

Gerçek Zamanlı El Takibi ile Sanal Direksiyon Kullanma Gerçekleştirimi

Gamze MAFİZER ve Orhun DALABASMAZ

Hacettepe Üniversitesi Bilgisayar Mühendisliği Bölümü

Görüntü İşleme Grubu

Ankara, Türkiye

gamze_mafizer@hotmail.com, odalabasmaz@yahoo.com.tr

Özet

Bu makale, gerçek zamanlı görüntü işleme teknikleri kullanılarak geliştirilmiş bir proje örneğini sunmaktadır. Proje kapsamında, 2 boyutlu görüntüler işlenmiş olup, deri rengi ayırt edilerek el takibi yapılmıştır. Proje önce sabit görüntüler üzerinde çalıştırılmış, daha sonraki gelişim aşamasında gerçek zamanlı modellenmiştir. Gerçek zamanlı çalıştırıldığında donanımsal gerekliliklere ve güçlü bilgisayarlara ihtiyaç duyulmuştur. Proje, Intel i7 işlemcili ve yüksek donanıma sahip bilgisayarlarda geliştirilmiştir ve 640x480 görüntüler üzerinde çalışılmıştır.

1. Giriş

Proje, görüntü işleme konusu üzerine geliştirilmiştir. Proje kapsamında amaçlanan hedef, sayısal web kamerası kullanılarak gerçek zamanlı el takibi sistemi geliştirmek olmuştur. Hedef doğrultusunda takip edilecek ellerin, direksiyon simidini tuttuğu varsayılacak ve kullanıcının araba sürüyormuş gibi direksiyonu sola veya sağa döndürmesi beklenecektir.

2. Proje Geliştirme Araçları

Proje, MATLAB programlama dilinin sunduğu olanaklarla geliştirilmiş olup, IPT (Image Processing Tools) araçları kullanılarak tamamlanmıştır.

Projenin ilk evrelerinde gerçek zamanlı çalışılmamış ve yüksek hızda verim alınmıştır. Gerçek zamanlı çalışıldığında ise donanımsal gerekliliklere ve güçlü bilgisayarlara ihtiyaç duyulmuştur. Bu nedenle gerçek zamanlı çalıştırılacak projenin gerekli donanımlara sahip bir bilgisayarda denenmesi gerekmektedir.

3. Proje Gelişim Aşamaları

Proje başlangıcından itibaren, genel görüntü işleme teknikleri kullanılarak

başlanmış, aşama aşama geliştirilerek gerçek zamanlı çalışmaya sunulmuştur. Projenin genel ilerleyiş aşamaları şu şekilde olmuştur;

a. Başlangıç aşaması

Öncelikle, projeye başlamadan önce görüntü işleme ile ilgili temel kavramlar üzerinde çalışmalarda bulunuldu. Görüntünün temizlenmesi, elleri gösteren piksellerin gruplandırılması ve görüntü üzerinde doğrunun gösterilmesi gibi projenin çeşitli aşamalarında kullanılan temel fonksiyonların öğrenilmesi üzerine çalışıldı.

Görüntünün temizlenmesi aşamasında imdilate, imerode ve bwareaopen gibi fonksiyonlardan yararlanıldı. Böylelikle görüntüdeki gürültülerin temizlenmesi ve görüntüde sadece ellerin bırakılması mümkün hale geldi.

Projenin ilk zamanlarında öncelikli olarak araştırma yapıldı. Proje konusu kullanıcının sanal direksiyonu kullanması olduğundan ellerin tanınması öncelikli amaç haline geldi. Dolayısıyla öncelikli olarak el tanıma¹, el hareketi tanıma² ve yüz tanıma³ konuları araştırıldı ve ilgili makaleler incelendi (Bkz. kaynaklar).

El tanıma gibi konusunda değişik yaklaşımlar bulunmaktadır. Bu yaklaşımlar arasında geometrik özelliklerden yararlanmak, deri renginden yararlanmak ve örüntü kullanmak gösterilebilir. Yapılan araştırmalar sonucu en uygun yaklaşımın deri tanıma⁴ olacağı düşünülmüştür. Görüntüdeki kullanıcının elleri deri renginden faydalanılarak bulunmaktadır. Bunun için kullanımda bazı kısıtlar konulmuştur (Bkz. kısıtlar).

Yüz tanımda ise yine örüntüleri kullanmak, yüzün bileşenlerinin oranı gibi yaklaşımlar kullanılabilmektedir. Yüz tanıma konusunda yapılan araştırma sonucu uygulamanın görüntü üzerinde birden fazla işlem yapması yavaşlattığından kullanıcının

sadece ellerinin görüntüde olması bir kısıt olarak belirlenmiştir.

b. Deri tanıma

Deri tanıma konusunda pikselin RGB, HSV, YCbCR, TSL gibi değerleri üzerinden gidilebilmektedir.

