

Video dizilerindeki araç plakalarının FE-Yoğunlaştırma algoritması kullanılarak izlenmesi

İlhan Kubilay YALÇIN*, **Muhittin GÖKMEN**

İTÜ Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Programı, 34469, Ayazağa, İstanbul

Özet

Bu çalışmanın amacı, araç plakalarının üç boyutlu uzayda konum ve yöneliminin bulunması için video görüntüsünden izlenmesidir. Eğer nesnenin altı dereceli uzay serbestliği belirlenmek isteniyorsa, durum uzayı altı boyutlu olur. Her serbestlik derecesi için olası değerler kümesini 100 elemanlı kabul edersek, bu nesneyi her olası durumu deneyerek izleyebilmek için görüntü verisi üzerinde 100^6 karşılaştırma yapmamız gerekmektedir. Bu sınırlı çözünürlük ve altı dereceli serbestlik uzayında dahi, bu şekilde gerçek zamanlı izleme yapmanın mümkün olmadığı açıktır. Stokastik izlemenin ardında yatan düşünce, her olası nesne durumunu denemek yerine, durum hakkında tahminlerde bulunmak ve bu tahminleri o anki video karesi ile karşılaştırarak sonuçları bir sonraki video karesi için tahmin yapmakta kullanmaktır. Son yıllarda, bilgisayar ile görüntü işleme problemlerinde Parçacık Filtreleri'nin kullanımına yönelik bir ilgi görülmektedir. Bilgisayar ile görüntü işleme problemlerinde kullanılan özel Parçacık Filtresi'ne Yoğunlaştırma algoritması veya Ardışıl Önem Örnekleme denmektedir. Bu yöntem hareketli nesnelere için gürbüz bir izleme olanağı sunmaktadır. Öte yandan, bu algoritmanın yakınsaması büyük oranda parçacık sayısı ve dinamik modelin doğruluğu arasındaki ilişkiye bağlıdır. Bu tezde Yoğunlaştırma algoritmasını iyileştirmek amacıyla FE-Yoğunlaştırma algoritması önerilmektedir. Bu algoritma Farksal Evrim ve Yoğunlaştırma algoritmalarının bir birleşimidir. FE-Yoğunlaştırma algoritması üç boyutlu uzayda tek bir kamerayla araç plakası konum ve yöneliminin izlenmesi için kullanıldı. Genişletilmiş Kalman filtresi, Yoğunlaştırma, Genetik Yoğunlaştırma ve FE-Yoğunlaştırma algoritmalarının izleme başarımları karşılaştırıldı. FE-Yoğunlaştırma algoritması diğer üç algoritmaya göre çok daha iyi başarımlar göstermektedir.

Anahtar Kelimeler: Araç plakası izlenmesi, yoğunlaştırma, FE-Yoğunlaştırma.

*Yazışmaların yapılacağı yazar: İlhan Kubilay YALÇIN. ilhan.yalcin@bte.mam.gov.tr; Tel: (262) 677 25 43.

Bu makale, birinci yazar tarafından İTÜ Fen Bilimleri Enstitüsü, Bilgisayar Mühendisliği Programında tamamlanmış olan "3D Model Based Stochastic Tracking of License Plates in Video Sequences" adlı doktora tezinden hazırlanmıştır. Makale metni 01.06.2007 tarihinde dergiye ulaştı, 06.09.2007 tarihinde basım kararı alınmıştır. Makale ile ilgili tartışmalar 31.04.2009 tarihine kadar dergiye gönderilmelidir.

Tracking of license plates in video sequences using DE-Condensation algorithm

Extended abstract

Automated vehicle identification (AVI) is still an important research issue and drawing attention in machine vision community. Its potential commercial applications are automatic barrier systems, automatic payment of parking or highway toll fee, automatic locating of a stolen vehicle, automatic calculation of traffic volume and so on. License plate enables us to identify a vehicle and its owner. License plate recognition is the most effective method for identification of the vehicle. A suitable and promising solution to vehicle identification is visual recognition of the license plate from camera view. This approach is applicable because it does not require vehicles to carry additional equipment such as special RF transmitters. Without additional cost, these systems are capable of installation to the field.

But visual license plate detection and recognition is a very difficult task. It is quite a challenging problem because vehicles are running in an outdoor environment, where lighting conditions can change rapidly, weather conditions can cause poor image quality, license plates can be dirty or in poor condition and occlusions can occur frequently. Therefore, Visual License Plate Recognition (VLPR) systems may fail because of uncontrollable external conditions. Beside the challenging nature of the problem, the high-dimensional nature of the VLPR problem may impose a significant computational load on the target processing platform.

The aim of this work is 3D tracking of license plate in order to determine the state (spatial position and 3D orientation) of the license plate from sequential frames of the video. This can be accomplished in a brute force way by testing every possible orientation and translation and then selecting the one that best fits the current frame. If all six degrees of spatial freedom of the object are to be determined, the state space of the object is six dimensional. Setting the number of possible values of each degree of freedom to 100, the task of tracking by brute force then requires 100^6 comparisons of a state with the image data. Even with such a limited resolution and a six dimensional feature space it is clear that, it is com-

putationally impossible to perform tracking in real time by brute force. That is where stochastic tracking is meaningful. A stochastic process is one whose behaviour is non-deterministic in that the next state of the environment is partially but not fully determined by the previous state of the environment. Instead of comparing every possible configuration of the object with each video frame, the idea behind stochastic tracking is to make a set of guesses of the state, compare these guesses with the current frame, and use the result of this comparison as the basis for a new set of guesses when the next frame comes. The new guesses are made by selecting the best guesses from the last frame and applying a model of the movement of the object from one frame to the next. The set of guesses (called particles or samples) will frame by frame converge around the correct state of the object.

