

## İÇİNDEKİLER

|   |       |
|---|-------|
| İÇİNDEKİLER .....   | i     |
| 1. GİRİŞ .....  | 3     |
| 2. PROJEDE YAPILAN ÇALIŞMALAR .....                         | 4     |
| 2.1 Kavramsal Tasarım .....                                 | 4     |
| 2.2 İş-Zaman Çizelgesi .....                                | 5     |
| 2.3 Projede Gerçekleşen İş Paketleri ve Ara Çıktıları ..... | 6,7,8 |
| 3. KAYNAKLAR .....  | 9     |

## 1. GİRİŞ

Sürücü yorgunluk tespit sistemi'nin gerçek zamanlı ve güvenilir olması trafik güvenliği açısından son derece önemlidir. Trafik; yol, araç ve insan üçlüsünden oluşmaktadır. Bunlardan herhangi birinde bozukluk olması trafik kazalarına yol açmaktadır. Bu kazaların başlıca nedenlerinden biri de uykusuzluktur. Uyku ile ilişkili bu problemin önüne geçmek için sürücü ile herhangi bir teması olmayan gömülü bir sistem tasarlanmalıdır. Günümüzde bu tür uygulamalar tespit yöntemlerine göre üç farklı sınıfa ayrılıyor. Bunlar;

- Fizyolojik yöntemler; EKG gibi cihazlar ile sürücünün yorgunluk tespiti yapılabiliyor. Başarılı bir yöntem olmasına karşın ek cihazlara ihtiyaç duyulduğu için pek kabul edilebilir bir yöntem değildir.
- Sürüş şekli veya araba hareketlerine göre tespit yapılan yöntemler. Bu yöntemde sürücünün aracı sürüş şekline göre bir anormallik olup olmadığının tespiti yapılıyor. Burada da yolun bozuk olmasından dolayı sürücü farklı sürüş davranışları sergileyeceğinden dolayı bu yöntem de pek tercih edilmiyor.
- Bilgisayarla görü ile yapılan tespit yöntemleri: Sürücü yorulduğunda yüzü'nde bir takım fiziksel değişiklikler olacaktır. Bunlar; sıklıkla esnemek, başın düşmesi(kontrol edememesi), gözün kapanmaya başlaması, gözkapağı'nın irisi kapatmaya başlaması gibi fiziksel değişiklikler bilgisayarla görü yöntemleri kullanılarak saptanabilir.

Proje kapsamı'nda gerçek hayatta da sıklıkla tercih edilen bilgisayarla görü yöntemleri kullanılmıştır. Burada tespit yapılırken önemli olan gözün açık veya kapalı olduğuna karar verilmesidir.

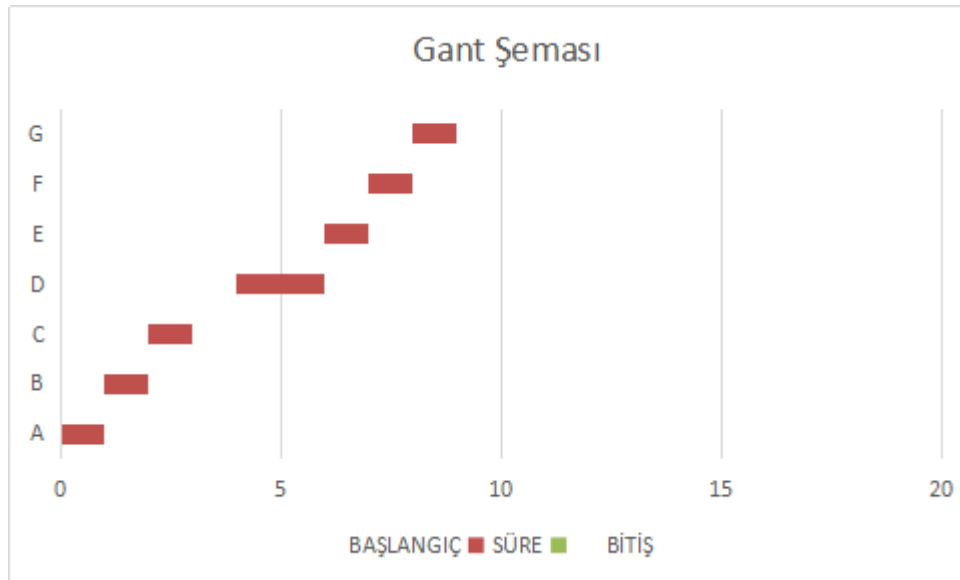
Bu projede viola-jones algoritması kullanılarak yüz ve göz tespiti yapılmıştır. Gözün açıklık durumuna göre iris tespit edilmiştir. Proje'nin ilerleyen aşamaları'nda PERCLOS algoritması kullanılarak gözün ne kadar açık veya kapalı olduğu yüzde olarak bulunabilir. Mesela yüzde yirmi'den az bir açıklık varsa gözün kapalı olduğuna karar verilip buna göre arka planda farklı sistemler eklenebilir.

