

MATLAB Ortamında Trafik Levhalarının Algılanması ve Tanınması

Traffic Sign Detection and Recognition in MATLAB

Ozan BORAZAN, Kaan Aykut KABAKÇI, Mertkan AKKOÇ
Elektronik ve Haberleşme Mühendisliği Bölümü
İstanbul Teknik Üniversitesi
İstanbul, TÜRKİYE
{borazan,kabakcik,akkocm}@itu.edu.tr

Özetçe —Bu çalışmada sürücü kaynaklı dikkat hatalarını önlemek için onlara trafik levhalarını hatırlatmak veya insansız araçlarda trafik levhalarının tespit edilmesinde kullanılması amacıyla yöntem önerilmektedir. Önerilen yöntemde levhaların renklerine göre HSV renk uzayında İmge Bölgesinde Renklendirme(Blob Coloring) kullanılarak levha tespit edilebilmektedir. Bu tespit sonucunda veri tabanında bulunan levha görselleri ile karşılaştırma yapılarak levha tanınabilmektedir.

Anahtar Kelimeler—Trafik levhaları, HSV, imge bölgesinde renklendirme, tespit, karşılaştırma, tanıma.

Abstract—This paper presents reminding drivers traffic signs to prevent attention mistakes of them and a method to recognize of traffic signs at unmanned vehicle. Traffic signs can be determined using Blob Coloring at HSV color-space in this presented method. Then traffic signs can be recognized by comparing with traffic signs images found at database.

Keywords—Traffic signs, HSV, blob coloring, detection, recognition, comparison.

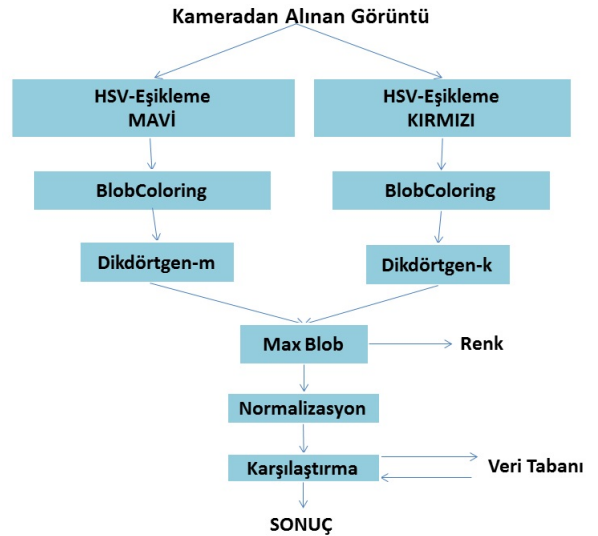
I. GİRİŞ

Trafik levhalarının tespit edilmesi ve tanınması hem sürücünün dikkat hatalarından kaynaklanan kazaların önlenmesi hem de insansız araçlarda kullanılması açısından önemli rol oynar. Aynı zamanda bu işlemin doğru ve hızlı yapılabilmesi gerekmektedir. Önerilen yöntemin, doğru tespit yapabilmesi için trafik levhalarının temel özelliklerinden olan rengin ayrımı, hızlı olabilmesi için ise temel düzeydeki yöntemler göz önüne alınmıştır. Levhaların tespit edilmesinde kameradan alınan görüntünün HSV renk uzayında eşikleme yöntemi kullanılarak renk ayrımı yapılmaktadır [1]. Renk ayrımından sonra levhanın yerini tespit edebilmek için İmge Bölgesi Renklendirmesi yapılmaktadır; renklendirme işlemi ardından en büyük imge bölgesi seçimi yapılmakta ve bu bölge ile levhanın yeri belirlenmektedir [2]. Levhanın tanınması sürecinde ise tespit edilen bölgeyi orjinal görüntüden keserek siyah beyaza çevirip karşılaştırma işleminde görüntülerin boyut farkından kaynaklanan hataları önlemek için normalizasyon işlemi yapılmaktadır [3]. Karşılaştırma işleminde kullanılan veri tabanı da aynı şekilde kesme işleminin ardından siyah beyaza çevrilmiş görüntüyü normalizasyon yapılarak

kaydedilmesi ile oluşturulmuştur. Bu şekilde oluşturulmuş olan veri tabanındaki görüntülerin her biri ile karşılaştırma yapıp fark alındıktan sonra en küçük farka sahip görüntü sonuç olarak döndürülmektedir.