In recent years, there has been a great interest in applying Particle Filtering to computer vision problems. This specialized Particle Filtering method for computer vision problems is introduced as Condensation or Sequential Importance Sampling. Condensation algorithm utilizes factored sampling and given dynamic models to propagate an entire probability distribution for object's position and shape over time. It can perform successfully robust tracking of object motion. On the other hand, its convergence greatly depends on the trade off between the number of particles/hypotheses and the fitness of the dynamic model. For example, in cases where the dynamics are complex or poorly modelled, thousands of samples are usually required for real applications.

In order to improve the performance of the Condensation algorithm, DE-Condensation algorithm is proposed, which is an integration of the Differential Evolution and Condensation algorithms. DE-Condensation algorithm is utilized for spatial position estimation and tracking of license plates in 3D from monocular camera view. The performance and computational load of the Extended Kalman filter, Condensation Algorithm, DE-Condensation algorithm and Genetic Condensation algorithm are compared for evaluating DE-Condensation Algorithm's performance.

Keywords: License Plate Tracking, Condensation, DE-Condensation.

Giriş

Görsel Araç Plakası Tanıma (GAPT) sistemleri temel olarak üç kısımdan oluşmaktadır. Bunlar, plakanın tespiti, plakanın izlenmesi ve plakanın tanınması (okunması) kısımlarıdır. Bu üç bölümden belki de en önemlileri tespit ve izleme kısımlarıdır. Araç plakası tespiti araç plakası görüntüsünün bütün görüntü üzerinde aranmasını içerir. Bu amaçla birçok yöntem geliştirilmiştir. Bu yöntemler arasında plaka karakterlerinin yol açtığı dikey kenar yoğunluğu, kenar çıkartımı ve plaka renginin işlenmesi gibi yaklaşımlar mevcuttur. Bütün bu yöntemler sözü edilen özellikleri bulmak amacıyla bir arama yöntemini de öngörmektedirler. Bu yöntemler arasında Genetik algoritmalar, Ortalama Kayma Algoritması, Kayan Pencere ve Vektör Kuantalama gibi birçok yöntem mevcuttur. GAPT sistemleri içerisinde bu çalışma daha çok plakanın tespiti ve özellikle plakanın izlenmesine odaklanmıştır.

Metin konumlandırma tabanlı tekniklerin dayandığı ana gerçek, plaka üzerindeki karakterlerin hızlı parlaklık değişikliğine yol açtıklarıdır. Kenar özellikleri çıkartılmasından sonra bu kenarlar morfolojik işlemlere tabi tutulur ve plaka bölgeleri oluşturulur. Fakat karmaşık sahnelerde kapama, açma gibi morfolojik teknikler gürültülü çıktı ve çok sayıda aday bölge üretmektedir. Kenar veya plaka sınırlarının tespiti kenar bölgesinin tam olarak bulunamaması, çeşitli plaka tiplerinin olması ve plaka kenarlarının parçalı olması dolayısıyla iyi sonuçlar vermeyebilmektedir. Sonuçta plaka bölgesi bulma işlemi çok sayıda aday bölge vermekte ve bunların içinde doğru bölgeleri seçmek zor olmaktadır. Morfolojik işlemlerin performansını arttırmak için Dubey'de (2005) yeni bir morfolojik işlem yaklaşımı sunulmaktadır. Bu yöntem daha çok birleşik bölge vermekte ve bunu belli ön şartlara uyarak başarmaktadır.

Plaka bölgelerinin çıkartımı ve en uygun bölgenin bulunmasına yönelik diğer çalışmalar, histogram benzerliği kullanılması (Hsi-Jian vd., 2004), wavelet dönüşümünün kullanılması (Ching-Tang vd., 2005), Ortalama Kayma Algoritmasının kullanılması (Wenjing vd., 2005), genetik arama yönteminin kullanılması (Xiong

vd., 2004) ve renk tabanlı yaklaşımlar (Sang Kyoon vd., 1996; Zayed vd., 2004) olarak örneklendirilebilir.

Plaka tespit algoritmaları bütün görüntüyü işlemek durumundadır. Böyle bir durum hesaplama yükü bakımında bu algoritmaların kötü olduğunu göstermektedir. Hâlbuki plakanın izlenmesi bize ilerleyen video karelerinde olası plaka bölgelerinin kapsamını daraltmakta ve görüntü üzerinde (İlgili Olunan Alan) İOA tanımının yapılabilmesine olanak tanımaktadır. Böylece plaka tespiti yalnızca tanımlanan İOA üzerinde yapılabilmektedir. Plaka izlenmesi temel olarak plakanın yerinin bir sonraki video karesinde tahmin edilmesi amacıyla yapılmaktadır.

