

El Şekil Bilgisinin Çıkarılması ve Takibi ile Sanal Mouse Kontrolü

Virtual Mouse Control with Hand Gesture Information Extraction and Tracking

Ramazan Özgür DOĞAN
Bilgisayar Teknolojileri Bölümü
Gümüşhane Üniversitesi
Gümüşhane, Türkiye
ramazan.dogan@gumushane.edu.tr

Hülya DOĞAN, Cemal KÖSE
Bilgisayar Mühendisliği Bölümü
Karadeniz Teknik Üniversitesi
Trabzon, Türkiye
{hulya, ckose}@ktu.edu.tr

Özetçe—Görüntü tabanlı el hareket takibi ile gerçekleştirilen sanal mouse uygulamaları insan bilgisayar etkileşimi alanında önemli çalışmalardan biridir. Yapılan bu çalışmada insan bilgisayar etkileşimini sanal mouse uygulamasıyla gerçekleştirmek amaçlanmıştır. Çalışma el bölgesi takibi, takip bölgesinden özellik çıkarımı ve çıkarılan bu özelliklerin sınıflandırılması olarak üç ana bölümden oluşmaktadır. Çalışmada el bölgesi takibi Camshift (Continuously Adaptive Mean Shift) algoritması ile gerçekleştirilmiş, takip bölgesinden görsel kelime çantaları ile özellikler çıkarılıp gruplandırılmış ve gruplanan bu özellikler destek vektör makineleri ile sınıflandırılarak farklı mouse hareketleri gerçekleştirilmiştir. Takip, özellik çıkarma ve sınıflandırma yöntemlerinin başarılarının ölçümü için ayrıntılı testler yapılmış ve sistemin yüksek başarımla güvenilir olarak çalıştığı görülmüştür.

Anahtar Kelimeler — insan bilgisayar etkileşimi; camshift; görsel kelime çantaları; destek vektör makineleri

Abstract— Virtual mouse implemented with hand gesture tracking based on image is one of studies in human-computer interaction. In this study that human-computer interaction is implemented with virtual mouse is purposed. This study consists of three main steps that are hand gesture tracking, features of hand region extraction and classification of these features. In this study hand gesture tracking is generated with Camshift (Continuously Adaptive Mean Shift) algorithm, features of hand gestures are extracted with bag of visual words and these features are classified with support vector machines. Detailed tests are performed to compare success of tracking, features extraction and classification methods and that this system works successfully is shown.

Keywords — human-computer interaction; camshift; bag of visual words; support vector machines; virtual mouse.

I. GİRİŞ

Geliştirilen bilgisayar sistemlerinin daha kolay ve etkin kullanımı için mouse, klavye gibi bilgisayar-insan etkileşim araçları kullanılmakta ve bu araçların kısıtlamalarını engellemek için günden güne yeni çalışmalar yapılmaktadır [1]. El bölgesi tanıma ve takip sistemleri bilgisayar-insan arası etkileşimi daha kolaylaştırmakta ve kullanıcıların herhangi fiziksel temasta bulunmadan bilgisayarı yönetmesi için sanal mouse uygulamasının temelini oluşturmaktadırlar. Literatürde el bölgesi tespit yaklaşımları görsel ve elektronik eldiven tabanlı olmak üzere iki gruba ayrılmaktadır. Elektronik eldiven tabanlı yaklaşımlar bilgisayar ile iletişim kuracak kişinin ağır bir araç olan eldiven giymesini gerektirmektedir. Bu insan ile bilgisayar arasındaki iletişim seviyesinin düşmesine neden olmaktadır. Görsel tabanlı yaklaşımlarda ise el bölgesinin kişiden kişiye değişmesi, arka plan değişikliklerine olan duyarlılık ve ışıktan etkilenme gibi sebeplerden dolayı etkin bir algoritma geliştirilememiştir. Ek olarak literatürde el bölgesi takibi için ise geliştirilmiş ilk çalışmalarda kullanılan yaklaşımların çoğu ten algılamaya dayalı algoritmalarlardır [2]. Elin ten rengi kameradan bilgisayara aktarılan görüntülerden el takibi yapmak için uzun bir süre kullanılmış bir özelliktir. Snake [3], Meanshift [4], SIFT (Scale Invariant Feature Transform) [5], SURF (Speed Up Robust Features) [6], Kalman Filtresi [7] ve Condensation [8] algoritmaları el bölgesi takibi amacıyla kullanılan diğer yaklaşımlardandır. Literatürde sanal mouse gerçekleştirmeyi amaçlayan diğer bir çalışma ise Zhou ve arkadaşlarının yaptığı çalışmadır. Bu çalışmada mouse kontrolleri el hareketleri ile gerçekleştirmiş, el bölgesi bulmak için SIFT özellik çıkarma algoritması kullanılmış, el bölgesi takibi ise Camshift algoritması ile gerçekleştirilmiştir [9].