Şekil 1: Kullanılan Levha Çeşitleri



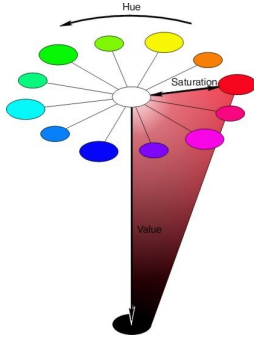
Şekil 2: Yöntem Algoritması

II. TRAFİK LEVHASI TESPİTİ

Bu aşamadaki amaç, kameradan alınan görüntüde levhanın yerinin doğru bir şekilde bulunmasıdır. Levhaların özelliklerini incelediğimizde belirgin özelliklerinden biri olan renk ayırımından yararlanılmış ve projenin temel düzeyde olması nedeniyle de sadece kırmızı ve mavi levhalar ele alınmıştır. Bilgisayar ortamında kaydedilen görüntülerde renk uzayı olarak RGB ve HSV uzayları bulunmaktadır. RGB uzayı ile yapılan tespitler, farklı ortamlardan daha fazla etkilendiği için HSV uzayının kullanılması tercih edilmiştir. Yapılan denemelerle kırmızı ve mavi renkleri için ayrı renk tonu(hue), canlılık(saturation) ve parlaklık(value) değerleri belirlenmiş ve eşikleme işlemi yapılmıştır. Eşikleme işlemi ile birlikte kırmızı ve mavi bölgeler tespit edildikten sonra imge bölgesinde renklendirme yöntemi kullanılarak en büyük kırmızı veya mavi bölge bulunup levhaların sınırlarının belirlenmesi yapılmış ve levha tespiti işlemi sonlandırılmıştır.

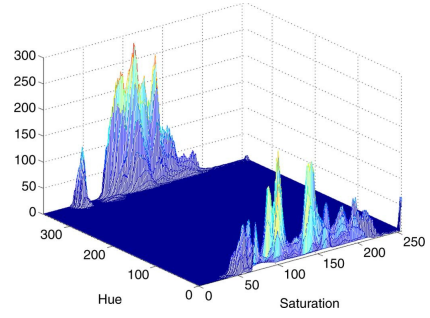
A. Renk Eşiklenmesi

Bilgisayar ortamında kaydedilen görüntüler RGB ve HSV renk uzayları bulunmaktadır. Kaydedilen görüntü bir matris olarak tanımlanmakta ve matris elemanlarının her biri ise piksel olarak isimlendirilmektedir. Bilgisayar ortamında görüntüyü üç farklı düzlem olarak düşünürsek RGB uzayında ilk düzlemde bulunan matris elemanları görüntünün kırmızı(red), ikinci düzlemde bulunan matris elemanları yeşil(green) ve üçüncüde ise mavi(blue) değerlerini içermektedir. HSV uzayında ise ilk düzlemde görüntünün renk tonu(hue), ikincide canlılık(saturation), üçüncüde ise parlaklık(value) bilgisi yer almaktadır.

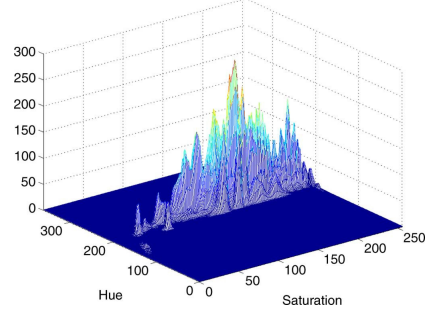


Şekil 3: Sabit renk tonu ile örnek HSV modeli [1]

RGB uzayında yapılacak olan tespit işleminde ortam aydınlatmasından kaynaklanan hatalar daha fazla olacağından HSV uzayı tercih edilmiştir. MATLAB ortamında renk tonu, canlılık ve parlaklık değerleri 0-1 arasında ölçeklendirilmiştir. Yapılan denemeler ile kırmızı ve mavi renkleri için seçilen değer aralıkları aşağıda verilmiştir.



