

İnsansız Hava Araçları için Eşanlı Konumlandırma ve Haritalama 2D Simultaneous Localization and Mapping for Unmanned Aerial Vehicles

Mehmet Kök, Billur Barshan

Elektrik ve Elektronik Mühendisliği Bölümü
Bilkent Üniversitesi, 06800 Ankara

mkok@ee.bilkent.edu.tr, billur@ee.bilkent.edu.tr

Özetçe

Sabit yükseklikte uçtuğu varsayılan İnsansız Hava Araçları (İHA) için görü-tabanlı bir Eşanlı Konumlandırma ve Haritalama (EKVH) algoritması sunulmaktadır. Araç üzerindeki kameradan elde edilen görüntülerin öznelikleri kullanılarak farklı yer işaretleri saptanmaktadır. Yer işaretlerine Genişletilmiş Kalman Süzgeci (GKS) uygulanmış ve EKVH probleminin çözümü için bazı benzetim sonuçları sunulmuştur.

Abstract

In this work, a 2D vision-based Simultaneous Localization and Mapping algorithm is developed for an Unmanned Aerial Vehicle flying at a constant altitude. We use visual features of images obtained from an on-board camera to identify different landmarks. Using these landmarks we apply the well-known Extended Kalman Filter to the SLAM problem and present some simulation results.

1. Giriş

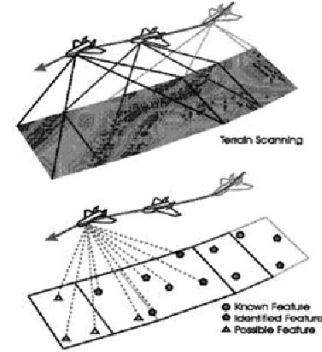
İnsansız hava araçları günümüzde keşif, gözetleme, hedef saptama ve izleme gibi amaçlarla kullanılabilen oldukça özerk araçlar haline gelmektedir. Tam özerkliği sağlamanın önemli bir adımı da gezinimde özerkliği sağlamaktır. İHA, kendi üzerindeki algılayıcıları kullanarak eş zamanlı olarak hem bir harita çıkarmalı hem de bu haritayı kullanarak kendi konumunu doğru olarak belirleyebilmelidir. Bu da EKVH problemi olarak bilinmektedir.

EKVH problemine farklı olasılıksal çatılarda farklı çözümler önerilmiştir. Bunlar arasında özellikle Kalman ve parçacık süzgeçleri [4, 11] ile Beklenti-Enbüyütme algoritması [10] yaygın olarak kullanılmaktadır. Yapılan çalışmalarda bu yöntemler daha çok gezgin yer araçlarında lazer, sonar, radar gibi uzaklık algılayıcılarından elde edilen verilerle odometri ve eylemsizlik duyucularından elde edilen bilgiler birleştirilerek kullanılmıştır. Görüntü tabanlı EKVH algoritmaları ise görece daha yeni bir yaklaşımdır. Bunlarda ise genelde Harris köşe sezicisi [6] veya Lowe tarafından geliştirilmiş olan Ölçekten Bağımsız Öznitelik Dönüşümü (ÖBÖD) [8] gibi algoritmalar ile kamera görüntülerinden elde edilen ayırdedilebilir noktalar dayanak noktaları olarak kullanılarak yine Kalman Süzgeci türevleriyle ya da parçacık süzgeci gibi tekniklerle EKVH problemi çözülmeye çalışılmıştır [3, 5, 9].

Hava araçlarında EKVH problemi daha yakın zamanda ele alınan bir problem olup yer araçlarına göre özellikle

hareket dinamiği ve gürültü seviyeleri nedeniyle farklı zorluklar göstermektedir. Kim ve Sukkarieh gerçek bir İHA'nda, yere dik bakan kamera ve kontrol girdileri aracılığıyla EKVH algoritmasını uygulamışlar ve başarılı sonuçlar elde etmişlerdir [7]. Sözkonusu bu çalışmada sahaya yapay olarak yerleştirilen beyaz yer işaretleri bir görüntü algılayıcı tarafından sezimlenmekte ve bu işaretlerin bilinen boyutları kullanılarak uzaklık bilgisi kestirilmektedir. Angeli *et al.* tarafından yapılan çalışmada ise tamamen görüntü tabanlı bir EKVH algoritması önerilmektedir [1]. Bu çalışmada odometri bilgisi kameradan gelen görüntülerden kestirilmekte ve Kanade-Lucas-Tomasi (KLT) izlemesi ile bulunan noktalar ÖBÖD betimleyicileri ile birlikte yer işaretleri olarak kullanılmaktadır.

