

Nesne Takibi Yapan Robot Uygulaması (RasPiBot) Object Tracking Robot Application (RasPiBot)

¹Mehmet Enes Çoban, ²Burakhan Çubukçu and ^{*1}Uğur Yüzgeç
^{*1}Bilecik Şeyh Edebali Üniversitesi, Mühendislik Fakültesi, Bilgisayar Mühendisliği Bölümü, Bilecik, Türkiye
² Bilecik Şeyh Edebali Üniversitesi, Bilgi İşlem Dairesi Başkanlığı, Bilecik, Türkiye

Özet

Robotların ve otonom araçların çevrelerini iyi tanımaları, etraflarındaki nesnelere anlamlandırabilmeleri ve onları takip edebilmeleri önemlidir. Bir robot takip etmesi gereken bir nesneyi kaybettiğinde ne yapacağını bilemez. Bu uygulamalarda büyük sorunlara yol açar ve robotun çalışmamasına neden olur. Bu çalışmada, hareketli robotun renkli nesneyi takip edebilmesi hedeflenmiştir. Bu amaçla, Raspberry Pi 3 minibilgisayar ve mobil robot platformunda bir kamera kullanılmıştır. Raspberry Pi3'teki gömülü görüntü işleme, SimpleCV açık kaynaklı çerçeve kullanılmıştır. Robotun önündeki kamera sayesinde robot çevresini tespit eder ve nesnenin hareketine ve konumuna göre hareket eder.

Anahtar Kelimeler: Raspberry Pi, Otonom Araç, Robot, Görüntü İşleme.

Abstract

It is important for robots and autonomous vehicles to know their environment well, to make sense of the objects around them and to follow them. A robot doesn't know what to do when it loses an object that it has to follow. This leads to major problems in applications and causes the robot to stop functioning. In this study, it is aimed that the moving robot will be able to follow the colored object. For this purpose, Raspberry Pi 3 minicomputer and a camera were utilized on the mobile robot platform. For the embedded image processing on Raspberry Pi3, the SimpleCV open source framework was used. Thanks to the camera in front of the robot, the robot detects its surroundings and moves according to the movement and position of the object.

Key words: Raspberry Pi, Autonomous Vehicle, Robot, Image Processing.

1. Giriş

Günümüz teknolojinin gelişmesiyle beraber robotlar ve otonom araçlara yönelik çalışmalar da artmıştır. Otonom araçlar içerisinde bulundukları otomatik kontrol sistemleri sayesinde bir sürücüye ihtiyaç duymadan yolu ve çevresini algılayarak sürücü müdahalesi olmadan seyir edebilen araçlardır. Otonom araçlar radar, GPS, odyometre, bilgisayar görüşü gibi teknolojiler ve teknikler kullanarak çevresindeki nesnelere algılayabilmektedir [1]. Geleceğe yön verebilecek kapasiteye sahip bu teknoloji ABD, Japonya, İngiltere gibi ülkeler tarafından aktif olarak kullanılmaktadır ve desteklenmektedir [2].

Günümüzde insanların devamlı hareket halinde olması nedeniyle sabit robotların yeteri kadar faydalı olmadığı anlaşılmıştır ve bu nedenle hareket halinde olan robot araçlar geliştirilmeye başlanmıştır. Bununla beraber hareketli robotların çevrelerini tanıma, onlarla etkileşime girme ihtiyaçları da doğmuştur.

*Sorumlu yazar: Adres: Bilecik Şeyh Edebali Üniversitesi, Bilgisayar Mühendisliği Bölümü, 11210, Gülümbe Yerleşkesi, Bilecik. E-mail adresi: ugur.yuzgec@bilecik.edu.tr, Tel: +902282141555

İnsanın çevresini görme duyusuyla tanınması, hareketli robotların da çevrelerini sensörler yerine bir görüş mekanizmasıyla tanınması fikrinin önünü açmıştır. Bu görüş mekanizmasını sağlayan kameralar sayesinde farklı sensörlere gerek olmadan nesnelerin uzaklığı, şekli, boyutu, rengi gibi bilgiler edinilebilmektedir. Bu sebepten dolayı hareketli robotlarda kamera kullanılması avantajlı bir durum haline gelmiştir [3].

