	√
Siluet tabanlı									
Cootes vd., 1998	√	√	×	×	√	×	√	√	√
Nascimento ve Marques, 2007	×	√	×	×	×	×	√	√	√
Zhu vd., 2007	×	×	×	√	×	×	√	√	√

4. SONUÇ

Bu çalışmada, video imgelerindeki hareketli nesnelerin takibini gerçekleştirebilen modern nesne takip yöntemleri hakkında detaylı bir araştırma yapılmış ve elde edilen bilgiler ışığında önerilen takip yöntemleri yapısal olarak sınıflandırılmıştır.

Modern nesne takip algoritmalarının ayırt edici karakteristik özelliklerini belirleyebilmek amacıyla dokuz farklı kıstas belirlenmiş ve her bir kıstasa takip yöntemlerinin verdiği yanıtlar göz önüne alınarak sınıflandırma yapılmıştır. Elde edilen sonuçlar tablolar halinde listelenmiştir.
Böylelikle okuyucuya, ortama ve şartlara bağlı olarak değişebilen nesne takip ihtiyacına en uygun yanıtı verebilecek olan nesne takip algoritmasını rahatlıkla seçebilme imkânı sunulmuştur.

KAYNAKLAR

Berbeco, R. I., Mostafavi, H., Sharp, G. C. and Jiang, S. B. 2005. "Towards fluoroscopic respiratory gating for lung tumours without radiopaque markers", Phys. Med. Biol. 50 4481–90.

Bertalmio, M., Sapiro, G., and Randall, G., 2000, "Morphing active contours". IEEE Trans. Patt. Analy. Mach.Intell. 22, 7, 733–737.

Bettencourt, K., Somers, D. C., 2007, "Effects of task difficulty on multiple object tracking performance", Journal of Vison, 7(9):898, 898a.

Chang, W. Y., Chen, C.S., Jian, Y.D., 2008, "Visual Tracking in High-Dimensional State Space by Appearance-Guided Particle Filtering", 17(7):1154-67.

Comaniciu, D., and Meer, P. 2002. Mean shift: A robust approach toward feature space analysis. IEEE Trans. Patt. Analy. Mach. Intell. 24, 5, 603-619.

Comaniciu, D., Ramesh, V., and Meer, P. 2003. Kernel based object tracking. IEEE Trans. Patt. Analy. Mach. Intell. 25, 564-575.

Cootes, T. F., Edwards, G. J., and Taylor, C. J. 1998. Active Appearance Model. Proc European Conference on Computer Vision, Vol. 2, pp. 484-498, Springer.

Cootes, T. F., Taylor, C. J., Cooper, D. H., and Graham, J. 1995. Active Shape Models-Their training and application, Computer vision and image understanding, Vol. 61. No:1, pp.38-59.

Cordea, M. D., Petriu, E. M., Petriu, D. C., 2008, "Three-Dimensional Head Tracking and Facial Expression Recovery Using an Anthropometric Muscle-Based Active Appearance Model", IEEE Transactions on Instrumentation and Measurement, 57(8): 1578 – 1588.

Dong, Y.; DeSouza, G.N.; 2009, "A new hierarchical particle filtering for markerless human motion capture", Computational Intelligence for Visual Intelligence, IEEE Workshop on, Page(s):14 – 21.

Doucet, A.; Johansen, A.M., 2008, "A tutorial on particle filtering and smoothing: fifteen years later". Technical report, Department of Statistics, University of British Columbia.

Evans, M.; Hastings, N.; and Peacock, B. "Probability Density Function and Probability Function." §2.4 in Statistical Distributions, 3rd ed. New York: Wiley, pp. 9-11, 2000.

Fan, Z. , Wu, Y., Yang, M., 2005, "Multiple Collaborative Kernel Tracking", Computer Vision and Pattern Recognition, Vol 2, Pages: 502- 509.

Fieguth, P. and Terzofoulos, D. 1997. Color-based tracking of heads and other mobile objects at video framerates. In IEEE Conference on Computer Vison and Pattern Recognition. 21-27.