## 2. PROJEDE YAPILAN ÇALIŞMALAR

### 2.1 Kavramsal Tasarım

İrisin yerini belirlemek için, göz resmi bir takım işlemlerden geçirilecektir. Bunlar; Histogram dengeleme, orijinal resmi binary dönüştürme, aşındırma ve genişletme işlemleridir. Histogram dengeleme resmin parlaklığını artıracak bu sayede eşik belirlemek daha kolay olacak ve ışık yoğunluğu değişiminde de verimli bir şekilde resim üzerinde işlem yapılabilir. Binary dönüştürme, bir eşikleme işlemidir. Gri tonlamalı resimde göz bebeği karanlık olacağından eşikleme yöntemi ile iris tespit edilebilecektir. Aşındırma ve genişletme işlemleri arka arkaya uygulandığında açma işlemi yapmış olacaktır. Bu işlem ile göz kapağı ve kirpikler binary resimden kaldırılmış olacaktır ve bu sayede iris ile karıştırılması engellenmiş olacaktır. Böylece iris daha açık bir şekilde belirlenmiş olacaktır. İrisin açıklık seviyesi %20'nin altında olduğunda gözün kapalı olduğuna karar verilecektir. Bunun dışındaki açıklık durumunda gözün ne kadar açık olduğu yüzde olarak verilecektir.

### 2.2 İş-Zaman Çizelgesi



A:RGB resim üzerinde viola jones algoritması ile gözün yeri tespit edilmiştir.Kesme işlemi ile göz imgesi elde edilmiştir.

**B:** Elde edilen resim üzerinde gri skala dönüşümü, histogram dengeleme ve ikilik (binary)'ye çevirme ve morfolojik işlemler yapılmıştır.

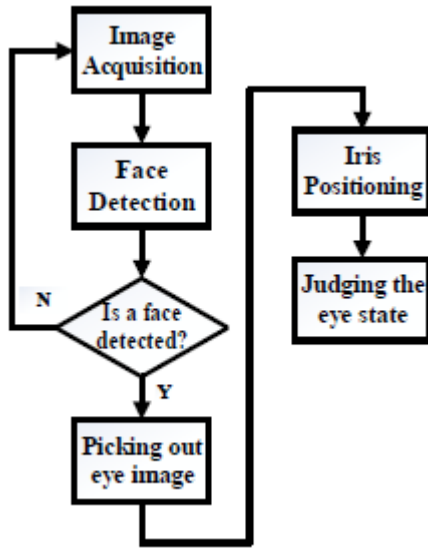
**C:** Hatalar ve eksikler giderildikten sonra resim üzerinde iris belirlenmiştir.

**D:** Aynı işlemleri uygulamak için bu kez video çekilip kare kare ayrılarak iris açıklığının farklı seviyelerinde resimler elde edilmiştir.

**E:** Elde edilen resimler üzerinde B adımıdaki işlemler yeniden gerçekleştirilmiştir.

**F:** İrisin pozisyonu belirlenmiş ve resim üzerinde gösterilmiştir.

**G:** Farklı resimlerdeki irisin açıklık kapalılık durumuna göre algılanan kısmın da küçüldüğü gözlenmiştir.



### 2.3 Projede Gerçekleşen İş Paketleri ve Ara Çıktıları

İlk olarak orijinal resim'den göz kısmını ayırabilmek için viola jones algoritması kullanılarak gözün yeri tespit edilmiştir. Tespit ettiğimiz göz imgesi'nde irisin gürbüz tespitinin yapılabilmesi için kaşın bulunduğu kısım resimden imcrop komutu ile kesilmiştir.

```
EyeDetect = vision.CascadeObjectDetector('RightEyeCART');  
Image = imread('1.bmp');  
bb_Eye = step(EyeDetect, Image);  
eye_crop = imcrop(Image, bb_Eye(1, :));
```

Viola-jones algoritması ile resim'den göz kısmı bulunmuştur. imcrop ile bulunan resimden göz imgesi bir değişkene atılmıştır. Ekran çıktısı aşağıdaki gibidir.



Bu resimden kaş kısmı çıkarılarak irisin tespiti kolaylaşacaktır. Aynı zamanda bu işlemlerin farklı resimler üzerinde de düzgün çalışabilmesi için resmin boyutu imresize komutu ile sabitlenmiştir. Bu sayede kaş kısmını resimden çıkarmak daha kolay olmuştur.