Literatürde bilgisayarlı görü ile robot uygulamaları alanında pek çok çalışmaya rastlanmaktadır. Bunlardan bazıları burada özetlenmiştir. Gregor Klančar ve arkadaşları, hızlı ve dinamik takip gerçekleştirilmesi gereken bir oyunda Bayer biçimli görüntü dosyası ve piksel sınıflandırma yöntemi kullanarak robotların futbol oynamasını sağlayan bir çalışma yapmıştır [4]. Sanghoon Kim ve arkadaşları, çeşitli sensörler kullanarak nesne takibi yapabilen bir robot gerçekleştirmişlerdir. Engellerden kaçınmak için kızılötesi ve ultrasonik sensörler kullanmışlardır [5]. Mustafa Serter Uzuner ve arkadaşları ise görme tabanlı bir robot ile farklı renklerdeki nesnelerin gerçek zamanlı takibini gerçekleştirmişlerdir. Farklı renklerde nesnelerin takibini yapabilen robot, gerçek zamanlı işleyişe uygun olacak şekilde 96ms ile 106ms aralığında bir tepki verme süresine sahiptir. Mavi ve kırmızı renkli nesnelerin %100, yeşil renkli nesnelerin ise %60'lık bir tanınma başarısına sahip olduğu görülmüştür [6]. Yüz tanıma, dışarıdan bireye herhangi bir müdahale olmadan kimliğini doğrulama ve modellemeye imkân sağlayan bir yöntemdir. Sözsüz iletişim açısından vücudumuzda en fazla veriyi içeren bölüm insan yüzüdür. Aynı zamanda insan yüzünde oluşan mimikler sayesinde insanlar arası iletişimdeki gibi, insan-makine etkileşiminde de değerli bilgiler sunmaktadır. Örnek vermek gerekirse, insan yüzlerine dair veri setleri bilgisayar donanım ve yazılımı ile işlenerek uzman sistemler olarak insan duygularının tespit edilmesi ve analizinde kullanılabilir. [7].

Bu çalışmada, ön tarafına yerleştirilmiş bir kamera ile önceden belirlenen bir nesnenin takibini yapan bir robot uygulaması gerçekleştirilmiştir. Bu robot; güvenlik, keşif ve gerçek zamanlı takip uygulamalarında kullanılabilir özelliklere sahiptir. Bu robotun, herhangi bir sensör kullanmadan sadece görüntü işleme ve bilgisayarlı görme tekniklerini kullanarak nesnelere takip etmesi sağlanmıştır. Aracın kontrolü Raspberry Pi mini bilgisayarı ile sağlanmıştır ve gerekli kodlar Python dilinde yazılmıştır. Sistemin çalışmasını test etmek amacıyla çeşitli ortamlarda ve arka planlarda hareketli yeşil topun takibinin yapılması sağlanmıştır.