Forsyth, D. A., Arikan, O., Ikemoto, L., O'Brien, J., and Ramanan, D., 2005, "Computational Studies of Human Motion:Part 1, Tracking and Motion Synthesis", Computer Graphics and Vision, 1(2/3):77–254.

Haritaoglu, I., Harwood, D., and Davis, L. 2000. W4: Real-time surveillance of people and their activities. IEEE Trans. Patt. Analy. Mach. Intell. 22, 8, pp.809–830.

http://en.wikipedia.org/wiki/Video_tracking. 10/10/2009.

http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0405/MISHRA/kde.html. 03/02/2008.

http://school.maths.uwa.edu.au/~duongt/seminars/intro2kde. 03/02/2008.

Lee, S., Kang, J., Shin, J., Paik, J., 2007, "Hierarchical active shape model with motion prediction for real-time tracking of non-rigid objects"

Leven, W. F., Lanterman, A.D., 2009, "Unscented Kalman Filters for Multiple Target Tracking With Symmetric Measurement Equations", Proceedings of SPIE, the International Society for Optical Engineering, 5810:56-67.

Martinez, W. L., Martinez, A. R., 2002, "Computational Statistics Handbook with MATLAB", Chapman & Hall/CRC. Chapter 8.

Munder, S.; Schnorr, C.; Gavrila, D.M., 2008, "Pedestrian Detection and Tracking Using a Mixture of View-Based Shape–Texture Models", IEEE Transactions on Intelligent Transportation Systems, 9(2): 333 – 343.

Nascimento, J.C.; Marques, J.S., 2007, "Robust Shape Tracking With Multiple Models in Ultrasound Images", IEEE Transactions on Image Processing, 17(3): 392 – 406.

Oh, S., Russell, S., and Sastry, S. S., 2009, Markov chain Monte Carlo data association for multi-target tracking. IEEE Transactions on Automatic Control, 54(3), 481–497.

Raykar, V. C., Duraiswami, R. 2006, "Fast optimal bandwidth selection for kernel density estimation", Proceedings of the sixth SIAM International Conference on Data Mining, pp. 524-528.

Salari, V. and Sethi, I. K., 1990, "Feature point correspondence in the presence of occlusion", IEEE Trans.Patt. Analy. Mach. Intell. 12(1):87–91.

Sheskin, D.J., 2003, "Handbook of Parametric and Nonparametric Statistical Procedures", Chapman & Hall/CRC.

Smal, K., Draegestein, N. Galjart, W. Niessen, E. Meijering, 2008, "Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images: Application to Microtubule Growth Analysis", IEEE Transactions on Medical Imaging, 27(6):789-804.

Ushakov, N.G., 2001, "Density of a probability distribution", in Hazewinkel, Michiel, Encyclopaedia of Mathematics, Kluwer Academic Publishers, ISBN 978-1556080104.

Veenman, C., Reinders, M., and Backer, E., 2001, "Resolving motion correspondence for densely moving points", IEEE Trans. Patt. Analy. Mach. Intell. 23(1):54–72.

Xue J., Zheng N., Geng J., Zhong X., 2008, "Tracking multiple visual targets via particle-based belief propagation", IEEE Trans Syst Man Cybern B Cybern. 38(1):196-209.

Yilmaz, A., 2007, "Object Tracking by Asymmetric Kernel Mean Shift with Automatic Scale and Orientation Selection", Computer Vision and Pattern Recognition, Page(s):1 – 6.

Yilmaz, A., Javed, O., and Shah, M., 2006, "Object Tracking: A Survey", ACM Comput. Surv., 38(4):13.

Zhao, T., Nevatia, R., Wu, B, 2008, "Segmentation and Tracking of Multiple Humans in Crowded Environments". IEEE Trans. on pattern analysis and machine intelligence. Vol:30, No:7.

Zhu, G., Zhang, S., Chen, X., Wang, C., 2007, "Efficient Illumination Insensitive Object Tracking by Normalized Gradient Matching", IEEE Signal Processing Letters, 14(12): 944 – 947.