



## İLERİ GÖRÜNTÜ İŞLEME YARDIMCI NOTLARI -2024

Matlab, video dosyalarını işlemek ve canlı kamera görüntülerini analiz etmek için güçlü araçlar sunar. Bu ders notlarında video dosyalarının nasıl açılacağını, kare kare işleneceğini ve sonuçların kaydedileceğini öğreneceğiz. Ayrıca, canlı kamera görüntülerinin nasıl yakalanacağını ve işleneceğini de inceleyeceğiz. Bu beceriler, özellikle görüntü işleme, nesne takibi ve hareket analizi gibi konularda kullanılabilir.

Hedefler:

- Matlab kullanarak video dosyası okuma ve işleme.
- Kare kare video işleme yöntemleri.
- Canlı kamera görüntülerinin yakalanması ve işlenmesi.
- Örnek uygulamalar ile becerilerin pekiştirilmesi.

### 1. Video Okuma

Video dosyalarını Matlab'da işlemek için VideoReader fonksiyonu kullanılır. Bu fonksiyon ile video dosyası açılarak kare kare işlenebilir.

#### Örnek: Bir video dosyasını açma ve kare sayısını bulma

```
videoFile = 'video.mp4'; % Video dosyasının yolu
vidObj = VideoReader(videoFile); % Video dosyasını aç

% Video bilgileri
disp(['Kare Sayısı: ', num2str(vidObj.NumFrames)]);
disp(['Kare Hızı (fps): ', num2str(vidObj.FrameRate)]);
```

### 2. Kare Kare Video İşleme

Bir videoyu kare kare işleyerek her kare üzerinde işlemler yapabiliriz. Örneğin, her kareyi gri tonlamaya çevirmek ve işlenmiş kareleri yeni bir video dosyası olarak kaydetmek mümkündür.

#### Örnek: Her kareyi gri tonlamaya çevirme ve yeni bir video oluşturma

```
outputVideo = VideoWriter('output.avi'); % Yeni video dosyası oluştur
open(outputVideo);

while hasFrame(vidObj)
    frame = readFrame(vidObj); % Her kareyi oku
    grayFrame = rgb2gray(frame); % Gri tonlamaya çevir
    writeVideo(outputVideo, grayFrame); % Gri kareyi yeni videoya yaz
end

close(outputVideo);
disp('İşleme tamamlandı ve yeni video oluşturuldu.');
```



### 3. Canlı Kamera Görüntülerinin İşlenmesi

Matlab, bilgisayarınıza bağlı kameraların görüntülerini işlemek için webcam fonksiyonunu sağlar. Öncelikle, Bilgisayara bağlı bir kamerayı kullanmak için önce webcam nesnesini oluşturmanız gerekir.

#### Örnek: Kamera bağlantısını kurma ve görüntü yakalama

```
cam = webcam; % Kameraya bağlan  
frame = snapshot(cam); % Bir kare yakala  
imshow(frame); % Görüntüyü göster
```

### 4. Canlı Görüntü Akışını İşleme

Canlı görüntü akışını işlemek, video dosyasına benzer şekilde kare kare görüntü yakalayarak mümkündür. Aşağıdaki örnek, canlı kameradan gelen görüntüyü gri tonlamaya çevirip ekranda göstermektedir.

#### Örnek 4: Canlı kamera görüntüsünü işleme

```
cam = webcam; % Kameraya bağlan  
  
figure;  
while true  
    frame = snapshot(cam); % Kameradan bir kare yakala  
    grayFrame = rgb2gray(frame); % Gri tonlamaya çevir  
    imshow(grayFrame); % Gri görüntüyü göster  
    drawnow; % Ekranı güncelle  
end
```

### 5. Nesne Tespiti

Bir videoda veya canlı görüntüde belirli bir nesnenin takibi yapılabilir. Örneğin, belirli bir renk aralığındaki nesnelere bulmak için renk tabanlı filtreleme kullanılabilir.