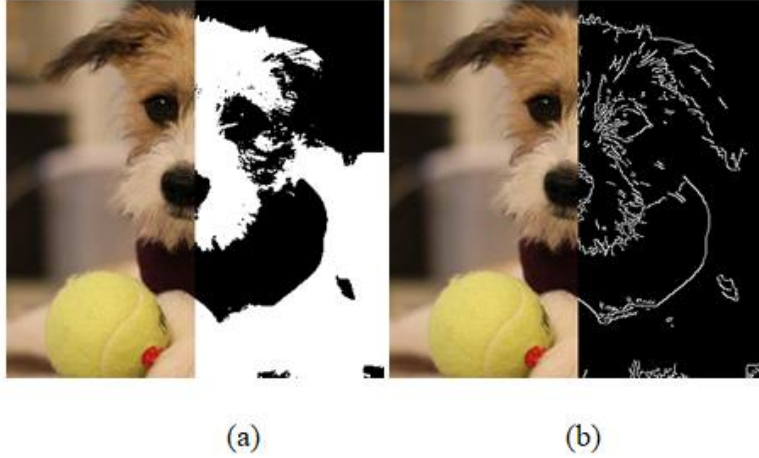
2. Kullanılan Yazılımlar ve Donanımlar

Bu çalışmada donanım olarak bir Raspberry Pi 3 mini bilgisayarı, bir mobil robot platformu, bir web kamerası ve güç için bataryalar kullanılmıştır. Düşük güç tüketimine sahip bu mini bilgisayarlar medya oynatıcı, web sunucu, IoT gibi alanlarda kullanılabilir. Üzerinde soğutma fanları olmadığı için çok sessiz çalışmaktadır [7]. HDMI bağlantısı ile istenildiğinde bir monitöre bağlanabilir, USB portları sayesinde klavye ve fare gibi çeşitli USB birimleri bağlanabilmektedir. Raspberry Pi3 mini bilgisayarının genel teknik özellikleri aşağıda verilmektedir [8]:

- 64-bit quad-core ARMV8 işlemci
- 1.2GHz çalışma frekansı
- 1GB RAM
- Dahili WiFi
- Bluetooth 4.1

- 40 Adet GPIO, 4 Adet USB girişi, 4 uçlu Stereo çıkışı ve video çıkışı
- Full HDMI
- Raspberry Pi Kamera bağlantısı için CSI kamera portu
- Raspberry Pi 7" dokunmatik ekran için DSI ekran portu
- Micro SD soketi

Bu çalışmada en önemli aşamalardan birisi de görüntü işleme kısmıdır ve bu yüzden gömülü sistemlerde sıklıkla kullanılan bir açık kaynak kütüphanesi (SimpleCV) tercih edilmiştir. SimpleCV, bilgisayarlı görme işlemleri gerçekleştirebilmeye yardımcı olan bir açık kaynak kütüphanesidir [9]. Görüntü işlemede popüler bir kütüphane olan OpenCV'nin karmaşık yapısına karşılık daha basit ve kullanıma hazır fonksiyonlar sayesinde gömülü sistemler üzerinde görüntü işlemek SimpleCV Kütüphanesi ile oldukça kolaydır. SimpleCV Python dilinde yazılmış bir kütüphanedir ve Linux, Windows ve Mac gibi farklı işletim sistemlerine sahip platformlarda çalışabilmektedir. SimpleCV Kütüphanesi ile görüntü işlemenin yanı sıra video işleme uygulamaları da yapılabilmektedir. Şekil 1'de SimpleCV Kütüphanesindeki bazı metod örnekleri verilmiştir. Görüntülere Şekil 1.a'da threshold fonksiyonu, Şekil 1.b'de ise kenar bulma metodu uygulanmıştır [10].



Şekil 1. SimpleCV Kütüphanesindeki bazı metod örnekleri, (a) eşik seviyesi fonksiyonu, (b) kenar bulma fonksiyonu

Bilgisayarın bir görüntüdeki nesnelere bulabilmesi için nesnelere birbirinden ayrılması gerekir. Bu ayrılan nesnenin daire mi, dikdörtgen mi, insan mı, araba mı olduğunu değerlendirmek ise sonraki işlemdir. Bu işleme Blob bulmak denir. Blob kelimesi, bazı ortak özellikleri paylaşan bir resimdeki bağlı piksel grubu (örneğin gri tonlama değeri) olan Binary Large Object kelimesinden gelir. Large terimi, bir görüntüde yalnızca belirli boyuttaki nesnelere fark edileceğini belirtirken, geri kalanı "gürültü" olarak tanımlanır ve yöntem bunu görmezden gelir. Blob tespitinin amacı, bu belirli bölgeleri tanımlamak ve işaretlemektir.