Çoğu izleme sistemi kenar bilgisinin veya belli özelliklerin izlenmesini amaçlamaktadır. Çünkü bu özellikler doğal olarak görüntüde mevcuttur. Bu özellikler hem avantajlar hem de dezavantajlar içermektedir. Belli bir nesnenin izlenmesi için birçok yöntem önerilmiştir, bu yöntemler özellik tabanlı yöntemler ve model tabanlı yöntemler olarak ikiye ayrılabilir. Özellik tabanlı yöntemler nesnenin görüntüden çıkartılan belli özelliklerinin takibini amaçlamaktadır. Model tabanlı yöntemler ise 2B ve 3B modeller kullanarak nesneyi izlemeyi amaçlar. Bu çalışmada 3B yöntemleri ele alacağız. Özellik noktalarının eşlenmesi 3B nesne modelinin kontrol noktalarının kenar bilgisine uydurulmasıyla gerçekleştirilebilir. Bu kontrol noktaları görüntüyü bölütleme için değil eşleme için kullanılmaktadır. Eşleme işlemi 4 doğrultuda arama yaparak basit ve etkili olabilir.

Özellik noktaları, üzerinde belirgin desen bulunan nesnelere için etkilidir ve geometrik bozulma ve aydınlatma değişimlerine karşı gürbüzdür. Fakat belirgin bir desen taşımayan nesnelere için bu yöntemler iyi sonuç vermemektedir. Kenar bilgisine dayalı izleme, keskin kenar bilgisi olan ve güçlü parlaklık değişimi gösteren görüntüler için daha elverişlidir. Fakat karmaşık ve desenli nesnelere için kötü sonuç vermektedir. Gerçek hayatta ne yazık ki nesnelere desenli olması veya olmaması gibi ayrımlar yapılamayacağı için bu iki yöntem birbirine karşılık değil yan

yana bütünleyici olarak kullanılmalıdır. Desen özellik noktaları civarlarındaki desen özellikleri ile belirlenebilirken, bunu kenar bilgisine dayalı özellik noktaları için söylemek o kadar kolay değildir. Bu noktaları belirlerken birden fazla olasılığı göz önünde bulundurmamak gerekebilir.

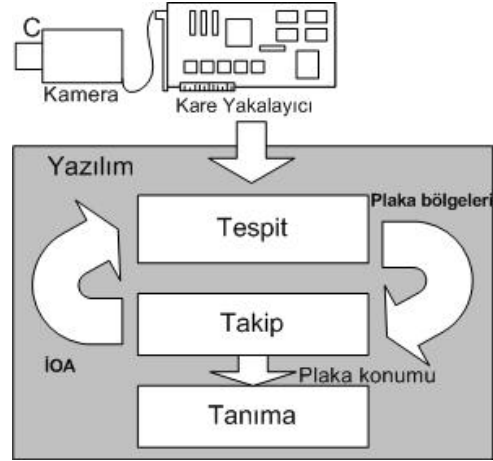
Model tabanlı yaklaşım 3B modelin en fazla görüntü ile benzerlik gösterdiği durumu bulmayı hedef alır. Bu problem sonuç olarak bir en iyileme (optimizasyon) problemi niteliği taşır ve sistem durumunun tahminini olasılıksal olarak yapan bir yöntem olarak incelenebilir. Ne yazık ki görüntü gibi çok boyutlu durum uzaylarında ve doğrusal olmayan bir sistem için sistem durumunun tahmini kolay değildir. Bu tür bir tahmin için gerekli algoritmalar, Extended Kalman Filtresi, Unscented Kalman Filtresi ve Ardışıl Monte Carlo yöntemleri olarak sıralanabilir. Ardışıl Monte Carlo yöntemlerinin en popüler olanı günümüzde Parçacık Filtresi'dir (Particle Filter). 3B model tabanlı izleme yaklaşımının uygulamalarını (Vacchetti vd., 2004; Ponsa vd., 2005; Wen-Yan vd., 2005) bir çok çalışmada görmek mümkündür.

GAPT sistemi mimarisi

GAPT sistemleri tek bir video karesiyle çalışabilecekleri gibi, birden fazla kare ile de çalışabilirler. Birden fazla video karesi kullanımının performans artımına yol açacağı kesindir. Önceki bölümde ifade edildiği gibi, bir GAPT sistemi üç kısımdan oluşmaktadır. Bunlar sırasıyla, plaka tespiti, plaka izlenmesi ve plaka tanıma kısımlarıdır. Bu kısımlar arasındaki veri akışı Şekil 1'de gösterilmiştir. Bu diyagram temel olarak tespit ve izleme arasındaki ortak çalışmayı göstermektedir. İzleme işlemi tespit işlemine bir sonraki video karesinde ele alınacak İOA tanımını sağlamaktadır. Bir sonraki video karesinde plaka bölgeleri yalnızca İOA içerisinde aranmaktadır.