#### Örnek: Kırmızı renkteki nesnelere tespit etme

```
cam = webcam; % Kameraya bağlan  
  
while true  
    frame = snapshot(cam); % Kameradan bir kare yakala  
    redChannel = frame(:,:,1); % Kırmızı renk kanalını al  
    binaryMask = redChannel > 150; % Kırmızı renk eşikini uygula  
    imshow(binaryMask); % Sonucu göster  
    drawnow; % Ekranı güncelle  
end
```



## 6. Hareket Algılama

Bir görüntüdeki hareketi tespit etmek için ardışık kareler arasındaki farklar kullanılabilir. Hareket algılama, güvenlik sistemleri ve otomatik video izleme sistemlerinde yaygın olarak kullanılır.

### Örnek: Hareket algılama

```
cam = webcam; % Kameraya bağlan
prevFrame = snapshot(cam); % İlk kareyi al ve sakla

while true
    currentFrame = snapshot(cam); % Şu anki kareyi al
    diffFrame = imabsdiff(rgb2gray(currentFrame), rgb2gray(prevFrame)); % İki
    kare arasındaki farkı al
    imshow(diffFrame > 30); % Fark büyükse hareket var
    prevFrame = currentFrame; % Şu anki kareyi güncelle
    drawnow; % Ekranı güncelle
end
```

## 6. Canlı Görüntü Veya Videoların Diske Kaydedilmesi

Matlab, videoların farklı formatlarda kaydedilmesine de olanak tanımaktadır. Örneğin, sıkıştırılmamış AVI formatında veya MPEG-4 gibi formatlarda video oluşturulabilir.

### Örnek: Karelerin gri tonlamaya dönüştürülmesi ve yeni bir video oluşturulması

```
outputVideo = VideoWriter('output.avi'); % Yeni video dosyası oluştur
open(outputVideo);

while hasFrame(vidObj)
    frame = readFrame(vidObj); % Her kare okunur
    grayFrame = rgb2gray(frame); % Gri tonlamaya çevrilir
    writeVideo(outputVideo, grayFrame); % Yeni videoya yazılır
end

close(outputVideo);
disp('İşlem tamamlanmış ve yeni video oluşturulmuştur.');
```



## Video Dosyalarının okunması ve kare kare (çerçeve çerçeve) analiz edilmesi:

```
% Bu program dosyaya kayıtlı bir video dosyasından çerçeveler halinde imge  
% okur, bir önceki çerçeve ile farkını alır, ikilik görüntü yapar ve son  
% halini bir figure penceresi üzerinden video olarak gösterir.  
% Bu işlemleri yaparken okuduğu çerçeveleri *.png dosyaları olarak  
% bilgisayara kaydetmez. Bu nedenle hızlı çalışır.
```

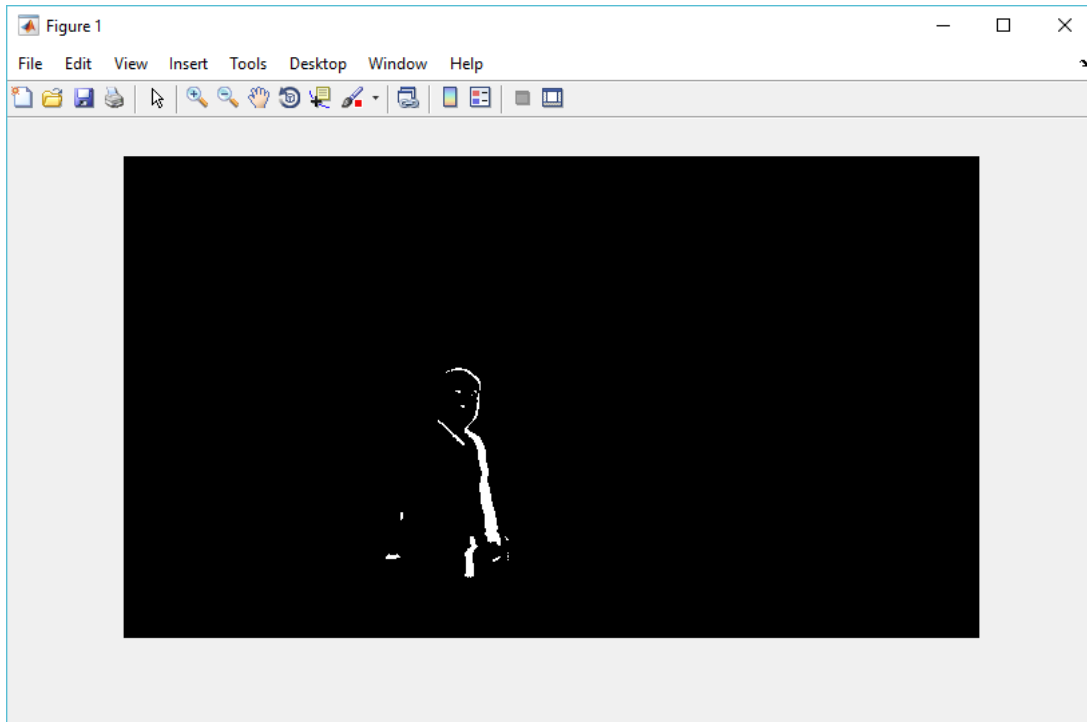
```
clc;clear;close all;  
% filename = 'video1.mp4';  
filename = 'video2.mp4';  
mov = VideoReader(filename);  
artis=3;
```