Bu çalışmada sabit yükseklikte uçtuğu varsayılan bir İHA için 2B görüntü-tabanlı bir EKVH algoritması geliştirilmiştir. İHA üzerindeki kameradan elde edilen görüntülere ÖBÖD algoritması uygulanarak elde edilen yer işaretleri GKS ile EKVH algoritmasında kullanılmaktadır [2].

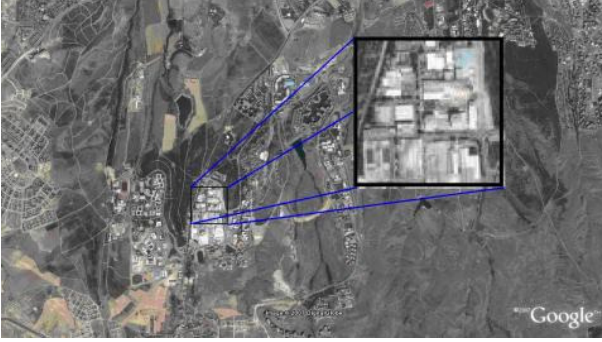


Şekil 1: İHA hareketi ve yer işaretleri [7].

2. Sistem Bilgisi

Bu çalışmada benzetimi yapılan İHA (Şekil 1) Şekil 2'deki görüntü ile gösterilen alanda (Bilkent Üniversitesi Kampüsü, Ankara) sabit yükseklikte uçmakta ve yere dik bakan bir kamera ile görüntü almaktadır. Bu alan GoogleEarth programından sabit yükseklikte elde edilen 1280×719 boyutunda bir görüntü ile temsil edilmektedir ve kamera görüş alanı bu büyük görüntünün 80×80'lik alt-görüntüleri olarak alınmıştır.

İHA'nın ve yer işaretlerinin konumları ile uzaklıklar piksel cinsinden verilmektedir.



Şekil 2: Benzetimde İHA'nın taradığı varsayılan alan.

Benzetim için kullanılan araç modeli ise $x - y$ koordinatları ve de yönelim açısı olmak üzere 3 serbestlik derecesine sahip bir modeldir. Kalman süzgecinde kullanılan durum vektörü İHA'nın konumu (x_r, y_r) , yönelim açısı (θ_r) ve yer işaretlerinin konumlarından (x_{l_j}, y_{l_j}) oluşmaktadır:

$$\begin{aligned} X_{IHA} &= [x_r, y_r, \theta_r], \\ X_{map} &= [x_1, y_1, x_2, y_2, \dots, x_n, y_n], \\ X &= [X_{UAV}, X_{map}]. \end{aligned} \quad (1)$$

Burada n toplam yer işareti sayısıdır.

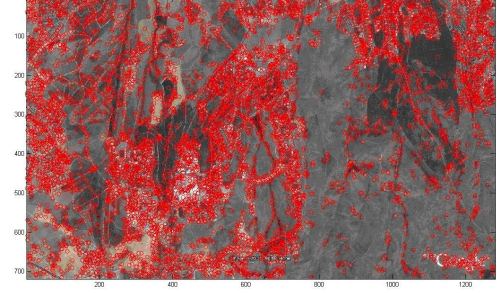
İHA'nın hareket modeli ise Denklem (2)'de verilmiştir. Denklem (2)'de U birim zamandaki hız, q ise her serbestlik derecesiyle ilişkilendirilmiş sıfır ortalamalı, zamandan bağımsız Gauss dağılımlı gürültü olarak verilmiştir.