Blob analizinde üç ana yöntem vardır: çıkarma, temsil ve sınıflandırılma. Çıkarma işlemi, birbirine bağlı piksellerin bulunmasıyla başlar ve bu komşu piksellerin önceden belirlenmiş özelliklere uyduğu anlamına gelir. Bir sonraki adımda, daha önce çıkarılan piksellerin özellikleri temsil edilir. Algılamadaki son adım, aranan tüm özelliklerle eşleştiğini doğrulamak için daha önce bir prototip ile tanımlanan gösterimi karşılaştırmaktır.

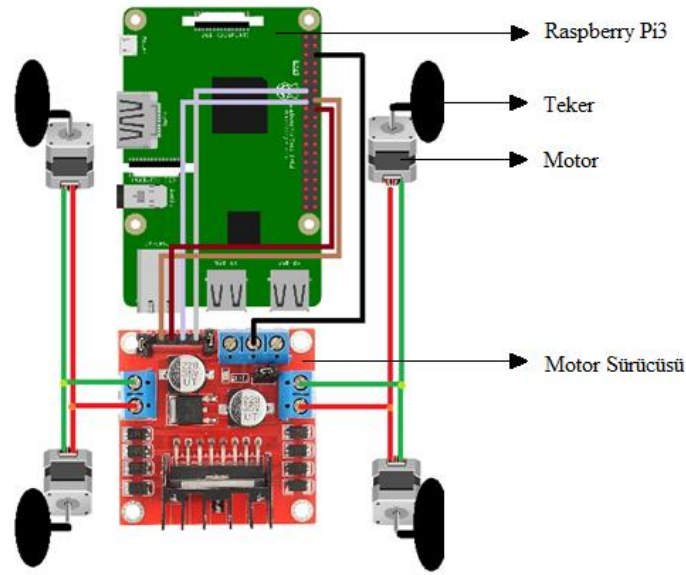
Bu çalışmada kullanılan SimpleCV Kütüphanesi fonksiyonları aşağıda verilmektedir.

- *class SimpleCV.Camera.Camera(cameraindex=0)*: Camera sınıfı, kameradan görüntü almaya yarar. Kamera etkinleştirildiği zaman başka bir işlem tarafından kullanılamaz.
- *class SimpleCV.Display.Display(resolution=(640, 480), flags=0, title='SimpleCV')*: WindowStream, görüntünün gösterildiği bir ekran açar. Varsayılan çözünürlük 480p'dir.
- *class SimpleCV.ImageClass.Image(source=None, camera=None, colorSpace=0)*: Image sınıfı kaynak türlerini kolaylıkla birbirine dönüştürmeyi sağlar. Görüntü 8-bitlik 3 kanallı RGB görüntülerine dönüştürülür.
- *colorDistance(color=(0, 0, 0))*: Image sınıfı içinde yer alan bu işlem, her piksel renginin verilen renk değerine göre uzaklığını gösteren bir görüntü döndürür. Renk değerine uzak olan renkler daha açık renkte, hedef renge daha yakın olan renkler ise daha koyu bir renkte temsil edilir.
- *stretch(threshlow=0, threshhigh=255)*: Image sınıfı içinde yer alan bu işlem, alt ve üst eşik değerini alarak filtreleme yapar. Alt eşik altındaki herhangi bir şey siyaha (0) ve üst eşik üstündeki herhangi bir şey beyaza (255) dönüştürülür.
- *binarize(thresh=-1, maxv=255, blocksize=0, p=5)*: Image sınıfı içinde yer alan bu işlem, ikili eşik değeri uygulayarak eşik değerinden yüksek değerleri maxv değerine, düşük değerleri ise siyaha dönüştürür.
- *erode(iterations=1)*: Image sınıfı içinde yer alan bu işlemde görüntüye morfolojik erozyon yapılır. Küçük gürültüler kaldırılır ve bulunan Blob'lar pürüzsüzleştirilir. Erozyon işlemi minik noktaları tespit etmekte çok etkilidir. Kernel, görüntüyü hareket ettirerek kernel içindeki en küçük değeri alır. İterasyon değeri işlemin kaç kez uygulanacağını ifade eder.