Kaç çerçevede bir okuyacağımızı belirledik.

```
%Bir önceki çerçeve başlangıç için boş olarak açılıyor.  
rEski=zeros(mov.Height,mov.Width);  
rEski=uint8(rEski);
```

Çerçeve çıkartma işleminde şu anki çerçeveden bir önceki çerçeve çıkartılır. Ancak ilk işlemde bir önceki çerçeve olmadığı için boş bir çerçeve hazırlanıyor.

```
%okuma işlemi for döngüsünde başlıyor.  
for k=1:artis:mov.NumberOfFrames;  
r=read(mov,k);  
r=rgb2gray(r);  
fark=imsubtract(rEski,r);  
fark=fark>30;  
%aşağıdaki kod ile her fark çerçevedeki piksellerin toplamı alınıyor.  
%En yüksek fark iki çerçeve arasındaki en yoğun değişimi gösterir.  
fprintf('Frame No=%d Fark Pikseller=%d\n',k,sum(fark(:)));  
fark=imresize(fark,0.5);  
figure(1);  
imshow(fark);  
rEski=r;  
end
```





**AKÜ TEKNOLOJİ FAKÜLTESİ**  
**MEKATRONİK MÜHENDİSLİĞİ ANA BİLİMDALI**



## **VİDEO OKUMA ve ALARM OLUŞTURMA**

Aşağıdaki program ile video çerçeveleri arasında yoğun bir fark oluştuğunda Arduino kullanılarak bir led yakma işlemi gerçekleştirilmiştir. Led arduino'nun D2 pinine montajlanmıştır. Led yerine buzzer da kullanılabilir. Genel olarak video alarm sistemlerinde de piksel hareketliliği kullanılmaktadır.

```
clc;clear;close all;
a=arduino('COM3','Uno');
% filename = 'video1.mp4';
filename = 'video2.mp4';
mov = VideoReader(filename);
artis=3;

%Bir önceki çerçeve başlangıç için boş olarak açılıyor.
rEski=zeros(mov.Height,mov.Width);
rEski=uint8(rEski);

%okuma işlemi for döngüsünde başlıyor.
for k=1:artis:mov.NumberOfFrames;
r=read(mov,k);
r=rgb2gray(r);
fark=imsubtract(rEski,r);
fark=fark>30;
%aşağıdaki kod ile her fark çerçevedeki fark piksellerin toplamı alınıyor.
%En yüksek fark iki çerçeve arasındaki en yoğun değişimi gösterir.
fprintf('Frame No=%d Fark Pikseller=%d\n',k,sum(fark(:)));
fark=imresize(fark,0.5);
figure(1);
imshow(fark);
%Eğer fark piksellerin toplamı 1000'den büyük ise led yanacaktır.
if sum(fark(:))>1000
    writeDigitalPin(a,'D2',1)
else
    writeDigitalPin(a,'D2',0)
end
rEski=r;
end
```