Yapılan çalışmada kullanıcıya hiçbir cihaza fiziksel temasta bulunmadan bilgisayar yönetimi ve kontrolü

sağlamak amacıyla sanal mouse uygulaması gerçekleştirilmiştir. Sanal mouse uygulaması için önerilen yaklaşımda ilk olarak bilgisayar kontrol edecek kişinin elini kamera görüş alanında belirlenen hedef çerçeveye yerleştirilmesi beklenmektedir. Hedef çerçevede belirlenen eşleme el şekli tespit edilince Camshift algoritması başlatılmakta ve el bölgesinin takibi gerçekleştirilmektedir. El bölgesi takibi esnasında farklı el pozisyonları için özellik çıkarımı gerçekleştirilmekte, daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile bilgisayar kontrolü gerçekleştirilmektedir. Yapılan çalışmada el bölgesi özellikleri çıkarılması için GKÇ (Görsel Kelime Çantaları), çıkarılan el bölgesi özelliklerinin sınıflandırılması için ise DVM (Destek Vektör Makineleri) kullanılmaktadır.

II. SANAL MOUSE UYGULAMASI

Yapılan çalışma el bölgesi takibi, el bölgesi özelliklerinin çıkarılması ve el bölgesi özelliklerinin sınıflandırılması olmak üzere üç işlem adımından oluşmaktadır.

A. El Bölgesi Takibi

Yapılan çalışmada el bölgesi takibi için Meanshift temeline dayalı Camshift algoritması kullanılmıştır. Camshift algoritması ilk olarak bir kullanıcı arabiriminde etkin bir şekilde yüz izleme gerçekleştirmek için tasarlanmıştır. Bu algoritma adaptif olasılık dağılımı kullanmakta ve görüntü özelliklerine göre hedef çerçevenin boyutları değişmektedir.

B. El Bölgesi Özelliklerinin Çıkarılması

Çalışmada takip sırasında kameradan gelen görüntülerin özellik çıkarılması sürecinde gerçek zamanlı deneysel sonuçlarının daha tatminkâr olduğu görülerek GKÇ yöntemi kullanılmıştır. GKÇ görüntüdeki bulunmuş anahtar noktaları etrafındaki bölgeleri temsil eden yerel tanımlayıcıları bulmak için geliştirilmiştir [10]. GKÇ modeli genel olarak dört adımdan oluşmaktadır:

Adım 1. Anahtar noktası belirlemek için metot seçme ve anahtar noktaların etrafında yerel bölgeleri tanımlama:

Anahtar nokta belirleme ve yerel tanımlayıcılar görsel kelime çantaları modelinin ön aşaması olarak düşünülmekte ve bu modelin temelini oluşturmaktadırlar. Literatürde yapılan çalışmalarda farklı belirleyiciler kullanılmıştır. Belirleyiciler kullanılan özellik ve boyuta göre farklı sonuçlar göstermektedirler. Harris, SIFT, SURF özellik çıkarma algoritmaları literatürde kullanılan popüler anahtar noktası belirleme algoritmalarındandır. Literatür araştırmalarında SURF algoritmasının SIFT algoritmasına göre daha hızlı olduğu görülmüş ve bu yüzden çalışmada anahtar noktası belirleyici olarak SURF algoritması kullanılmıştır.