Kamera modeli

Burada verilen uygulamada iğne deliği kamera (pin hole) modeli kullanılmıştır. Bu model 3B uzayda tanımlı bir noktanın perspektif izdüşümü ile kamera düzleminde karşı geldiği 2B noktayı



Şekil 1. GAPT sistem mimarisi

vermektedir. Bu dönüşüm iç ve dış (intrinsic, extrinsic) parametre setleri olmak üzere iki parametre seti tarafından tanımlanır. İç parametreler fokal uzaklık (f), benek boyutları (k_u, k_v), izdüşümün orta noktası, görüntü koordinatları ($u_0 \ v_0$)^T olmak üzere boyutlandırma, dönme ve kaydırma olarak bir izalama (affine) dönüşümü tanımlamaktadır. Bütün bu dönüşümler 3x3 R dönüş matrisi ve 3x1 T öteleme vektörü tarafından tamamen tanımlanır. Böylece aşağıdaki dönüşüme ulaşılır.

$$\begin{pmatrix} x \\ y \\ s \end{pmatrix} = \begin{pmatrix} \alpha_u & 0 & u_0 & 0 \\ 0 & \alpha_v & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} R & T \\ 0^T & 1 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \quad (1)$$

Burada $(x \ y \ s)^T$ görüntü üzerindeki izdüşümünün homojen koordinatları, $\alpha_u = \frac{f}{k_u}, \alpha_v = \frac{f}{k_v}$, (burada k_u ve k_v benek genişliği ve uzunluğudur). R döndürme matrisi, $(X \ Y \ Z \ 1)^T$ homojen koordinatlarda 3B uzaydaki nokta, T ise öteleme matrisi olup tüm birleştirilmiş matris perspektif izdüşüm matrisi adını alır. Bu matris 3B uzaydaki bir nokta ile 2B görüntü düzlemindeki bir noktayı eşleştirir. Bir dönüşüm matrisi R , üç matrisin bir çarpımı olarak yazılabilir. Bu üç matris sırasıyla X , Y ve Z koordinatları etrafın-

daki dönüşü tanımlar. Bu dönüşümler birden fazla şekilde ifade edilebilmektedir. Örnek olarak, α, β, γ sırasıyla X, Y ve Z eksenleri etrafında Euler dönüş açıları olsun. Bu durumda aşağıdaki denklemler elde edilir,

$$R_x = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{pmatrix},$$

$$R_y = \begin{pmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{pmatrix},$$

$$R_z = \begin{pmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad R = R_x \cdot R_y \cdot R_z$$

$$T = \begin{pmatrix} \Delta x \\ \Delta y \\ \Delta z \end{pmatrix} \quad (2)$$

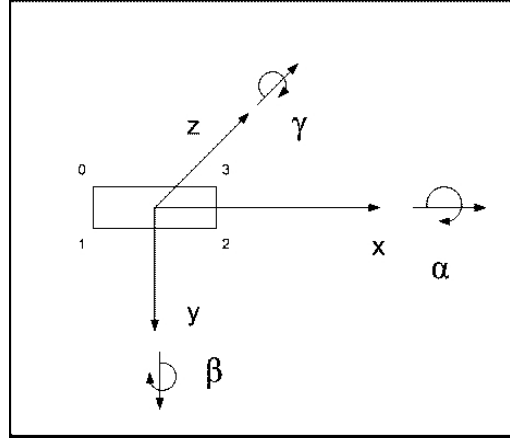
Bu şekilde gerçek dünya koordinatları olan (X Y Z) değerlerinden görüntü üzerindeki (x y s) koordinatlarına dönüşüm elde edilmiş olur. Gerçek benek koordinatları p_x, p_y ise $p_x = x/s$ $p_y = y/s$ bölümleri ile elde edilir.

Sistem durumu

Bir araç plakası nesnesinin 3B uzaydaki durumu altı değişken ile ifade edilebilir. Bunlar sırasıyla X, Y, Z kartezyen koordinatları, Euler açıları, sırasıyla α (X eksen etrafında dönüş miktarı), β (Y eksen etrafında dönüş miktarı) ve γ (Z eksen etrafındaki dönüş miktarı) dır. Böylece bir araç plakası nesnesi için sistem durumu $s_k = [X \ Y \ Z \ \alpha \ \beta \ \gamma]^T$ olarak tanımlanabilir. Durum vektörünün değişkenlerinin ifadesi Şekil 2' de görülebilir.

Sistem modeli

Sistem durumunu tanımladıktan sonra sistemin zaman içindeki değişimini modelleyen ayrık zamanlı doğrusal olmayan dinamik sistemi detaylandırılabilir. Burada yapılan varsayım araç plakasının düzlem normalinde hareket ettiğidir.



Şekil 2. Koordinat sistemi

Buna göre $s_k = f_k(s_{k-1}, w_{k-1})$ denklemi sistem durumunun değişimini modellesin, böyle bir durumda dinamik sistem stokastik fark denklemi ile gösterilirse,

$$s_{k+1} = s_k + \begin{bmatrix} c_k \cdot \sin\beta_k \\ -c_k \cdot \sin\alpha_k \cdot \cos\beta_k \\ c_k \cdot \cos\alpha_k \cdot \cos\gamma_k \\ 0 \\ 0 \\ 0 \end{bmatrix} + N(0, \sigma), \quad s_k = \begin{bmatrix} X_k \\ Y_k \\ Z_k \\ \alpha_k \\ \beta_k \\ \gamma_k \end{bmatrix} \quad (3)$$

denklemi elde edilir. Bu denklemde α, β, γ Euler açıları, c_k ise sabit hız büyüklüğüdür.