3. Nesne Takibi Yapan Robot Uygulaması

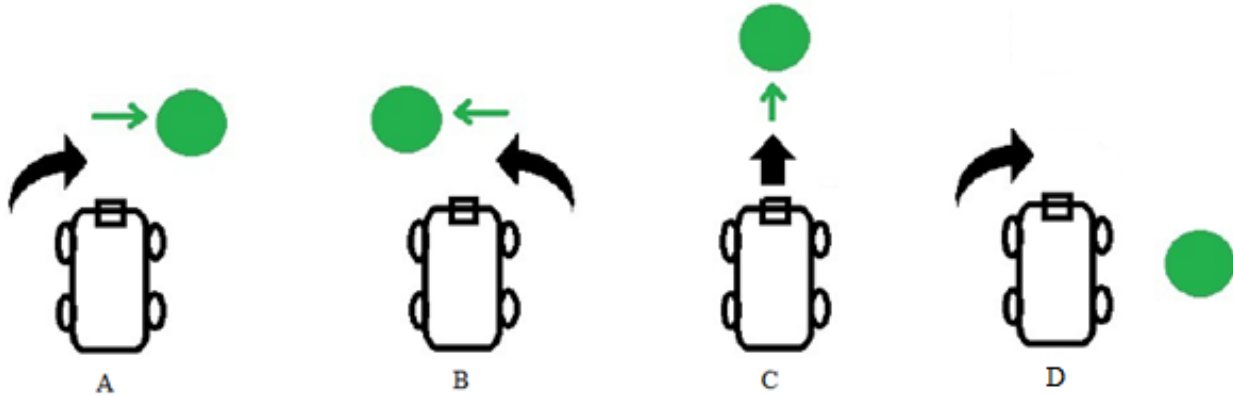
Bu çalışma kapsamında önceden belirlenen yuvarlak ve yeşil renkli bir nesnenin takibini yapan bir robot gerçekleştirilmiştir. Robot üzerinde dört motor, dört tekerlek ve bir de motor sürücü (L298N) kartından oluşmaktadır. Motorların gerekli akımı çekebilmesi için harici bir Li-Po batarya kullanılmıştır. Raspberry Pi3 mini bilgisayarının beslemesi için ise ayrı bir güç kaynağı tercih edilmiştir. Şekil 2'de motorların, motor sürücü devreye ve Raspberry Pi3 mini bilgisayarına bağlantısı gösterilmiştir.

Robot, ilk çalıştığında sağa dönerek yeşil yuvarlak nesneyi aramaya başlar. Yeşil bir daire görene kadar bu dönüş işlemini gerçekleştirir. Yeşil bir daire gördükten sonra ise dönüşünü sonlandırır ve yeşil daireye doğru ilerlemeye başlar. Nesneye belirli bir mesafe yaklaştıktan sonra durur ve nesnenin hareket etmesini bekler. Nesne hareket ettikten sonra nesnenin hareketine göre robot takibe başlar (Şekil 3).



Şekil 2. Raspberry Pi3, motor sürücüsü ve motorların bağlantısı.

Eğer görüntü merkezindeki nesne sağa/sola hareket ederse robotta aynı şekilde nesneyi görüntünün merkezine tekrar yerleştirmek için sağa/sola hareket edecektir. Eğer nesne görüntü merkezinden uzaklaşır ise yuvarlak nesnenin çap bilgisi azalacağından robot ileri doğru hareket ederek nesneyi takip edecektir. Robot aldığı görüntü üzerinde nesneyi tespit ettiği sürece bu şekilde sürekli olarak nesneyi takip ederek hareketine devam eder. Nesnenin hızlı hareketi halinde robotun üzerindeki kameradan alınan görüntülerde nesne tespit edilemediğinde RasPiBot sağa dönerek nesneyi tekrar aramaya başlar.