Adım 2. Görsel sözlük oluşturulması:

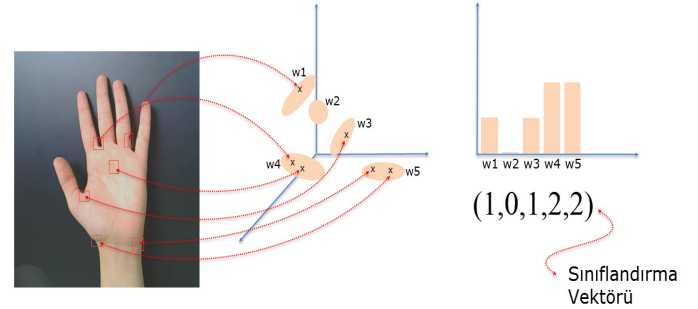
Görüntü üzerinde özellik noktası belirlemek için SIFT ve SURF gibi yerel tanımlayıcılar kullanıldığında çok sayıda özellik noktası bulunmaktadır. Bu yoğunluk dayanıklılık için önemli olmakla birlikte özellik sayısı boyunun fazla olması işlem yükünü artırmaktadır. Görsel kelime çantaları ile konumları değiştirilmeden özellik noktalarının toplanıp sayısallaştırılması sağlanarak tanımlayıcılar kısaltılmaktadır. Görsel kelimelerde yerel tanımlayıcıların sayısallaştırılması için görsel sözlükler oluşturulmalıdır. Görsel kelime sözlüklerini oluşturmak için özellik noktalarını giriş olarak alan kümeleme metotları kullanılmaktadır. Kümeleme metotları kullanılarak her küme için sözlükten tek bir görsel kelime atanmaktadır. Görsel sözlükler en yakın kümelerin merkezlerinin etiketlenmesi ile özellik noktalarının sayısallaştırılmasında kullanılmaktadır. Yapılan çalışmada kümeleme metodu olarak k-ortalama algoritması kullanılmıştır.

Adım 3. Görsel sözlük kullanılarak anahtar noktaların nicelenmesi:

Yerel tanımlayıcıların niceleştirilmesi bulunan özellik noktalarının herhangi bir görsel kelimeye atanması sürecidir. Bu adımda kümelenen her gruba ait anahtar noktası belirlenerek görsel kelimeler niceleştirilir.

Adım 4. Görsel kelimelerin frekansları ile görüntünün temsil edilmesi:

Bir önceki adımda nicelenen her görsel kelime art arda birleştirilerek görüntüyü temsil eden sınıflandırma vektörü elde edilir.



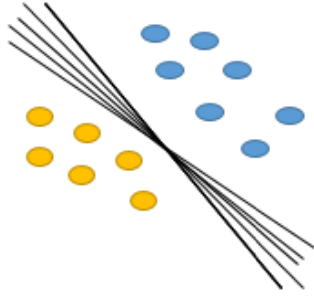
Şekil 1. GKÇ ile el görüntüsünün sınıflandırma vektörünün elde edilmesi

C. El Bölgesi Özelliklerinin Sınıflandırılması

El bölgesi takibi esnasında ilk olarak el bölgesi özellikleri çıkarılmakta, daha sonraki aşamada ise çıkarılan el bölgesi özellikleri sınıflandırılarak el bölgesinin hareketlerinin yorumlanması ile sanal mouse uygulaması gerçekleştirilmektedir. Yapılan çalışmada el bölgesi özelliklerinin sınıflandırılması için destek vektör makinaları kullanılmıştır.