Gözlem modeli

Sistemimizde bir de gözlem modeli olmalıdır. Gözlem modeli anlatılan iğne deliği kamera modeli ile belirlenmektedir. Bu model araç plakasının 3B köşe noktalarının kamera düzlemine iz düşürülmesine karşılık gelir. Bu durumda ölçüm değerleri b_k plakanın köşe noktalarının 2B izdüşüm koordinatları olur. $b_k = h_k(s_k, v_k)$ denklemi ile verilen gözlem modeli s_k durumu ile görüntüden elde edilen ölçümün b_k ilişkisini belirler. Eğer b_k ölçümleri doğrudan elimizde olsaydı, s_k durumuna karşılık gelen inovasyon değeri $b_k - h_k(\hat{s}_k)$ olarak bulunabilecekti, fakat uygulamada ölçüm değerlerine doğrudan ulaşmak mümkün değildir. Ölçüm değerleri araç

plakasının köşe noktalarının görüntü işleme teknikleri ile bulunması ile ancak mümkün olabilir. Böyle bir işlem problemin farklı bir bakış açısıyla çözümüne karşılık gelir. Bunun yerine verilen bir ölçüm b_k ve durum s_k değeri için $p(b_k | s_k)$ olasılık değerini elde etmek mümkündür. Uygulamada $b_k - h_k(\hat{s}_k)$ inovasyon değeri yerine $p(b_k | s_k)$ olasılık değerlerini kullanılmaktadır. Bu olasılık değerlerinin kullanımı bir sonraki bölümde FE-Yoğunlaştırma algoritması kapsamında anlatılmaktadır.

FE-Yoğunlaştırma algoritması

FE-Yoğunlaştırma algoritmasının adımları Tablo 1’de verilmiştir. FE-Yoğunlaştırma algoritması

FE optimizasyon algoritmasıyla (Feoktistov, 2006), Yoğunlaştırma Algoritması’nın (Isard ve Blake, 1998; Price vd., 2005) bir birleşimidir. FE optimizasyon adımları Yoğunlaştırma Algoritması’nda kullanılan örneklerin iyileştirilmesini sağlamaktadır.

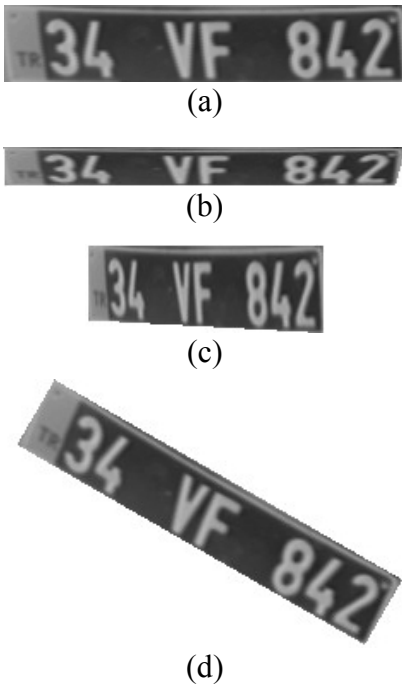
Yoğunlaştırma algoritmasının en önemli eksikliklerinden birisi parçacıkların güncellenmesinde yeni gelen ölçüm değerlerini kullanmamasıdır. Bu sebeple parçacıklar zaman güncelleme-sinden sonra olasılık dağılım fonksiyonunun kuyruğunda kalabilmekte ve olasılık dağılım fonksiyonunu iyi bir şekilde ifade etmekten uzak kalmaktadırlar. Böyle bir durumda FE algoritması yardımıyla düşük olasılıklı parçacıkların

Tablo 1. FE-Yoğunlaştırma algoritması

- Başlangıç: $\{s_0^n, \pi_0^n\}$ $n = 1, 2, \dots, N$ örnek kümesini yarat, burada s ’ler örnekler ve π ’ler ise örneklere karşı düşen ağırlıklar veya olasılıklardır.
- $k=1, 2, \dots$ için tekrarlayarak olasılık dağılımımızı ilerletelim, burada k zaman basamaklarıdır:
 - Örnekleri $\{s_{k-1}^n\}$ ’den $\{s_k^n\}$ ’e $s_k^n = f(s_{k-1}^n, c_k) + N(0, \sigma)$ denklemine göre ilerletelim.
 - Yeni olasılık ağırlıklarını, $\pi_k^n = p_1(b_k^n | s_k^n)$ denklemine göre hesaplayalım. Burada $p_1(b_k^n | s_k^n)$ verilen bir s_k^n durumuna karşılık b_k^n ölçümünün olasılığıdır. Ağırlıkları $\sum_n \pi_k^n = 1$. olacak şekilde güncelleyelim.
 - $G=1, 2, \dots, NI$ için tekrarlayarak NI nesil oluşturalım. Burada $\{s_{k,G=1}^n, \pi_{k,G=1}^n\} = \{s_k^n, \pi_k^n\}$.
 - Yeni nesli $\{s_{k,G+1}^n, \pi_{k,G+1}^n\}$ şöyle oluşturalım, her bir $s_{k,G}^n$ örneği için bunu tekrarlayalım.
 - $\{s_{k,G}^n, \pi_{k,G}^n\}$ içerisinde $\pi_{k,G}^n$ olasılığı ile örnekleme yapalım ve $(s_{k,G}^{r1}, s_{k,G}^{r2}, s_{k,G}^{r3})$ örneklerini seçelim, öyle ki $r1, r2, r3$ birbirinden farklı olsun.
 - $v_{G+1}^n = s_{k,G}^{r1} + F \cdot (s_{k,G}^{r2} - s_{k,G}^{r3})$ denklemine göre yeni bir v_{G+1}^n vektörü oluşturalım.
 - $s_{k,G}^n$ ve v_{G+1}^n arasında çaprazlama uygulayarak u_{G+1}^n vektörünü elde edelim.
 - Eğer $p_2(b_k | u_{G+1}^n) > p_2(b_k | s_{k,G}^n)$ değerinden büyük ise $s_{k,G+1}^n$ değerini u_{G+1}^n ’ye eşitleyelim, aksi halde $s_{k,G}^n$ değerini koruyalım.
 - Yeni $\pi_{k,G+1}^n = p_2(b_k | s_{k,G+1}^n)$ ağırlıklarını $\sum_n \pi_{k,G+1}^n = 1$ olacak şekilde güncelleyelim.
 - Yeni örnek kümesi $\{s_k^n, \pi_k^n\}, n = 1, 2, \dots, N$ böylece FE iyileştirmesi sonucunda elde edilir şöyle ki $\{s_k^n, \pi_k^n\} = \{s_{k,G=NI}^n, \pi_{k,G=NI}^n\}$.