$$\begin{aligned} x_r(t+1) &= x_r(t) + U_x + q_x, \\ y_r(t+1) &= y_r(t) + U_y + q_y, \\ \theta_r(t+1) &= \theta_r(t) + U_\theta + q_\theta. \end{aligned} \quad (2)$$

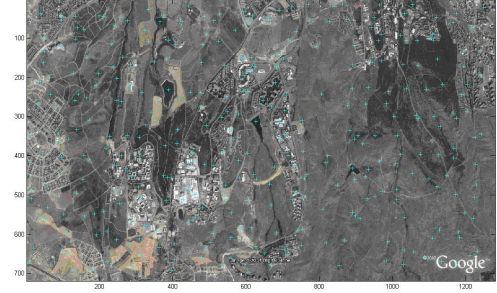
3. Görüntüden Yer İşaretleri Belirleme

Yer işaretleri bir alanda dayanak noktası olarak alınabilecek ayırt edilebilir noktalar olarak tanımlanmıştır. Yer işaretlerini algılamak ve belirlemek amacıyla ÖBÖD kullanılmaktadır. ÖBÖD görüntü öznelikleri kullanılarak görüntüler arası eşleştirme ve görüntüden nesne tanıma gibi amaçlarla geliştirilmiş bir öznelik özütleme algoritmasıdır. Bu algoritma ile elde edilen öznelikler görüntü ölçeğine, dönmeye, kısmi olarak da bakış açısına, aydınlatmaya ve kısmi örtmelere karşı dayanıklıdır [8].

ÖBÖD, Şekil 2'de gösterilen imgeden gerek duyulandan çok daha fazla (yaklaşık 9000) öznelik noktası çıkarmaktadır (Şekil 3-(a)). Bu kadar çok sayıda nokta hem EKVH algoritması için gereksiz hem de GKS için hesaplama açısından uygun değildir. ÖBÖD noktalarını azaltmak için tüm alan 30×30 piksellik hücrelere bölünerek, her hücreden ÖBÖD belirleyici vektör büyüklüğü en fazla nokta seçilerek yaklaşık 900 noktadan oluşan küçültülmüş bir yer işareti veritabanı oluşturulmuştur (Şekil 3-(b)). Bu aşamada bu işlemi algoritma çalışırken yapabilecek bir yöntem gerekmektedir.



(a)



(b)

Şekil 3: (a) ÖBÖD'le bulunan tüm öznelikler ve konumları. (b) Yer işaretleri azaltma işlemi sonucu kalan ÖBÖD noktaları.

4. Genişletilmiş Kalman Süzgeci

4.1. Hareket ve Gözlem Modeli

İHA için verilen hareket modeli aşağıdaki durum denklemleriyle gösterilebilir:

$$\mathbf{x}(k) = \begin{bmatrix} x_r(k) \\ y_r(k) \\ \theta_r(k) \\ \vdots \\ x_{l_j}(k) \\ y_{l_j}(k) \\ \vdots \end{bmatrix} \quad \mathbf{u}(k) = \begin{bmatrix} u_1(k) \\ u_2(k) \end{bmatrix},$$

$$\mathbf{x}(k) = \begin{bmatrix} x_r(k-1) + \cos(\theta_r(k-1))u_1(k) \\ y_r(k-1) + \sin(\theta_r(k-1))u_1(k) \\ \theta_r(k-1) + u_2(k) \\ \vdots \\ x_{l_j}(k-1) \\ y_{l_j}(k-1) \\ \vdots \end{bmatrix} + \mathbf{w}(k).$$

Burada $\mathbf{x}(k)$ durum vektörü, $\mathbf{u}(k)$ sisteme verilen kontrol girişleri, $\mathbf{w}(k)$ ise $\mathbf{Q}(k)$ kovaryansa sahip sıfır ortalamalı, zamandan bağımsız Gauss dağılımlı süreç gürültüsüdür. EKVH probleminin doğası gereği durum vektörü $\mathbf{x}(k)$ 'nin boyutu yeni yer işaretleri eklendikçe büyümektedir. Yer işaretleri hareketle değişmediklerinden, bu durumlar için süreç gürültüsü sıfırdır.