DVM 1995 yılında Vapnik tarafından geliştirilmiş temeli istatistiksel öğrenmeye dayalı eğitilmiş sınıflandırma tekniklerinden biridir. Danışmanlı öğrenme temelli bu sınıflandırma tekniğinde verilerin aşırı öğrenmesinden

kaçınılması tahmin doğruluğunu maksimum yapmak amaçlanmaktadır. Eğitim seti üzerinde öğrenme yaparak test verisi üzerinde doğru tahmin yapmayı genelleştirmeyi amaçlayan bir makine öğrenmesidir [11]. DVM sınıflandırıcılarında kullanılan girdi verileri doğrusal olarak ayrılabilir olduğunda verileri ayırabilen birçok sayıdaki doğru arasından marjini en yüksek olan doğruyu seçmek hedeflenmektedir. Kullanılan verilen doğrusal olarak ayrılmıyor ise, veriler doğrusal olmayan haritalama kullanılarak yüksek boyuta çevrilmekte ve yeni boyutta marjini yüksek olan doğruyu bulmak amaçlanmaktadır [12-13].



Şekil 2. Doğrusal ayrılabilen veriler

Eğitim seti: $\{x_i, y_i\} \quad i = 1, 2, 3, \dots, L$

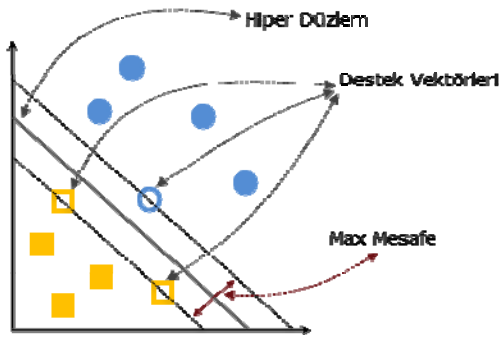
Sınıf etiketleri: $y_i \in \{-1, +1\}$

Ayırıcı düzlem: $\langle w, x \rangle + b = 0$

w : çoklu düzlemin normali (ağırlık vektörü)

b : bias

x : $\langle w, x \rangle + b = 0$: çoklu düzlem üzerindeki herhangi bir noktayı temsil etmektedir.



Şekil 3. İki doğrusal ayrılabilen sınıf aracılığıyla hiper düzlemin gösterimi

Kesikli çizgilerle gösterilen ve ayırıcı çoklu düzleme paralel olarak çizilmiş eşit uzaklıktaki iki çoklu düzlem

(doğru) arasındaki uzaklığa marjin denilmektedir. Bu kesikli çizgiler şu şekilde ifade edilmektedir:

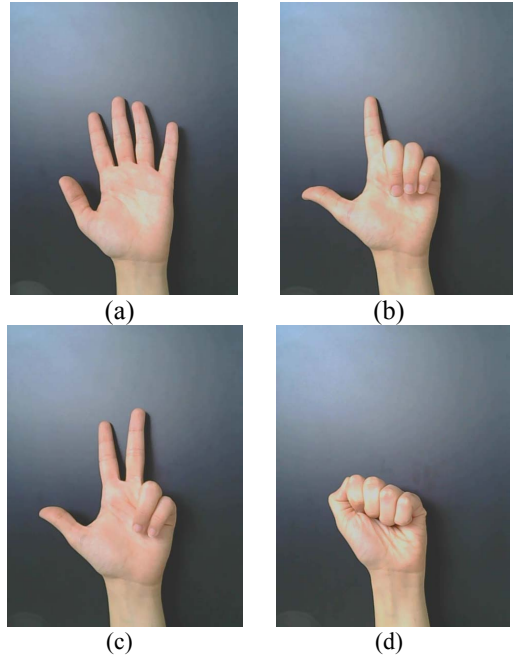
$$y_i = +1 \text{ için } \langle w, x_i \rangle + b = +1$$

$$y_i = -1 \text{ için } \langle w, x_i \rangle + b = -1$$

Karar fonksiyonu ise;

$$f(x) = (\sum \alpha_i y_i x_i \cdot x) + b$$

El hareketleri ile sanal mouse uygulaması amaçlayan bu çalışmada veritabanı olarak 60 farklı kişiden alınmış farklı el görüntüleri kullanılmıştır. Oluşturulan veri tabanında 4 farklı sınıfa ait görüntüler bulunmaktadır. Çalışmada sınıflar dört farklı mouse hareketini temsil etmektedir.