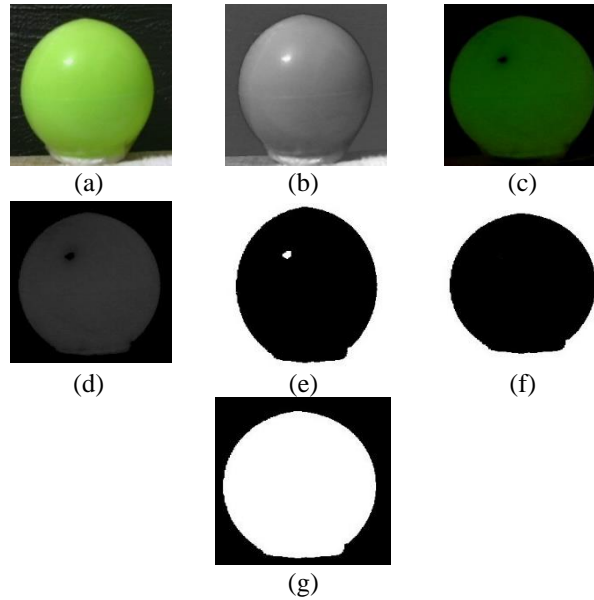


Şekil 3. Robotun nesnenin konum değişikliğine göre yaptığı hareketleri.

Uygulamada nesnenin takip edilebilmesi ve görüntüdeki nesnelerin net seçilebilmesi için kameranın hareketsiz veya hareket hızının çok düşük olması gerekir. Bunun için robotun hareketlerini kontrol etmek amacıyla Raspberry Pi3 mini bilgisayarının yazılımsal PWM özelliği kullanılmıştır. Bu sayede robotun daha yavaş hareket etmesi ve kameradan alınan görüntünün daha net olması sağlanmıştır. İleri yönde PWM doluluk oranı %20, sağa ve sola dönmek için ise

%35 olarak kullanılmıştır. Robotun dönüşlerinde daha fazla güç verilmesinin sebebi sağa/sola dönerken daha fazla güce ihtiyaç duymasından dolayıdır. Yeşil yuvarlak bir top olarak belirlenen hedef nesnenin tespit edilebilmesi için SimpleCV Kütüphanesi kullanılmıştır. Nesneyi tespit edebilmek için aşağıdaki adımlar uygulanmaktadır. Şekil 4'de bu adımların ekran görüntüleri verilmiştir. Şekil 5'de gerçekleştirilen robotun bir fotoğrafı gösterilmiştir.

- Görüntüyü al
- Görüntüyü yeşil renk değerlerine göre düzenle
- İlk görüntü ile düzenlenen görüntü arasındaki farkları bul
- Görüntüyü grileştir
- İkili formata dönüştür
- Görüntüdeki gürültüleri kaldır
- Görüntünün renk değerleri tersine çevir



Şekil 4. Görüntünün elde edilmesi işlemleri



Şekil 5. Gerçekleştirilen robotun (RasPiBot) fotoğrafı

Şekil 6'da nesne takibi yapan robotun akış diyagramı gösterilmiştir.



Şekil 6. RasPiBot uygulamasının akış diyagramı

4. Sonuçlar ve Tartışma

Robotlar ve otonom araçlar çevrelerini tanımak için çok çeşitli sensörlere ihtiyaç duymaktadır. Kameralar, robotların ihtiyaç duyduğu sensör verilerini tek başlarına sağlayabilir. Kamera ve görüntü işleme sayesinde bir nesnenin uzaklığı, boyutu, büyüklüğü, rengi gibi çeşitli bilgileri elde edilebilir. Ardından bu nesne, kamera açısından ani bir şekilde çıkmadığı sürece çeşitli algoritmalar yardımıyla takip edilebilir [11,12].

Bu çalışmada, Raspberry Pi3 mini bilgisayarı üzerinde gömülü kodları Python dilinde yazılmış bir robotun çevresindeki yeşil renkli bir yuvarlak topu bulması ve onun hareketlerini takip etmesi sağlanmıştır. Robot, yuvarlağın uzaklığına göre nesneye yaklaşmakta ve nesnenin konumuna göre sağa veya sola dönüş yaparak nesneyi görüş açısından kaybetmemeye çalışır. Robotun ve nesnenin bulunduğu ortamın ışık miktarına bağlı olarak nesnenin tespiti zorlaşmaktadır. Nesnenin çok hızlı ve ani hareketlerinde ise robotun nesneyi takip etmesinde problemler ortaya çıkmaktadır. Fakat sabit ve yeteri kadar aydınlatılmış bir ortamdan alınan görüntülerde %90'a yakın başarı oranı ile yeşil nesne tespit ve takip edilebilmektedir.