## VİDEO OKUMA ve YENİDEN KAYDETME

Yukarıdaki programlar aracılığıyla video kayıtlarını program yardımıyla çerçeve - çerçeve okuyabildiğimizi gördük. Okuduğumuz çerçeveler üzerinde bir takım işlemler de yaptık ve sonucunda yeni çerçeveler elde ettik. Örneğin ardışık çerçeveleri birbirinden çıkartarak siyah beyaz çerçeveler elde ettik. Bu yeni görüntüleri farklı bir dosya adı ile kaydetmek istediğimizde aşağıdaki programı kullanabiliriz. Program genel itibarıyla yukarıdaki programlar ile aynıdır. Sadece kayıt için birkaç fonksiyon eklenmiştir.

```
clc;clear;close all;
filename='video1.mp4';

mov=VideoReader(filename);
artis=5;

rEski=zeros(mov.Height,mov.Width);
rEski=uint8(rEski);

writerObj=VideoWriter('Ornek.avi');
writerObj.FrameRate=3;
open(writerObj);

for k=80:artis:mov.NumberOfFrames
    r2=read(mov,k);
    r=r2;
    r=rgb2gray(r)+30;
    fark=imsubtract(rEski,r);
    fark=fark>30;
    fark=im2uint8(fark);
    fprintf('Frame No:%d Fark Değeri:%d\n',k,sum(fark(:)));
    fark=imresize(fark,0.5);
    rEski=r;
    writeVideo(writerObj,r2);
end
close(writerObj);
```

Yeni video dosyası  
oluşturmak için eklenen  
fonksiyon ve kodlar.



AKÜ TEKNOLOJİ FAKÜLTESİ  
MEKATRONİK MÜHENDİSLİĞİ ANA BİLİMDALI



## CANLI VIDEO İLE İŞLEMLER: -2

```
%% Setup
clear;clc;close all;
% 2. webcam'a bağlanılıyor.
cam = webcam(2);

%%tek görüntü alma
% kameradan sadece bir görüntü alıyoruz.
img = snapshot(cam);
imshow(img)

%% Video
set(gcf,'Position',[200 -10 1000 1000]);%figür penceresinin yeri ayarlandı.
vidWriter = VideoWriter('videoDosyasil.avi');
open(vidWriter);
% 500 çerçeve alalım
for frames = 1:500
% tek bir çerçeve alıyoruz
a = snapshot(cam);
a=imresize(a,[350 350]);
r=a(:,:,3);%kırmızı bant ayrıldı
gray=rgb2gray(a);%resim gri seviye yapıldı
tespitR=imsubtract(r,gray); %kırmızı banttan gri seviye çıkarılarak sadece kırmızı olanlar bulunuyor.
tespitR=tespitR>30;%kırmızılıarı tespit edilen resim siyah beyaz yapılıyor.
figure(1);
subplot(1,2,1); imshow(a);
subplot(1,2,2); imshow(tespitR);
writeVideo(vidWriter,a);
end
close(vidWriter);
clear cam
```

Webcam(1) ise dizüstü bilgisayarın dahili kamerası, webcam(2) ise usb web cam tanıtılır.



## VİDEO OKUMA, DİSKE KAYDETME ve VİDEO DOSYASI OLUŞTURMA

Aşağıdaki program ile bir video dosyası çerçeve çerçeve okunmaktadır. Okunan çerçeveler önce gri seviyeye dönüştürülmekte, sonra ise belirli bir eşik değeri kullanılarak ikilik görüntü haline getirilmektedir. Bu ikilik görüntü çerçevelerinin ardışık olarak farkları alınmakta ve böylece ekranda bir hareketlilik olduğunda tespit edilmeye çalışılmaktadır.

Okunan her çerçeve gri seviye olarak ve fark görüntü olarak hard diskte belirlenen bir dizine belirli bir dosya adı formatında kaydedilmektedir. Gri seviye çerçevelerin dosya adı formatı: 'frGI\_%04d.png' dir. Fark görüntülerin dosya adı formatı ise: 'frGIBW\_%04d.png' dir.