Şekil 4. Veritabanındaki 4 farklı sınıfa ait görüntüler. (a) Mouse serbest hareketi, (b) Mouse sol tıklama hareketi, (c) Mouse sağ tıklama hareketi, (d) Mouse taşıma hareketi

Şekil 4' de oluşturulan veri tabanında bir kişiye ait 4 farklı görüntü bulunmaktadır. Yapılan çalışmada ilk olarak bilgisayar kontrolünü yapacak kişinin kamera görüş alanında belirlenen hedef çerçeveye girmesi beklenmektedir. İlk eşleme ve mouse sabit konumdayken elin serbest dolaşmasında takibin devamlılığı için şekil 4 (a) görüntüsü kullanılmaktadır. Çalışmada gerçekleştirilen sistemde mouse tek tıklama için şekil 4 (b), mouse sağa tıklama için şekil 4 (c), mouse konumu değiştirmek için şekil 4 (d) 2D görüntüleri GKÇ ile 1D vektörlere dönüştürülerek hedef çerçeve sınıflandırması gerçekleştirilmektedir.

III. DENEYSEL DEĞERLENDİRMELER

Çalışma hedef gerçek zamanlı bir uygulama geliştirmek olduğundan insan gözünün saniyede 24 adet görüntüye kadar görüntü akışının sürekliliğindeki kopmaları algılayamadığını düşündüğümüzde seçilecek yöntemin saniyede en az 24 görüntüde özellik çıkarabilecek yetenekte olmalıdır. Yani gerçek zamanlı bir görüntü için hesaplama süresi 0,04 saniyeyi geçmemelidir. Bu değerler doğrultusunda gerçek zamanlı uygulama için deneysel sonuçlar Intel Core i7 4770K Soket 1150 3.5GHz 8MB Cache 22nm işlemci, 8GB DDR3 ram bulunduran bilgisayar ile elde edilmiştir.

640x480 piksel çözünürlüklü bir kamera için hedef çerçevenin görüntü çerçevesi boyutlarının yarısından büyük olamayacağı düşünüldüğünde, yapılan çalışmalarda Camshift algoritmasının ortalama model oturtma sürelerinin 0,007 saniyelerde ve DVM sınıflandırma sürelerinin de 0,002 saniyelerde olduğu görülmüştür. En fazla zaman alan sürecin GKÇ ile özellik çıkarma olduğu saptanmıştır.

Görüntü Boyutları	GKÇ Özellik Çıkarma Süreleri	
	168 Eğitim Görüntüsünde	72 Test Görüntüsünde
100x100	0,009 sn	0,007 sn
200x200	0,019 sn	0,018 sn
300x300	0,031 sn	0,029 sn

Tablo 1. GKÇ özellik çıkarma yöntemiyle elde edilen sonuçlar.

GKÇ ile özellik çıkarmada da sürenin 300x300 lük görüntüye kadar 0.04 sn nin altında olduğu görülmekte ve bu değerlerin gerçek zamanlı uygulama için yeterli olacağı bilinmektedir.

Ayrıca özelliklerin DVM ile sınıflandırılmasında 240 görüntülük veri tabanının rasgele seçilen %70(168 görüntü) eğitim ve %30(72 görüntü) test görüntüsü olarak seçilmiş ve elde edilen başarı istatistikleride irdelenmiştir.

Görüntü Boyutları	72 Adet Test Görüntüsünde DVM Başarısı	
	Doğru Tahmin Edilen Görüntü Sayısı	Başarı %
100x100	68	94,44%
200x200	70	97,22%
300x300	71	98,61%

Tablo 2. GKÇ özellik çıkarma yöntemiyle elde edilen sonuçlar.