daha yüksek olasılıklı bölgelere taşınması, olasılık dağılım fonksiyonunun ayrık değerlerle ifadesinde daha başarılı olunmasını sağlamaktadır.

FE algoritması ile Yoğunlaştırma algoritması benzerlikler göstermektedir. FE algoritmasındaki nesiller Yoğunlaştırma algoritmasında parçacıklara karşılık gelmektedir. Aynı şekilde her bir parçacığın olasılığı, FE algoritmasında her nesil örneğinin aldığı uygunluk değerine karşılık düşürülebilir. İki algoritma arasındaki bu analogi, algoritmaların birleştirilmesine olanak tanımaktadır.

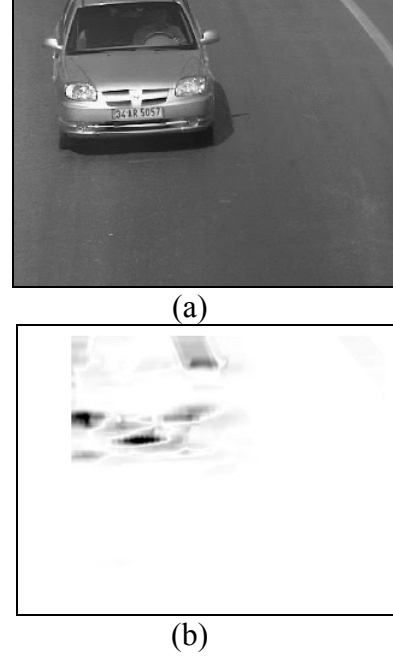


Şekil 3. Plakanın değişik dönüş açılarındaki görüntüsü (a) $\alpha = 0, \beta = 0, \gamma = 0$
 (b) $\alpha = 60^\circ, \beta = 0, \gamma = 0$ (c) $\alpha = 0, \beta = 60^\circ, \gamma = 0$
 (d) $\alpha = 0, \beta = 0, \gamma = 30^\circ$

Olasılık değerlerinin bulunması

Bu noktada açıklanması gereken konu, belirtilen olasılık değerlerinin nasıl hesaplandığıdır. Dikkat edilirse, FE-Yoğunlaştırma algoritmasında iki değişik olasılık değeri tanımlandı. Bunlar $\pi_k^n = p_1(b_k^n | s_k^n)$ ve $\pi_{k,G+1}^n = p_2(b_k | s_{k,G+1}^n)$ olasılık değerleridir. Bu koşullu olasılık değerleri FE-Yoğunlaştırma algoritmasında iki aşamalı bir iyileştirme olanağı sağlamaktadır.

İlk koşullu olasılık değeri $\pi_k^n = p_1(b_k^n | s_k^n)$, Şekil 4'de gösterilen Kayan Eş Merkezli Pencere algoritması (KEMP) (Anagnostopoulos vd., 2005) çıktısından elde edilmektedir.

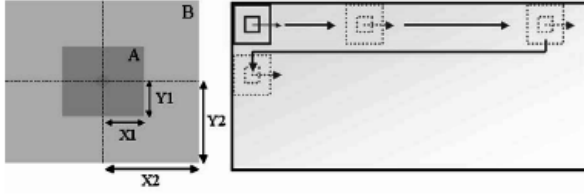


Şekil 4. KEMP filtre çıktısı (a) Orijinal görüntü
 (b) Kayan Eş Merkezli Pencere çıktısı