Yapılan bu çalışma, fabrikalarda benzer özelliklere sahip ürünlerin belirli alana taşınmasını sağlamak için nesne tespitinde kullanılabilir. Mobil robota eklenecek ekstra düzenekler ile ürünlerin taşınması sağlanabilir. Bir spor karşılaşmasında sahadaki topun takibini sağlamak amacıyla kameraların topun hareketine göre hareket etmesi sağlanabilir veya topun saha dışına çıkıp çıkmadığı bu kameralar aracılığıyla tespit edilebilir. Saha dışına çıkan toplar, robotun yardımıyla toplanarak toplama alanına bırakılabilir.

Gelecek çalışmalarda, robotun hedef nesneye yaklaştıktan sonra onunla etkileşime girmesi, örneğin nesneyi alarak başka bir konuma taşınması veya robotlar arasındaki bir futbol karşılaşmasında top ile şut çekmesi sağlanabilir.

Teşekkür

Bu çalışmada sunulan “*Nesne Takibi Yapan Robot Uygulaması (RasPiBot)*” isimli bildiri çalışması, Bilecik Şeyh Edebali Üniversitesi Bilgisayar Mühendisliği Bölümü laboratuvarlarında, Bilgisayar Mühendisliği Tasarım Çalışması dersi kapsamında gerçekleştirilmiştir.

Kaynakça

- [1] <http://www.endustri40.com/surucusuz-otonom-araclar/> (11 Mayıs 2018)
- [2] <https://www.donanimhaber.com/otomobil-teknolojileri/haberleri/ingiltere-butcesinden-otonom-ve-elektrikli-araclara-destek.htm> (11 Mayıs 2018)
- [3] N. Yılmaz, Ş. Sağıroğlu, M. Bayrak. “Genel amaçlı web tabanlı mobil robot: SUNAR”, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 21(4), 2013.
- [4] G. Klančar, M. Brezak, D. Matko, I. Petrović, “Mobile robots tracking using computer vision”, *Automatika: časopis za automatiku, mjerenje, elektroniku, računarstvo i komunikacije*, 46(3-4), 155-163, 2005.
- [5] S. Kim, S. Lee, S. Kim, “Object tracking of mobile robot using moving color and shape information for the aged walking”, Published in: Second International Conference Future Generation Communication and Networking, 2008. FGNC '08.
- [6] M. S. Uzuner, N. Yılmaz, M. Bayrak. “Görme tabanlı mobil robot ile farklı renklerde nesnelerin gerçek zamanlı takibi”, Gazi Üniversitesi Mühendislik-Mimarlık Fakültesi Dergisi, 25(4), 2013.
- [7] <http://www.elektrikport.com/teknik-kutuphane/raspberry-pi-nedir-arduino-ile-farklari-nelerdir-/8305#ad-image-0> (12 Mayıs 2018)
- [8] <http://market.samm.com/raspberry-pi-3> (12 Mayıs 2018)
- [9] <http://ammaraslan.blogspot.com.tr/2015/06/simplecv.html> (12 Mayıs 2018)
- [10] <http://simplecv.org/> (12 Mayıs 2018)
- [11] N.T. Siebel, S. Maybank, “Fusion of multiple tracking algorithms for robust people tracking”, In: Heyden A., Sparr G., Nielsen M., Johansen P. (eds) *Computer Vision: ECCV 2002. Lecture Notes in Computer Science*, vol 2353. Springer, Berlin, Heidelberg.
- [12] Y. Zhang, H. Hu, H. Zhou, “Study on adaptive Kalman filtering algorithms in human movement tracking”, Published in: *Information Acquisition*, 2005 IEEE International Conference.