IV. SONUÇ

Yapılan çalışmada el hareketleri ile bilgisayar kontrolünü gerçekleştirmek üzere sanal mouse uygulaması

geliştirilmiştir. Çalışmada öncelikle hedef çerçeveye giren kişinin Camshift algoritması kullanılarak el bölgesi takibi başlatılmakta ve takip esnasında hedef çerçeve tekrar güncellenmektedir. Eş zamanlı olarak takip için kullanılan hedef çerçeveden GKÇ ile özellik çıkarılarak 2D resimden 1D özellik vektörü elde edilmektedir. Daha sonra bu özellik vektörü destek vektör makineleri ile sınıflandırılarak veri tabanındaki 4 farklı mouse hareketini temsil eden sınıflardan birine izdüşürülmüştür.

Gerçekleştirilen çalışmada GKÇ ile özellik çıkarma süreleri ve DVM sınıflandırma başarısı deneysel olarak test edilerek gerçek zamanlı başarılı sonuçlar elde edilmiştir. Uygulamanın geliştirmeyi bekleyen adımları sınıflandırma sonucuna göre sistem çağrıları ile mouse konumlandırması ve çalışır sistemlere entegre edilmesidir.

KAYNAKÇA

- [1] Murthy R. S. G., Jadon R. S. A., "Review Of Vision Based Hand Gestures Recognition", International Journal of Information Technology, 2, 2 (2009) 405-410.
- [2] K. Sandeep, A. N. Rajagopalan, "Human Face Detection in Cluttered Color Images Using Skin Color and Edge Information", Indian Conference on Computer Vision, Graphics and Image Processing, Aralık 2002, Ahmedabad, Bildiriler Kitabı, 122-128.
- [3] Yao Y., Li C. T., "Hand Posture Recognition Using SURF with Adaptive Boosting", British Machine Vision Conference (BMVC), Eylül 2012, London, Bildiriler Kitabı, 50-60.
- [4] Denzler J., Paulus D. W., "Active Motion Detection and Object Tracking", International Conference on Image Processing (ICIP), Kasım 1994, Austin, Bildiriler Kitabı, 635-639.
- [5] Isard M. A., "Visual Motion Analyses by Probabilistic Propagation of Conditional Density", Ph.D. thesis, Robotics Research Group Department of Engineering Science, University of Oxford, Oxford, 1998.
- [6] Jiang L., Wang D., Cai Z., Yan X., "Survey of Improving Naive Bayes for Classification", Advanced Data Mining and Applications Lecture Notes in Computer Science, 4632, (2007) 134-145.
- [7] Asaari M. S. M., Suandi S. A., "Hand Gesture Tracking System Using Adaptive Kalman Filter", Intelligent systems design and applications (ISDA), International Conference on IEEE, Aralık 2010, Cairo, Bildiriler Kitabı, 166 -171.
- [8] Isard M., Blake A., "CONDENSATION-- conditional density propagation of visual tracking". International Journal of Computer Vision 29 (1): 5-28.
- [9] Zhou H., Xie L., Fang X., "Visual Mouse: SIFT Detection and PCA Recognition", Computational Intelligence and Security Workshops (CISW). International Conference on IEEE, Aralık 2007, Harbin, Bildiriler Kitabı, 263-266.
- [10] Dai J., Song W., Pei L., Zhang J., "Remote Sensing Image Matching via Harris Detector and SIFT Descriptor", Image and Signal Processing (CISP), International Conference on IEEE, Ekim 2010, Yantai, Bildiriler Kitabı, 2221-2224.
- [11] Burges C. J. C., "A Tutorial on Support Vector Machines for Pattern Recognition", Data Mining and Knowledge Discovery, 2, 2 (1998) 121-167.
- [12] Abe S., "Support Vector Machines for Pattern Classification", Second Edition, Springer, Kobe, 2010.
- [13] Byun H., Lee S. W., "Applications of Support Vector Machines for Pattern Recognition: A Survey", Pattern Recognition with Support Vector Machines, 2388, (2002) 213-236.