RGB bir pikselin kırmızı(Red), yeşil(Green) ve mavi(Blue) değerlerini göstermektedir. Normalleştirilmiş RGB, RGB değerlerine normalleştirme işlemi uygulanarak yeni değerlerin elde edilmesidir. Bu değerler şu şekilde hesaplanabilir:

$$r = \frac{R}{R+G+B} \quad g = \frac{G}{R+G+B} \quad b = \frac{B}{R+G+B}$$

Bu üç değer toplamı $r+g+b = 1$ ettiğinden iki bileşen hesaplandıktan sonra üçüncüsü hesaplanabilir.

HSV ise bir pikselin renk tonu(Hue), doygunluk(Saturation) ve parlaklığı (Value) değerlerini temsil etmektedir. Bu değerler şu şekilde hesaplanır:

$$H = \arccos \frac{\frac{1}{2}((R-G) + (R-B))}{\sqrt{((R-G)^2 + (R-B)(G-B))}}$$

$$S = 1 - 3 \frac{\min(R, G, B)}{R+G+B}$$

$$V = \frac{1}{3}(R+G+B)$$

TSL bir pikselin ton(Tint), doygunluk(Saturation) ve açıklık (Lightness) değerlerini gösterir ve hesaplanması:

$$S = [9/5(r'^2 + g'^2)]^{1/2}$$

$$T = \begin{cases} \arctan(r'/g')/2\pi + 1/4, & g' > 0 \\ \arctan(r'/g')/2\pi + 3/4, & g' < 0 \\ 0, & g' = 0 \end{cases}$$

$$L = 0.299R + 0.587G + 0.114B$$

YCrCb lineer olmayan RGB değerlerini temsil eder ve şu şekilde hesaplanır:

$$Y = 0.299R + 0.587G + 0.114B$$

$$C_r = R - Y$$

$$C_b = B - Y$$

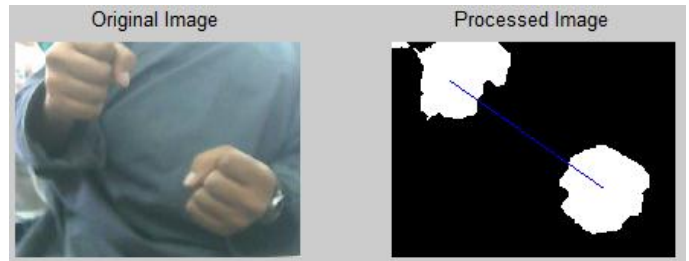
Bu değerlerin hesaplanmasından sonra belki eşik değerlerinin⁵ belirlenmesiyle derinin tanınması gerçekleştirilir. Eşik değerinin belirlenmesinde ortamdaki ışık, kullanıcın deri rengi gibi etkenler kullanılır.

Projede pikselin HSV değerleri kullanılmıştır. Sonra ortama göre eşik değerleri belirlenmiş ve bu eşik değerlerinin arasında kalan pikseller deri olarak tayin edilmiştir.

Deri tanıma hakkında daha ayrıntılı bilgi makaleler dizini altında bulunabilir. (bkz. Kaynaklar/makale 1)

c. Tek görüntü üzerinde çalışma

Öncelikle görüntüler üzerinde çalışıldı ve görüntü işleme teknikleri kullanılarak ellerin görüntü üzerinde tespiti ve gruplandırılması yapıldı. HSV renk değerleri kullanarak görüntü üzerinde ten renginin tanınması üzerine çalışıldı. Belirlenen eşik değerleri arasında kalan pikseller deri olarak tayin edildi ve görüntüde deri olarak belirlenen pikseller beyazla diğer kısımlarsa siyah olarak gösterilir. Bir sonraki adımda görüntüde kalan görüntüler imerode, imdilate ve bwareaopen fonksiyonları yardımıyla temizlendi. Daha sonra görüntüde kalan iki eli temsil eden pikseller bwlabel fonksiyonu aracılığıyla gruplandırılarak görüntü üzerinde iki adet el ortaya çıkarıldı.



Şekil 1. Tek görüntü üzerine çalışma

Tespit edilen iki adet el görüntüsü gruplandırılmış piksellerden oluştuğu için bu piksel değerlerinin maksimum ve minimum değerleri kullanılarak ellerin merkez noktaları hesaplandı ve iki adet noktaya indirgendi. Bu noktalar arasında doğru çizerek, el hareketine bağlı olarak çizginin değişimini gözlemledik. Ellerin konumunun ve gösterdiği yönün çizilen doğruyla uyumlu olduğunu doğruladık. Bu sayede direksiyon tutuyormuş gibi davranan kullanıcının, direksiyon çevirme hareketinin bilgisayar tarafından anlaşılması gerçekleştirilmiş oldu.