Şekil 4: Kırmızının Renk tonu - Canlılık Histogramı [4]



Şekil 5: Mavinin Renk tonu - Canlılık Histogramı [4]

Kırmızı için

Renk tonu: <0.14 veya >0.91
Canlılık: $0.3 < ve <1$
Parlaklık: $0.4 < ve <1$

Mavi için

Renk tonu: $0.48 < ve <0.81$
Canlılık: $0.1 < ve <1$
Parlaklık: $0.4 < ve <0.96$

Yazılan 'hsvSınırlama' isimli MATLAB kodu yukarıda verilen değerleri kullanarak eşikleme işlemini yapıp kırmızı veya mavi bölgeleri tespit etmektedir.

B. Levha Ayırt Etme

Levhanın ayırt edilmesi işleminde imge bölgesinde renklendirme yöntemi kullanılmaktadır ve bu işlemin yapılabilmesi için görüntünün iki farklı renk içermesi gerekmektedir. Levhanın rengine bakılmaksızın kırmızı ve mavi için eşikleme yapılmaktadır. Kırmızı veya mavi bölgeler tespit edildikten sonra yazılan 'TemelSecici' isimli MATLAB kodu ile bulunan bölgeler beyaza, geri kalan bölgeler siyaha çevirilerek en büyük alana sahip bölge aranılan imge olup sınırları belirlenerek çerçeveye alınmaktadır. Bulunan imge aynı zamanda renk bilgisini içermektedir çünkü mavi levhada kırmızı, kırmızı levhada mavi en büyük bölgeyi oluşturamamaktadır. Belirlen sınırlardan yararlanılarak siyah beyaza çevrilen orjinal resimden levhanın yeri kesilerek yeniden boyutlandırılmaktadır. Aşağıda sırasıyla kameradan alınan görüntü, imge bölgesinde renklendirme ile elde edilen görüntü ve tespit edilen levha görüntüsü gösterilmektedir.



III. TRAFİK LEVHASI TANIMA

Bir sonraki aşama olan levha tanımlama işleminde imge bölgesinden kesilip yeniden boyutlandırılmış yani normalize edilmiş görüntü veri tabanındaki görüntülerle karşılaştırılarak doğru sonucu bulmak hedeflenmektedir. Karşılaştırma işleminde sisteme ikili kodlama ile tanımlanmış yeniden boyutlandırılmış görüntü ile veri tabanındaki aynı şekilde oluşturulmuş resimlerin matrissel farkları alınır ve veri tabanında en küçük farka sahip olan görüntü ulaşılmak istenen görüntüdür.

A. Görüntüyü Yeniden Boyutlandırma

İmge bölgesi renklendirme yöntemi ile levha bulunup çerçevlendikten sonra kesilerek elde edilen görüntü veri tabanı ile karşılaştırılmak için şablona ait boyuta getirilmesi gerekmektedir. Bunun için kesilen görüntü 100x100'lük boyutlara getirilerek veri tabanındaki görüntülerle karşılaştırma yapılabilecek hale getirilmektedir. Bu işlem yazılan 'cutimresize' isimli MATLAB kodu ile gerçekleştirilmektedir.

B. Karşılaştırma

Seçilen levhaların kameradan alınan görüntülerinin HSV eşiklenmesi, imge bölgesinde renklendirilmesi, bulunan levhanın kesilip yeniden boyutlandırılması ve siyah beyaza çevrilmesi sonucu oluşturulan veri tabanı ile karşılaştırma işlemi gerçekleştirilmektedir. Karşılaştırma işlemi sırasında test edilecek görüntünün yukarıdaki bölümlerde anlatılan işlemlerden geçirildikten sonra veri tabanında bulunan her bir görüntü ile fark alınarak ve fark matrisinin elemanlarının mutlak değerce toplanması ile yapılmaktadır. Veri tabanındaki görüntüler ile bir vektör oluşturulmuştur ve bu görüntülerin her biri bilgisayar ortamında aslında matristir. Fark alma ise bilgisayar ortamında matris farkı işlemine denk gelmektedir. Kameradan alınan görüntü ve veri tabanındaki her bir görüntü ile elde edilen fark matrislerinin elemanlarının mutlak değerce toplanması sonucu elde edilen değerler arasından en küçük değere sahip olan fark matrisini veren veri tabanı görüntüsü sonucu göstermektedir.