Gözlem modeli ise şöyle özetlenebilir:

$$\mathbf{z}(k) = \begin{bmatrix} h_1(\mathbf{x}(k), l_1) \\ h_2(\mathbf{x}(k), l_2) \\ \vdots \\ h_n(\mathbf{x}(k), l_n) \end{bmatrix} + \begin{bmatrix} v_1(k) \\ v_2(k) \\ \vdots \\ v_n(k) \end{bmatrix}, \text{ burada } l_j = \begin{bmatrix} x_{l_j} \\ y_{l_j} \end{bmatrix}.$$

$$h_j(\mathbf{x}(k), l_j) = \begin{bmatrix} \sqrt{\phi^2 + \psi^2} \\ \arctan2(-\psi, -\phi) - \theta_r(k) \end{bmatrix}.$$

Burada h_j , j 'inci yer işaretinin araca göre uzaklığını ve yönelim açısını geri veren bir fonksiyon olup $\phi = x_r(k) - x_{l_j}(k)$, $\psi = y_r(k) - y_{l_j}(k)$ 'dir. Her yer işareti her adımda gözlemlenmediği için bir veri eşleştirme adımı gerekmektedir. Sistem doğrusal olmadığı için GKS kullanılmakta olup sistemin süreç ve gözlem modelleri şu şekilde özetlenebilir:

$$\begin{aligned} \mathbf{x}(k) &= \mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k), k) + \mathbf{w}(k), \\ \mathbf{y}(k) &= \mathbf{h}(\mathbf{x}(k), k) + \mathbf{v}(k). \end{aligned} \quad (3)$$

4.2. GKS Öngörü Adımı

Hareket modeli doğrusal olmadığı için GKS'de kullanılmak üzere Denklem (3)'de verilen $\mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k), k)$ fonksiyonunun gradyanı hesaplanarak aşağıda gösterilen \mathbf{F} matrisi elde edilir:

$$\mathbf{F}(k) = \nabla \mathbf{f} = \begin{bmatrix} 1 & 0 & -\sin \theta_r(k) u_1(k) & \mathbf{0} \\ 0 & 1 & \cos \theta_r(k) u_1(k) & \mathbf{0} \\ 0 & 0 & 1 & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{I} \end{bmatrix}.$$

Burada elde edilen \mathbf{F} kullanılarak öngörü adımı şu şekilde oluşturulur:

$$\begin{aligned} \mathbf{x}(k|k-1) &= \mathbf{f}(\mathbf{x}(k-1), \mathbf{u}(k), k), \\ \mathbf{P}(k|k-1) &= \mathbf{F}(k)\mathbf{P}(k-1|k-1)\mathbf{F}(k)^T + \mathbf{Q}(k). \end{aligned}$$

Burada yine $\mathbf{Q}(k)$ yer işaretlerine karşılık gelen durumlar için sıfırdır.

4.3. GKS Güncelleme Adımı

Gözlem modeli de doğrusal olmadığı için aynı şekilde $\mathbf{h}(\mathbf{x}(k), k)$ gözlem fonksiyonunun gradyanı alınarak \mathbf{H} matrisi aşağıdaki gibi elde edilmektedir:

$$\mathbf{H}(k) = \nabla \mathbf{h} = \begin{bmatrix} \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ \frac{\phi}{p_j} & \frac{\psi}{p_j} & 0 & 0 \dots 0 & \frac{-\phi}{p_j} & \frac{-\psi}{p_j} & 0 \dots 0 \\ \frac{-\psi}{p_j^2} & \frac{\phi}{p_j^2} & 1 & 0 \dots 0 & \frac{\psi}{p_j^2} & \frac{-\phi}{p_j^2} & 0 \dots 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots \end{bmatrix}.$$

Burada $p_j = \sqrt{\phi^2 + \psi^2}$ 'dir.