Kayan Eş Merkezli Pencere algoritması bir nokta için çevresinde bulunan iki adet pencerenin tanımladığı komşulukta, ortalama değer ve standart sapmanın oranlanması temeline dayanmaktadır. Bu iki pencereden biri diğerinin iki katı büyüklüğünde tanımlanmıştır. Böylece bu iki pencere resim üzerinde kaydırılarak pencereler içinde kalan benek değerlerinin ortalaması alınmakta ve standart sapmaları hesaplanmaktadır. Bu iki pencerede bulunan değerlerin birbirine oranlanması ise Şekil 4'te gösterilen çıktıyı üretmektedir. Şekil 5'te KEMP algoritmasında kullanılan A ve B pencerelerinin nasıl tanımlandığı gösterilmektedir. Burada $X1, Y1$ A pencere boyutları, $X2, Y2$ ise B pencere boyutlarıdır. Bu pencereler bütün görüntüde kaydırılarak ortalama ve standart sapma değerleri bulunur. Buna göre,

$$\begin{aligned} E[A] > E[B] &\Rightarrow \text{OrtOran} = \frac{E[A]}{E[B]}, \text{ değil } \text{OrtOran} = \frac{E[B]}{E[A]} \\ \sigma[A] > \sigma[B] &\Rightarrow \text{StdOran} = \frac{\sigma[A]}{\sigma[B]}, \text{ değil } \text{StdOran} = \frac{\sigma[B]}{\sigma[A]} \end{aligned} \quad (4)$$

$$\text{Uygunluk} = \text{OrtOran} + \text{StdOran}.$$



Şekil 5. KEMP algoritması

Her görüntü bölgesi için uygunluk değerleri hesaplanmaktadır. Bir s_k^n örneğinin olasılığı $\pi_k^n = p_1(b_k^n | s_k^n)$ izdüştüğü alandaki uygunluk değerlerinin yüksekliği ile belirlenmektedir.

İkinci koşullu olasılık $\pi_{k,G+1}^n = p_2(b_k | s_{k,G+1}^n)$ değeri ise Şekil 3’de görüldüğü şekilde her s_k^n örneği için elde edilen izdüşüm, döndürme ve ölçekleme işlemi sonucundaki görüntünün ilk karedeki görüntü ile karşılaştırılması ile elde edilmektedir. Bu karşılaştırma izdüşürülmüş s_k^n örneği ile görüntünün birbirinden çıkartılarak fark değerlerinin toplanması ile yapılmaktadır. Çıkartma işleminden önce görüntülerin ortalaması sıfır ve standart sapması bir olacak şekilde normalize edilmesi parlaklık değişimlerine karşı daha gürbüz sonuçlar vermektedir. İkinci koşullu olasılık fonksiyonu çok kısıtlı bir çerçevede iyileştirme imkânı vermektedir. Bu sebeple ilk olasılık fonksiyonuna ihtiyaç duyulmuştur. Bu şekilde iki seviyeli kaba ve ince optimizasyon gerçekleştirilmektedir. FE Yoğunlaştırma algoritması bu yapısı ile ciddi bir uygulama üstünlüğü sağlamaktadır.

Şekil 6’da örnek bir video karesi ile bu karede eşlenecek olan şablon gösterilmektedir. Şablon’un video üzerinde kaydırılarak karşılaştırma değerlerinin bulunması işlemi sonucunda $p_2(b_k | s_k^n)$ değerleri elde edilir. Bu eşleştirme işlemi şablonun, parçacık değerlerinde bulunan 3B koordinatlar ve Euler açıları kullanılarak kamera düzlemine iz düşürülmesi işleminden sonra yapılmaktadır.

Basit olarak gösterim amacıyla Şekil 7’de sadece iki boyutta kaydırma işlemi yapılarak karşılaştırma sonuçları normalize edilerek uygunluk değerleri gösterilmiştir. Böyle bir yöntem ile



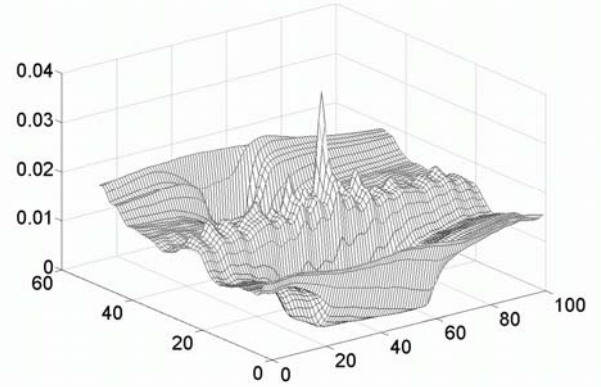
(a)



(b)

Şekil 6. Şablon karşılaştırma (a) Örnek video karesi (b) İlk kareden kesilen şablon

elde edilen uygunluk değerleri dar bir bölgede tepe yapmaktadır ve böyle bir fonksiyonun en yüksek değerinin bulunması işleminin güç olması nedeniyle iki seviyeli iyileştirmeye ihtiyaç duyulmaktadır. FE Yoğunlaştırma algoritması ilk önce parçacıkları $p_1(b_k^n | s_k^n)$ olasılık değerleri yardımıyla kaba olarak elemekte ve sonrasında $p_2(b_k | s_k^n)$ olasılık değerlerine göre iyileştirme yaparak plaka koordinatlarını daha az hata ile bulabilmektedir.



Şekil 7. Şablon karşılaştırma ile elde edilen uygunluk değerleri

Sonuçlar

Şekil 8’de ilk kareleri gösterilen test videosunda görüntülenen altı araç için köşe başına konum hata miktarları benek cinsinden Tablo 2’de görülmektedir. FE-Yoğunlaştırma algoritması, başarımlı değerlendirme amacıyla Yoğunlaştırma, Extended Kalman ve Genetik Yoğunlaştırma algoritmalarıyla

la (Zhu vd., 2005) karşılaştırılmıştır. Bütün algoritmalarda aynı olasılık değerleri ve aynı sayıda örnek (50) kullanılmıştır. Genetik ve FE-Yoğunlaştırma algoritmalarında iyileştirme 6 nesil ile sınırlı tutulmuştur.