Ayrıca bu programın sonunda bulunan kodlar ile fark görüntülerin videosu oluşturulmuştur. Bu şekilde bir yanda videonun orijinal görüntüsü akarken, bir yanda da ardışık çerçeve farkları ayrı bir video dosyası olarak izlenebilmektedir.

```
clc;clear;close all;
filename = 'C:\Users\EliteBook\Pictures\Camera
Roll\HareketTespitVideo1.mp4';
mov = VideoReader(filename);
artis=5; % kaç çerçevede bir okunacağını belirliyoruz.

%Bir önceki çerçeve başlangıç için boş olarak açılıyor.
rEski=zeros(mov.Height,mov.Width);
rEski=uint8(rEski);

%Varsa önceki dosyalar siliniyor.
dosyaAdi=sprintf('fr*.png');
dizin1=fullfile('C:\Users\EliteBook\Documents\MATLAB\videolar','GI-
2018',dosyaAdi);
delete(dizin1);

%okuma işlemi for döngüsünde başlıyor.
for k=1:artis:mov.NumberOfFrames;
dosyaAdi1=sprintf('frGI_%04d.png',k);
dizin1=fullfile('C:\Users\EliteBook\Documents\MATLAB\videolar','GI-
2018',dosyaAdi1);
dosyaAdi2=sprintf('frGIBW_%04d.png',k);
dizin2=fullfile('C:\Users\EliteBook\Documents\MATLAB\videolar','GI-
2018',dosyaAdi2);
r=read(mov,k); %k. çerçeveyi oku
r=rgb2gray(r);
imwrite(r,dizin1);%okuduğun çerçeveyi dizin1 de tarif edilen yere ve
belirlenen dosya adı formatında yaz.
fark=imsubtract(rEski,r);%bir önceki çerçeve ile farkını al.
fark=fark>30;%ikilik görüntü haline getir.
%şağıdaki kod ile her fark çerçevedeki fark piksellerin toplamı alınıyor.
%En yüksek fark iki çerçeve arasındaki en yoğun değişimi gösterir.
fprintf('Frame No=%d Fark Pikseller=%d\n',k,sum(fark(:)));
fark=double(fark);
imwrite(fark,dizin2);
rEski=r;
end
```

Burada verilen dosya adresleri kendi bilgisayarınıza uygun olmalıdır. Aksi takdirde program hata verecektir.





```
%video oluşturulacak.  
%Video yazma nesnesi oluşturuldu  
writerObj = VideoWriter('VideoYazmaOrnegil.avi');  
%Çerçeve oranını belirleme  
writerObj.FrameRate = 3;  
%Nesneyi aç ve ardışık olarak frameleri yaz  
open(writerObj)  
  
for i=1:artis:mov.NumberOfFrames;  
    % Frameleri okumaya başla  
    dosyaAdi2=sprintf('frGIBW_%04d.png',i);  
    dizin2=fullfile('C:\Users\EliteBook\Documents\MATLAB\videolar','GI-  
2018',dosyaAdi2); %görüntülerin yazdırılacağı dizin ve dosya adı formatı  
    input = imread(dizin2);  
    %Frame'i yaz.  
    writeVideo(writerObj, input);  
end  
% video yazma nesnesini kapat  
close(writerObj);
```

## VİDEO OKUMA ve VİDEO DOSYASI OLUŞTURMA (diske kaydetmeden)

Aşağıdaki programda video1.mp4 video dosyasından okunan çerçevelerin kendinden bir önceki çerçeveden çıkartılmak suretiyle farkı alınmakta ve belirli bir eşikdeğer ile siyah beyaz formata dönüştürülmektedir. Sonrasında doğrudan videoWriter nesnesi kullanılarak \*.avi uzantılı video dosyası oluşturulmaktadır.

```
clc;clear;close all;  
filename='video1.mp4';  
mov=VideoReader(filename);  
artis=4;  
rEski=zeros(mov.Height,mov.Width);  
rEski=uint8(rEski);  
  
writerObj=VideoWriter('Ornek.avi');  
writerObj.FrameRate=3;  
open(writerObj);  
  
for k=1:artis:mov.NumberOfFrames  
    r=read(mov,k);  
    r=rgb2gray(r);  
    fark=imsubtract(rEski,r);  
    fark=fark>30;  
    fark=im2uint8(fark);  
    fprintf('Frame No:%d Fark Değeri:%d\n',k,sum(fark(:)));  
    fark=imresize(fark,0.5);  
    rEski=r;  
    writeVideo(writerObj,fark);  
end  
close(writerObj);
```

writeVideo()  
fonksiyonu uint8  
sınıfından veri istediği  
için logical'a dönüşen  
fark matrisi uint8'e  
dönüştürüldü.