Her döngüde her yer işaretini gözlemleyemediğimizden, her bir döngüde sadece bazı yer işaretleri için gözlem değerleri bulunmaktadır. Bu nedenle güncelleme denkleminde sadece gözlemlenmiş durumları katmak için her bir gözlem ayrı ayrı elde edilmiş gibi ayrı ayrı hesaba katılmaktadır. Bu işlem için \mathbf{H} matrisinden elde edilen aşağıdaki \mathbf{H}_j alt-matrisi kullanılmakta ve sadece gözlemlenen duruma karşılık gelen değerler güncellenmektedir:

$$\mathbf{H}_j(k) = \nabla \mathbf{h}_j = \begin{bmatrix} \frac{\phi}{p_j} & \frac{\psi}{p_j} & 0 & 0 \dots 0 & \frac{-\phi}{p_j} & \frac{-\psi}{p_j} & 0 \dots 0 \\ \frac{-\psi}{p_j^2} & \frac{\phi}{p_j^2} & 1 & 0 \dots 0 & \frac{\psi}{p_j^2} & \frac{-\phi}{p_j^2} & 0 \dots 0 \end{bmatrix}.$$

Güncelleme denklemleri şu şekilde listelenebilir:

$$\begin{aligned} \mathbf{x}(k|k) &= \mathbf{x}(k|k-1) + \mathbf{W}\mathbf{v}, \\ \mathbf{P}(k|k) &= \mathbf{P}(k|k-1) - \mathbf{W}\mathbf{H}(k)\mathbf{P}(k|k-1), \\ \text{burada } \mathbf{v} &= \mathbf{z}(k) - \mathbf{h}(\mathbf{x}(k|k-1), k), \\ \mathbf{S} &= \mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^T + \mathbf{R}(k), \\ \mathbf{W} &= \mathbf{P}(k|k-1)\mathbf{H}(k)^T\mathbf{S}^{-1}, \\ \mathbf{R}(k) &= \begin{bmatrix} \mathbf{R}_1(k) & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2(k) & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \dots \end{bmatrix}. \end{aligned}$$

4.4. Yeni Yer İşareti Eklenmesi

EKVH probleminin doğası gereği bir döngüde gözlemlenen yer işaretleri haritada bulunan daha önceki gözlemlenmiş işaretlerle eşleşebileceği gibi, bu döngüde yeni yer işaretleri de gözlemlenebilir. Bu durumda bu yeni işaretlerin de haritaya eklenerek haritanın güncellenmesi gerekir. Bu amaçla hem durum vektörü $\mathbf{x}(k)$ hem de durum vektörüne bağlı kovaryans matrisi $\mathbf{P}(k)$ güncellenmelidir.

Aşağıda yer işareti iklendirme fonksiyonu $\mathbf{g}(\mathbf{x}(k), \mathbf{z}_j(k))$, yer işareti j 'in İHA'na göre uzaklığı r_j ve yönelim açısı θ_j 'in fonksiyonu olarak verilmiştir:

$$\begin{aligned} \mathbf{g}(\mathbf{x}(k), \mathbf{z}_j(k)) &= \begin{bmatrix} x_r(k) + r_j \cos(\theta_j + \theta_r(k)) \\ y_r(k) + r_j \sin(\theta_j + \theta_r(k)) \end{bmatrix}, \\ \mathbf{x}(k+1) &= \begin{bmatrix} \mathbf{x}(k) \\ \mathbf{g}(\mathbf{x}(k), \mathbf{z}_j(k)) \end{bmatrix}, \\ \mathbf{P}(k+1) &= \nabla \mathbf{Y}_{x,z} \mathbf{P}(k) \nabla \mathbf{Y}_{x,z}^T, \\ \nabla \mathbf{Y}_{x,z} &= \begin{bmatrix} \mathbf{I}_{n \times n} & \mathbf{0}_{n \times 2} \\ \nabla \mathbf{G}_x & \nabla \mathbf{G}_z \end{bmatrix}. \end{aligned}$$

Burada $\nabla \mathbf{G}_x$ ve $\nabla \mathbf{G}_z$, $\mathbf{g}(\mathbf{x}(k), \mathbf{z}_j(k))$ fonksiyonunun $\mathbf{x}(k)$ ve $\mathbf{z}_j(k)$ 'ye bağlı gradyanları olarak şu şekilde verilmiştir:

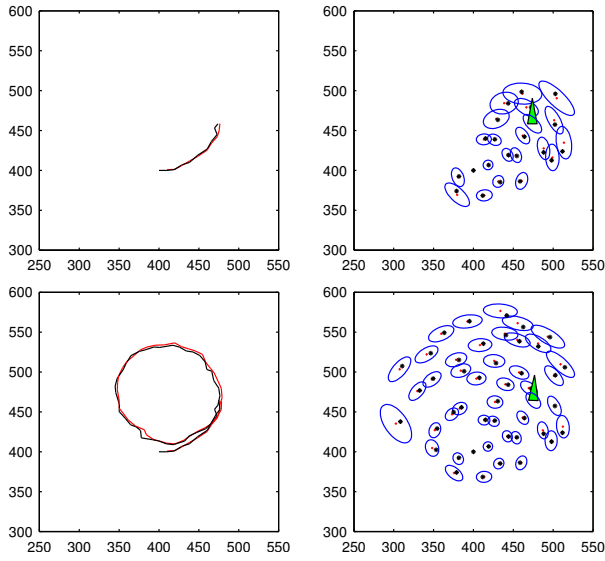
$$\begin{aligned} \nabla \mathbf{G}_x &= \begin{bmatrix} 1 & 0 & -r_j \sin(\theta_j + \theta_r(k)) & 0 & \dots & 0 \\ 0 & 1 & r_j \cos(\theta_j + \theta_r(k)) & 0 & \dots & 0 \end{bmatrix}, \\ \nabla \mathbf{G}_z &= \begin{bmatrix} \cos(\theta_j + \theta_r(k)) & -r_j \sin(\theta_j + \theta_r(k)) \\ \sin(\theta_j + \theta_r(k)) & r_j \cos(\theta_j + \theta_r(k)) \end{bmatrix}. \end{aligned}$$

5. Benzetim Sonuçları

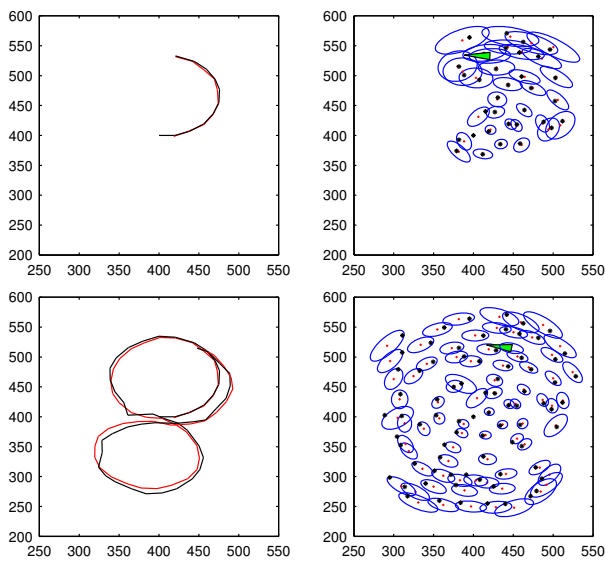
Yer işaretleri olarak küçültülmüş ÖBÖD veritabanı ve yukarıdaki denklemler kullanılarak MATLAB'da bir benzetim ortamı oluşturulmuştur. Burada biri dairesel diğeri sekiz şeklinde olmak üzere iki örnek yürütüm verilmiştir (Şekil 4 ve 5). Burdaki kırmızı yol aracın süreç gürültüsü sonucu gittiği gerçek yolu, siyah yol ise EKVH tarafından bulunan kestirilmiş yolu göstermektedir. Yer işaretleri daha fazla gözlemlendikçe kestirilen yol gerçek yola daha fazla yaklaşmaktadır, ancak yeni yer işaretleri gözlemlendiğinde gözlem gürültüsü sonucu kestirilen yol, gerçek yoldan uzaklaşmaktadır. Şekillerde soldaki grafikler gerçek ve kestirilmiş yolları, sağdaki şekiller ise

yer işaretlerinin gerçek ve kestirilmiş konumlarını 3σ güven sınırları ile birlikte göstermektedir. Burda kestirilmiş yol ve yer işaretleri konumlarının gerçek yol ve konumlara yakınsaması gözlenebilmektedir. Yeni gözlemlenmiş yer işaretlerinin 3σ elipsleri büyük olup, çok kere gözlemlenmiş yer işaretlerinin 3σ elipsleri ise giderek küçülmektedir.