$$\text{VERİ TABANI} = \begin{bmatrix} 100 \times 100 & 100 \times 100 & \dots & 100 \times 100 \\ (\text{imge}) & (\text{imge}) & & (\text{imge}) \end{bmatrix}_{1 \times N}$$

$$\text{Gelen Resim} = \begin{bmatrix} \dots \\ \dots \\ \dots \end{bmatrix}_{100 \times 100}$$

$$\text{Fark} = \text{Veri Tabanı} - \text{Gelen Resim} = \begin{bmatrix} 100 \times 100 & 100 \times 100 & \dots & 100 \times 100 \\ (\text{imge}) & (\text{imge}) & & (\text{imge}) \end{bmatrix}_{1 \times N}$$

$$\text{Fark} = \left[\sum_{i,j=1}^{100} F_{ij}, \sum_{i,j=1}^{100} F_{ij}, \dots, \sum_{i,j=1}^{100} F_{ij} \right]_{1 \times N}$$

$$\text{Sonuç} = \min(\text{Fark})$$

Şekil 6: Karşılaştırma İşleminin Matematiksel İfadesi

IV. DENEY

Veri tabanında bulunan rastgele seçilen beş tane görüntüler kullanılarak aydınlık ve karanlık ortamlarda farklı koşullarda yapılan deneyin sonuçları aşağıda gösterilmektedir.

Yapılan Testler ve Başarı Oranı

Park Yasak	%90	Cepheden(Aydınlık)	%100
Hız Sınırı 50	%80	Sağa Açılı(A)	%80
Sağa Keskin Viraj	%70	Sola Açılı(A)	%60
Tek Yön Sağ(M)	%60	Kameranın Sağında(A)	%100
Yaya Geçidi	%50	Kameranın Solunda(A)	%80
		Sağa Dönmüş(A)	%40
		Sola Dönmüş(A)	%60
		Cepheden(Karanlık)	%80
		Sağa Açılı(K)	%60
		Sola Açılı(K)	%40

Yukarıda bahsedilen deneyden farklı olarak veri tabanında bulunan tüm levha çeşitleri için aydınlık ve cepheden olacak şekilde yapılan deneyden elde edilen başarı oranı %81.25 olarak bulunmuştur.

V. SONUÇ

Bu çalışmada doğru ve hızlı sonuç alabilmek adına soruna temel bir düzeyde yaklaşılmıştır. Bu amaçla HSV, imge bölgesinde renklendirme ve fark alma gibi temel yöntemler kullanılmıştır. Buna rağmen elde edilen başarı oranının yeterli bir seviyede olduğu görülmektedir. Ancak, levhaların bulunduğu farklı ortamlardan dolayı kaynaklanan hataları da iyileştirmek adına; aynı zamanda daha fazla çeşitte levha kullanmak için 'k nearest neighbour', 'Circular Hough Transform' gibi yöntemleri de kullanılabilir.

TEŞEKKÜR

Bu çalışmada yardımlarını esirgemeyen Araş.Gör.Bahri ABACI, Araş.Gör.Dr. Abdulkerim ÇAPAR hocalarımıza teşekkür ederiz.

KAYNAKÇA

- [1] U. S. Aydın. Traffic sign recognition. *Middle east technical university, Master thesis*, 2009.
- [2] A. De la Escalera, J. M. Armingol, and M. Mata. Traffic sign recognition and analysis for intelligent vehicles. *Image and vision computing*, 21(3):247–258, 2003.
- [3] A. De La Escalera, L. E. Moreno, M. A. Salichs, and J. M. Armingol. Road traffic sign detection and classification. *Industrial Electronics, IEEE Transactions on*, 44(6):848–859, 1997.
- [4] S. Maldonado-Bascón, S. Lafuente-Arroyo, P. Gil-Jimenez, H. Gómez-Moreno, and F. López-Ferreras. Road-sign detection and recognition based on support vector machines. *Intelligent Transportation Systems, IEEE Transactions on*, 8(2):264–278, 2007.