d. Görüntü dizisi üzerinde çalışma

Tek görüntü üzerine gerçekleştirilen bu adımlar, daha sonra birçok benzer görüntünün arka arkaya işlenmesiyle bir video gibi

çalıştırıldı. Öncelikle direksiyon tutuyormuş gibi davranan kişinin direksiyonu sağa sola hareket ettiriyormuş gibi yaptığı görüntüler alındı. Bu görüntüler tek tek ilgili yordama verilerek görüntülerin işlenmesi sağlandı. İşlenen görüntüler öncelikle HSV değerlerine çevrildi. Daha sonra temel görüntü işleme yordamlarında geçirildi ve gürültülerden temizlenerek sadece iki elin kümelendiği siyah beyaz görüntü elde edildi ve görüntü dizisi üzerinden ellerin konumu izlendi.

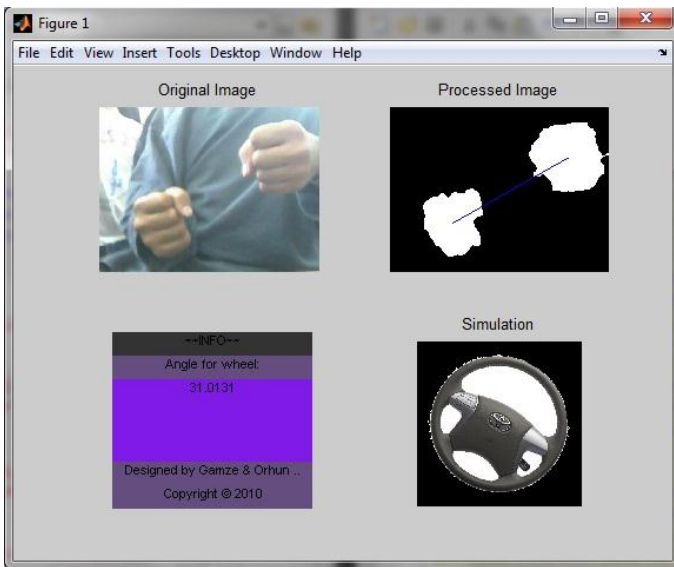
Ard arda gelen görüntülerde sağ ve sola dönme hareketleri yapılmıştır. Sonuçta işlenmiş olan görüntüde çizilen doğrunun takibiyle uyumlu olup olmadığı gözlenmiştir.

Bu adımın yapılmasının sebebi video üzerinde çalışmaya başlamadan görüntü dizileriyle basit bir şekilde video üzerinde çalışılıyormuş ortamı yaratmaktır.

Görüntü dizisi üzerine çalışmaların gerçekleştirilmesiyle, proje artık uygulamaya geçirilmeye hazır hale geldi ve ilgili adımlarla projenin gerçekleştirimi aşamasına geçildi ve projenin gerçekleştirimi sağlandı.

4. Projenin Gerçekleştirimi

Projenin gerçekleştirimi iki ayrı versiyondan oluşmaktadır. Birincisinde video verisi üzerinde kod işletilmiş, ikincisinde ise kameradan gerçek zamanlı görüntü alınarak program çalıştırılmıştır. Gelişim aşamalarının sonucunda elde edilmiş işletim dizgileri açıklanmıştır.



Şekil2. Projenin gerçekleştirimi

a. Video tabanlı işletim

Kısıtlarda belirtilen koşullar altında video kaydı alındı ve video üzerine çalışıldı. Kullanıcının direksiyon kullanıyormuş gibi davrandığı bir video çekildi ve bu video üzerine çalışıldı.

Video tabanlı işletim sırasında 'skin_detection' yordamı 'video.avi' parametresiyle çalıştırılır. Bu sırada, birçok insana ait deri renklerinin bulunduğu 'skin.jpg' adlı görüntü okunur. Bu renkler, görüntü üzerinde ellerin tespit edilmesi için kullanılacaktır. Okunulan deri renklerine ait görüntü HSV değerlerine dönüştürülür ve bir dizi işlemde geçirilir. Daha sonra taranacak olan H ve S değer aralıklarının bulunduğu diziler 'readVideo' adlı yordama parametre olarak gönderilir.

'readVideo' adlı yordam içerisinde, videodan kareler alındı ve tek tek 'hs_detectSkin' adlı yordama parametre olarak gönderildi.

'hs_detectSkin' adlı yordam işletilirken, görüntü alınmakta ve görüntü bir dizi işlemde geçirilmektedir. Görüntü öncelikle HSV değerlerine dönüştürülür, daha sonra yapı elemanı⁶ kullanılarak bir dizi görüntü işleme işlevinden⁷ geçirilir. Görüntü gürültülerden temizlendikten sonra, görüntüde iki adet kümelenmiş el silüeti kalıyor.