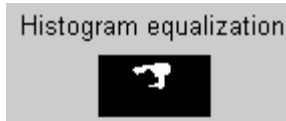
```
eye_crop = imresize(eye_crop, [30 30]);  
h = bb_Eye(1,3);  
w = bb_Eye(1,4);  
eye = imcrop(eye_crop, [1,10,h,w]);
```

Burada yapılan işlemler ikilik resim üzeri'nde yapıldığı için tespit edilmek üzere rgb resim girildiği'nde bunun tespit edilip ikilik resme çevrilmesi için aşağıdaki işlemler yapılmıştır.

```
test_size = 3;  
im_size = length(size(Image));  
if test_size == im_size  
    eye_crop = rgb2gray(eye_crop);  
end
```

Bu adımdan sonra resmimiz üzerinde histogram dengeleme, ikilik resimde eşik belirleme işlemleri gerçekleştirilmiştir. Sonuçta daha iyi kontrasta sahip bir resim elde edilmiştir ve bu resim üzerinde morfolojik işlemler uygulanması daha gürbüz bir algoritma olmasını sağlayacaktır.

```
%Histogram Equalization  
HistEq = histeq(Eye);  
%Binarization (threshold)  
t = 16;  
% find values below  
ind_below = (HistEq < t);  
% find values above  
ind_above = (HistEq >= t);  
% set values below to black  
HistEq(ind_below) = 255;  
% set values above to white  
HistEq(ind_above) = 0;
```



Bu adımda, aşındırma ve genişletme işlemleri arka arkaya uygulanmıştır. Burada amaç;görüntüde işimize yaramayacak öz niteliklerin ayrıştırılması ve işimize yarayacak öz niteliklerin (iris gibi) korunmasıdır.

```
%Morphologic Operations (Erosion & Dilation)
```

```
filledHistEq = imfill(HistEq, 'holes');
se = strel('ball',1,1);
eroded = imdilate(filledHistEq, se);
se2 = strel('ball',1,1);
dilate=imdilate(eroded, se2);
figure, imshow(eroded);
title('Erosion');
figure, imshow(dilate);
title('Dilation');
```



Son olarak elde etmiş olduğumuz öz nitelikleri resim üzerinde göstereceğiz.bwlabel komutu ile resim üzerinde etiketleme işlemi yapılmıştır.8 rakamı;etiketleme esnasında morfolojik işlemler sonucu elde edilen imge'nin her pikselde ve onun 8 komşuluğunda görüntünün eşleştirilmesi işlemidir.Default olarak bu rakam 2'dir fakat bu şekilde algoritma daha hızlı sonuç vermektedir.

```
[L,num] = bwlabel(dilate,8);
RP = regionprops (L, 'Area');
max=0;
for i=1:num
    if (RP(i).Area > max)
        max = RP(i).Area;
    end
end
for j=1:num
    if (max == RP(j).Area)
        maxx_ind = j;
    end
end
```

Bounding box yöntemi ile irisi dikdörtgen içerisinde göstereceğiz.

```
%Bounding Box
```

```
RP1 = regionprops (L, 'BoundingBox');
```

```

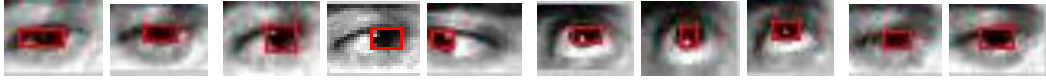
figure
imshow(Eye);
title('Iris Positioning');
hold on;
rectangle('Position',[RP1(maxx_ind).BoundingBox],'EdgeColor','r');
clear all;

```

Farklı açılarda ve açıklıktaki göz imgeleri için program çıktıları:



Aşağıdaki resimler karanlık olduğu için imadjust komutu ile kontrast artırılarak görüntü alınıp buraya eklenmiştir. Resimlerdeki gürültüler bu nedenden dolayı daha net görülmüştür.



Farklı bakış açıları'nda farklı açıklık seviyeleri'nde iris tespiti yapılmıştır.İrisin küçülmesi ile bulunan kısım da küçülmektedir.Bir sonraki aşamada PERCLOS yöntemi ile irisin görünürlük seviyesine göre gözün açık veya kapalı olduğunun kararı verilebilir.İris tespiti aynı zamanda hough dönüşümü yöntemi ile de yapılabilir. Buna göre farklı avantaj ve dezavantajlar olacaktır. Bu projede hough dönüşümü kullanılmamıştır.

### **3. KAYNAKLAR**

- [1] Aldrich MS, Automobile accidents in patients with sleep disorders. Sleep Dec;12(6):487-94,1989.
- [2] Ali Acioğlu, Ergun Erçelebi “Sürücü Uyku Durum Tespiti için Gerçek Zamanlı Göz Tespit Algoritması (Real Time Eye Detection Algorithm for PERCLOS Calculation 978-1-5090-1679-2/16/ 2016 IEEE.
- [3] A Real-time Fatigue Driving Detection System Design and Implementation Zhao-Bin Ma\*, Yang Yang\*, Fengyu Zhou\*, Xu Jian-Hua\* July 1-3-2015 ICACT2015