

Görüntü İşlemenin Temelleri Ödevi

Kodun aşamaları:

1. İlk olarak görüntüyü yüklüyoruz ve 'image' adında bir değişkene atıyoruz.
2. 'imshow' ve 'title' fonksiyonlarını kullanarak orijinal görüntüyü ekranda gösteriyoruz.
3. Görüntüyü RGB renk uzayına dönüştürmek için herhangi bir işlem yapmadan 'rgbImage' adında yeni bir değişkene atıyoruz.
4. Kırmızı renk için eşik değerlerini belirliyoruz. 'redThresholdLow' ve 'redThresholdHigh' değişkenleri, kırmızı renk için alt ve üst eşik değerlerini temsil eder.
5. Yeşil renk için eşik değerlerini belirliyoruz. 'greenThresholdLow' ve 'greenThresholdHigh' değişkenleri, yeşil renk için alt ve üst eşik değerlerini temsil eder.
6. Sarı renk için eşik değerlerini belirliyoruz. 'yellowThresholdLow' ve 'yellowThresholdHigh' değişkenleri, sarı renk için alt ve üst eşik değerlerini temsil eder.
7. Kırmızı renk için bir maske oluşturuyoruz. 'redMask' değişkeni, kırmızı renk bölgelerini belirlemek için RGB görüntü üzerinde belirli eşik değerlerine göre mantıksal bir maske oluşturur.
8. Yeşil renk için bir maske oluşturuyoruz. 'greenMask' değişkeni, yeşil renk bölgelerini belirlemek için RGB görüntü üzerinde belirli eşik değerlerine göre mantıksal bir maske oluşturur.
9. Sarı renk için bir maske oluşturuyoruz. 'yellowMask' değişkeni, sarı renk bölgelerini belirlemek için RGB görüntü üzerinde belirli eşik değerlerine göre mantıksal bir maske oluşturur.
10. Kırmızı renk bölgelerini bulmak ve çizmek için 'regionprops' ve 'rectangle' fonksiyonlarını kullanıyoruz. 'regionprops' fonksiyonu, kırmızı maske üzerindeki bağlantılı bileşenleri (bölgeleri) bulur. 'rectangle' fonksiyonu ise her bir bileşenin sınırlayıcı kutusunu (bounding box) çizer.
11. Yeşil renk bölgelerini bulmak ve çizmek için aynı adımları uyguluyoruz.
12. Sarı renk bölgelerini bulmak ve çizmek için yine aynı adımları uyguluyoruz.
13. Kırmızı renk bölgelerini göstermek için yeni bir şekil çizmek için bir 'figure' oluşturuyoruz.
14. Oluşturulan 'figure' üzerinde orijinal görüntüyü 'imshow' fonksiyonuyla gösteriyoruz.
15. Kırmızı renk bölgelerini temsil eden sınırlayıcı kutuları ('bounding box') çizmek için bir döngü kullanıyoruz. Her bir sınırlayıcı kutu, 'rectangle' fonksiyonuyla çizilir.
16. Aynı adımları takip ederek yeşil ve sarı renk bölgelerini göstermek için yeni bir 'figure' oluşturuyoruz ve renk bölgelerini çiziyoruz.

Ad: Emir Muhammet Aydemir
No: 171419008

Sonuçlar:



Şekil 1. Sarı renkli görüntünün tespiti



Şekil 2. Yeşil renkli görüntünün tespiti

Sarı Renk Bölgeleri



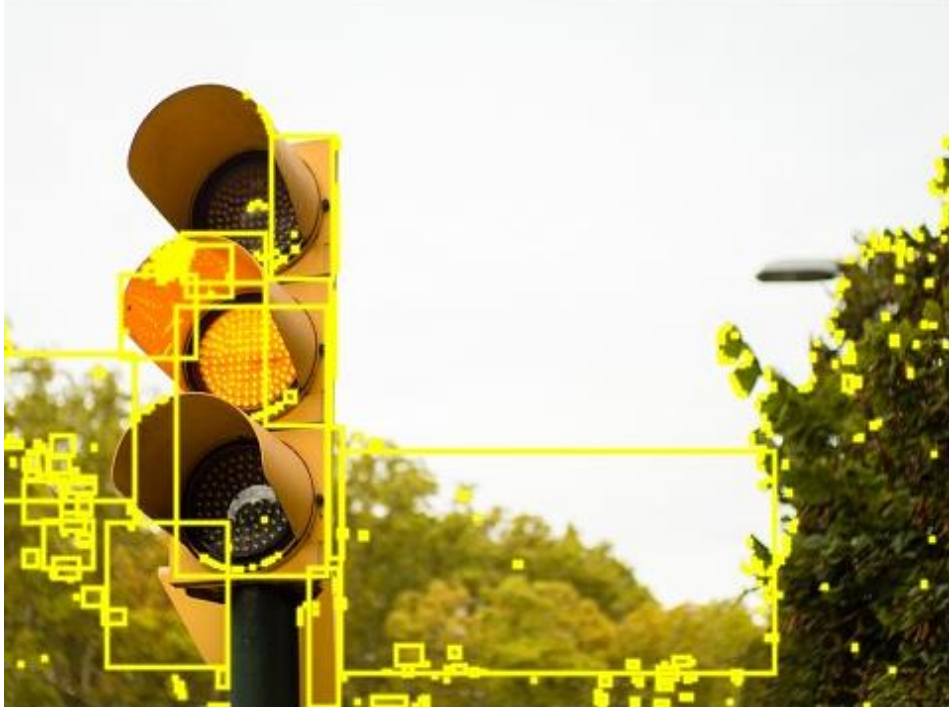
Şekil 3. Sarı renkli görüntünün tespiti

Yeşil Renk Bölgeleri



Şekil 4. Yeşil renkli görüntünün tespiti

Sarı Renk Bölgeleri



Şekil 5. Sarı renkli görüntünün tespiti

Yeşil Renk Bölgeleri



Şekil 6. Yeşil renkli görüntünün tespiti

Yeşil Renk Bölgeleri



Şekil 7. Yeşil renkli görüntünün tespiti

Kırmızı Renk Bölgeleri



Şekil 8. Kırmızı renkli görüntünün tespiti

Kırmızı Renk Bölgeleri



Şekil 9. Kırmızı renkli görüntünün tespiti

Kırmızı Renk Bölgeleri



Şekil 10. Kırmızı renkli görüntünün tespiti

Ad: Emir Muhammet Aydemir
No: 171419008

Sonuç olarak yazdığım kod içerisinde verilen eşik değerleri sebebiyle şekil 5 ve şekil 9 çizmemesi gereken yerleri de çizmiştir. Görüntü içerisinde trafik ışığı ile aynı renkte cisim var ise o da kırmızı dikdörtgen içerisine alındığı için şekil 5 ve 9'da sonuç hatalıymış gibi gözükmektedir. Renkleri algılama kısmında kusur yoktur. Tüm görsellerde renkleri doğru bir şekilde ayrıştırıp ilgili ışığı dikdörtgen içine alma işlemi başarılı bir şekilde gerçekleştirilmiştir.