(a)



(b)



(c)

Şekil 8. Test videosu 1. araç (a) 8. kare (b) 18. kare (c) 26. kare

Sonuçlar Yoğunlaştırma algoritmasının başarımının iyi olmadığını göstermektedir. Örnek değerlerini Extended Kalman filtresi ile ürettiğimiz uygulamada başarım, Yoğunlaştırma algoritmasına göre daha iyi çıkmıştır. Genetik Yoğunlaştırma algoritması Yoğunlaştırma algoritmasına göre ciddi bir iyileşme sağlamasıyla beraber, FE-Yoğunlaştırma algoritması en iyi başarımı sağlamıştır.

FE-Yoğunlaştırma algoritması iki farklı olasılık dağılımına göre iyileştirme imkânı tanıdığı için Yoğunlaştırma algoritmasına göre avantaj sağlamıştır. Öte yandan, FE-Yoğunlaştırma algoritması Yoğunlaştırma algoritmasıyla karşılaştırıldığında daha fazla işlem karmaşıklığına sahiptir.

Tablo 2. Test videosu için başarımların değerleri

Algoritma	Köşe Yerleştirme Hata Miktarı
Yoğunlaştırma	10.34 köşe başına benek
Extended Kalman	9.57 köşe başına benek
Genetik Yoğunlaştırma	6.17 köşe başına benek
FE-Yoğunlaştırma	4.24 köşe başına benek

Kaynaklar

- Anagnostopoulos, C., Anagnostopoulos, I., Tsekouras, G., Kouzas, G., Loumos, V., Kayafas, E., (2005). Using sliding concentric windows for license plate segmentation and processing, *Proceedings, Signal Processing Systems Design and Implementation*, 337 – 342, Athens.
- Ching-Tang, H., Yu-Shan, J., Kuo-Ming, H., (2005). Multiple license plate detection for complex background, *Proceedings, 19th International Conference, Advanced Information Networking and Applications AINA*, 2, 389 – 392, Taipei.
- Dubey, P., (2005). Heuristic approach for license plate detection, *Proceedings, IEEE Conference on Advanced Video and Signal Based Surveillance*, 366 – 370, Como.
- Feoktistov, V., (2006). *Differential Evolution In Search Of Solutions*, Springer, Berlin.
- Hsi-Jian, L., Si-Yuan, C., Shen-Zheng, W., (2004). Extraction and recognition of license plates of motorcycles and vehicles on highways, *Proceedings, 17th International Conference, Pattern Recognition, ICPR 2004*, 4, 356 – 359, Cambridge.
- Isard, M. ve Blake, A., (1998). A mixed-state condensation tracker with automatic model-switching, *Proceedings, Sixth International Conference on Computer Vision*, Jan., 107 – 112, Bombay.
- Price, K. V., Storn, R. M. ve Lampinen, J. A., (2005). *Differential Evolution a Practical Approach to Global Optimization*, Springer, Berlin.
- Ponsa, D., Lopez, A., Serrat, J., Lumberras, F., Graf, T., (2005). Multiple vehicle 3D tracking using an unscented Kalman Filter, *Proceedings, Intelligent Transportation Systems*, 1108 – 1113, Vienna.

- Sang Kyoon, K., Dae Wook, K., Hang Joon, K., (1996). A recognition of vehicle license plate using a genetic algorithm based segmentation, *Proceedings*, International Conference on Image Processing, **1**, 661 – 664, Lausanne.
- Vacchetti, L., Lepetit, V., Fua, P., (2004). Combining edge and texture information for real-time accurate 3D camera tracking, *Proceedings*, Third IEEE and ACM International Symposium Mixed and Augmented Reality, 48 – 56, Arlington.
- Wen-Yan, C., Chu-Song, C., Yi-Ping, H., (2005). Appearance-guided particle filtering for articulated hand tracking, *Proceedings*, IEEE Computer Society Conference Computer Vision and Pattern Recognition, **1**, 235 – 242, San Diego.
- Wenjing, J., Huafeng, Z., Xiangjian, H., Piccardi, M., (2005). Mean shift for accurate license plate localization, *Proceedings*, Intelligent Transportation Systems Proceedings, 566 – 571, Vienna.
- Xiong, J., Du, S., Gao, D., Shen, Q., (2004). Locating car license plate under various illumination conditions using genetic algorithm, *Proceedings*, 7th International Conference on Signal Processing ICSP '04, Volume 3, 2502 – 2505, Beijing.
- Zayed, M., Boonaert, J., Bayart, M., (2004). License plate tracking for car following with a single camera, *Proceedings*, The 7th International IEEE Conference on Intelligent Transportation Systems, 719 – 724, Washington.
- Zhu, Y., Zhi-Qiang, L., (2005). Genetic CONDENSATION for motion tracking, *Proceedings*, International Conference on Machine Learning and Cybernetics, **9**, 5542 – 5547, Guangzhou.