Burada dikkat çekilmesi gereken önemli noktalardan biri yer işaretleri sayısının oluşturduğu sorundur. Her ne kadar kullanılan yöntemle yer işareti sayısı ciddi oranda azaltılmış olsa da bu kadar yer işareti ile bile GKS algoritması giderek daha yavaş çalışmaktadır. Her yeni yer işareti ile durum vektörü boyutu artmakta, buna bağlı kovaryans matrisi ve güncelleme adımıdaki matrislerin boyutları büyümekte, bu nedenle her döngünün hesap süresi artmaktadır.



Şekil 4: Dairesel yol.



Şekil 5: Sekiz şeklinde yol.

6. Sonuç

Bu çalışmada görüntü öznelikleri kullanılarak görü-tabanlı bir EKVH algoritması geliştirilmiştir. Öznelikler yaygın olarak kullanılan ÖBÖD algoritması ile elde edilmiş, ve basit bir seçme yöntemi ile sayıları azaltılmıştır. ÖBÖD noktalarının azaltılması ya da daha uygun öznelik noktaları bulunması konusunda çalışmalarımız devam etmektedir. Hareket ve gözlem modeli doğrusal olmadığından EKVH GKS tabanlı olarak gerçekleştirilmiş ve bazı benzetim sonuçları sunulmuştur.

7. Teşekkür

Mehmet Kök TÜBİTAK yüksek lisans bursu ile, sunulan çalışma ise kısmen TÜBİTAK EEEAG-105E065 projesi tarafından desteklenmektedir.

8. Kaynakça

- [1] Angeli, A. and Filliat, D. and Doncieux, S. and Meyer, J.-A. 2D Simultaneous localization and mapping for micro aerial vehicles. In *Proc. European Micro Aerial Vehicles Conf.*, 2006.
- [2] Bar-Shalom, Y. and Li, X.R. *Estimation and Tracking-Principles, Techniques, and Software*. Artech House, Inc, Norwood, MA, 1993.
- [3] Davison, A.J. and Murray, D.W. Simultaneous localization and map-building using active vision. *IEEE T. Pattern Anal.*, 24(7):865–880, 2002.
- [4] Dissanayake, M. W. M. G., Newman, P., Clark, S., Durrant-Whyte, H.F., Csorba, M. A solution to the simultaneous localization and map building (SLAM) problem. *IEEE T. Robot. Autom.*, 17(3):229–241, June 2001.
- [5] Elinas, P. and Sim, R. and Little, J. J. σ SLAM: stereo vision SLAM using the Rao-Blackwellised particle filter and a novel mixture proposal distribution. *Proc. IEEE Int. Conf. Robotics Automation*, pages 1564–1570, May 15–19, 2006.
- [6] Harris, C. and Stephens, M. A combined corner and edge detector. *Proc. Fourth Alvey Vision Conf., Manchester*, 15:50, 1988.
- [7] Kim, J. and Sukkarieh, S. Real-time implementation of airborne inertial-SLAM. *Robot. Auton. Syst.*, 55:62–71, 2007.
- [8] Lowe, D. G. Distinctive Image Features from Scale-Invariant Keypoints. *Int. J. Comput. Vision*, 60(2):91–110, 2004.
- [9] S. Se, DG Lowe, and J. J. Little. Vision-based global localization and mapping for mobile robots. *IEEE T. Robot. Autom.*, 21(3):364–375, 2005.
- [10] Thrun, S. Robotic Mapping: A Survey. In G. Lakemeyer and B. Nebel, editors, *Exploring Artificial Intelligence in the New Millenium*. Morgan Kaufmann, San Francisco, CA, USA, 2002.
- [11] Thrun, S. and Montemerlo, M. and Koller, D. and Wegbreit, B. and Nieto, J. and Nebot, E. FastSLAM: An efficient solution to the simultaneous localization and mapping problem with unknown data association. *J. Mach. Learn. Res.*, 4(3):380–407, 2004.