Daha sonra bu noktaların orta noktası hesaplanarak bu noktalar arasında çizgiler çekiliyor. Bu işlenmiş görüntü yine ekranda ilgili yerde görüntüleniyor. Bu çizgilerin eğimleri hesaplanıyor ve ilgili açı bilgileri yine ekranda gösteriliyor. Önceden yüklenmiş olan direksiyon görüntüsü ekranda gösteriliyor ve en son adımda ise hesaplanan açı doğrultusunda direksiyonun sağa ya da sola dönüşü sağlanıyor.

b. Gerçek tabanlı işletim

Video tabanlı çalıştırdıktan ve olumlu sonuç alındıktan sonra, proje gerçek zamanlı çalıştırılmak üzere geliştirilmeye başlandı.

Gerçek zamanlı işletim sırasında ise 'skin_detection.m' kütüğü çalıştırılır. Video tabanlı işletimden farklı olarak, videodan kesit olarak alınan kareler yerine gerçek zamanlı olarak kameradan alınan görüntüler kullanılmaktadır. 'readVideo' adlı yordamda kamera kullanıma hazır hale getirilerek

görüntü alınmaya başlanır. Saniyede kameradan 3 veya 5 görüntü alınarak bu görüntüler öncelikle ekranda görüntülenmekte, daha sonra işlem yapılmak üzere 'hs_detectSkin' adlı yordam çağrılmaktadır.

Bu adımda video ile çalışan adımla aynı işlem aşamalarına ve arayüze sahiptir. 'skin_detection.m' içinde belirlenen eşik değerleri 'readVideo' adlı yordamdan gönderilen görüntüler üzerinde kullanılır ve 'hs_detectSkin' yordamında ellerin belirlenmesi, açının hesaplanması son olarak direksiyon görüntüsünün açığa göre döndürülmesi gerçekleştirilir.

Gerçek zamanlı işletim olduğundan dolayı işletim sırasında bir yavaşlama ve işlenen görüntülerin gösterilmesinde bir gecikme yaşanmaktadır.

5. Kısıtlar

Pentium IV ve üzeri işlemci ve 2GB belleğe sahip olan herhangi bir bilgisayarda program çalıştırılabilir. Program daha düşük donanıma sahip bilgisayarlarda da denenebilir fakat en iyi verimi alabilmek için gerekli sistem özellikleri desteklenmelidir.

Video tabanlı işletim için;

- 'avi' formatında 80 saniyeden uzun olmayan video kayıtları kullanılmalıdır.
- Alınan görüntü sıkıştırılmamış olmalıdır.

Gerçek zamanlı işletim için;

- 1.3 MP kamera ile görüntü alınmalıdır.

Görüntüler;

- İyi aydınlatılmış bir ortamda alınmalıdır.
- Görüntüde yalnızca bir kişi yer almalıdır.
- Görüntüde yer alan kişinin uzun kollu giymiş olması varsayılmıştır.
- Kişi üzerinden alınan görüntü karesi, kişinin boynundan alt tarafı kapsamalıdır ve mutlaka her iki eli de görüntü dahilinde yer almalıdır. Yani görüntüde başı olmayan, sadece iki eli görünen bir kişi silueti beklenmektedir.

6. Kaynaklar

- Web tabanlı kaynaklar
 - www.google.com
 - www.mathworks.com

- http://www.urwelcome.us/Jstill/Matlab/Skin_Detection.htm

• Yazılı kaynaklar

- **A survey on pixel-based skin color detection**
Vladimir Vezhnevets,
Vassili Sazonov,
Alla Andreeva
Moscow State University
- **Improved Automatic Skin Detection in Color Images**
Filipe Tomaz
Tiago Candeias
Hamid Shahbazzkia
Universidade do Algarve

Ek1 – Kullanılan yordamların işlevleri

imerode: görüntü üzerinde aşındırma işlevi görür

imdilate: görüntü üzerindeki boşlukları bir yapı elemanı yardımıyla beyazla doldurur

bwlabel: görüntü üzerinde oluşan beyaz renkli her bir parçacığı numaralandırır

rgb2hsv: rgb değerine sahip görüntüyü hsv değerlerine dönüştürür

strel: bir geometrik şekile sahip yapı elemanı oluşturur

imfill: ortadaki boşlukları beyazla doldurur

bwareaopen: belli bir pikselden az yapıdaki görüntüler atılır

aviread: video okumaya yarar

Ek2 - Açıklamalar

¹ **el tanıma:** hand recognition

² **el hareketi tanıma:** hand gesture recognition

³ **yüz tanıma:** face recognition

⁴ **deri tanıma:** skin detection

⁵ **eşik değeri:** threshold

⁶ **yapı elemanı:** structural element (strel)

⁷ **görüntü işleme işlevleri:** imerode, imdilate, imfill..



Şekil3.
Simülasyon
için kullanılan
direksiyon
görüntüsü