

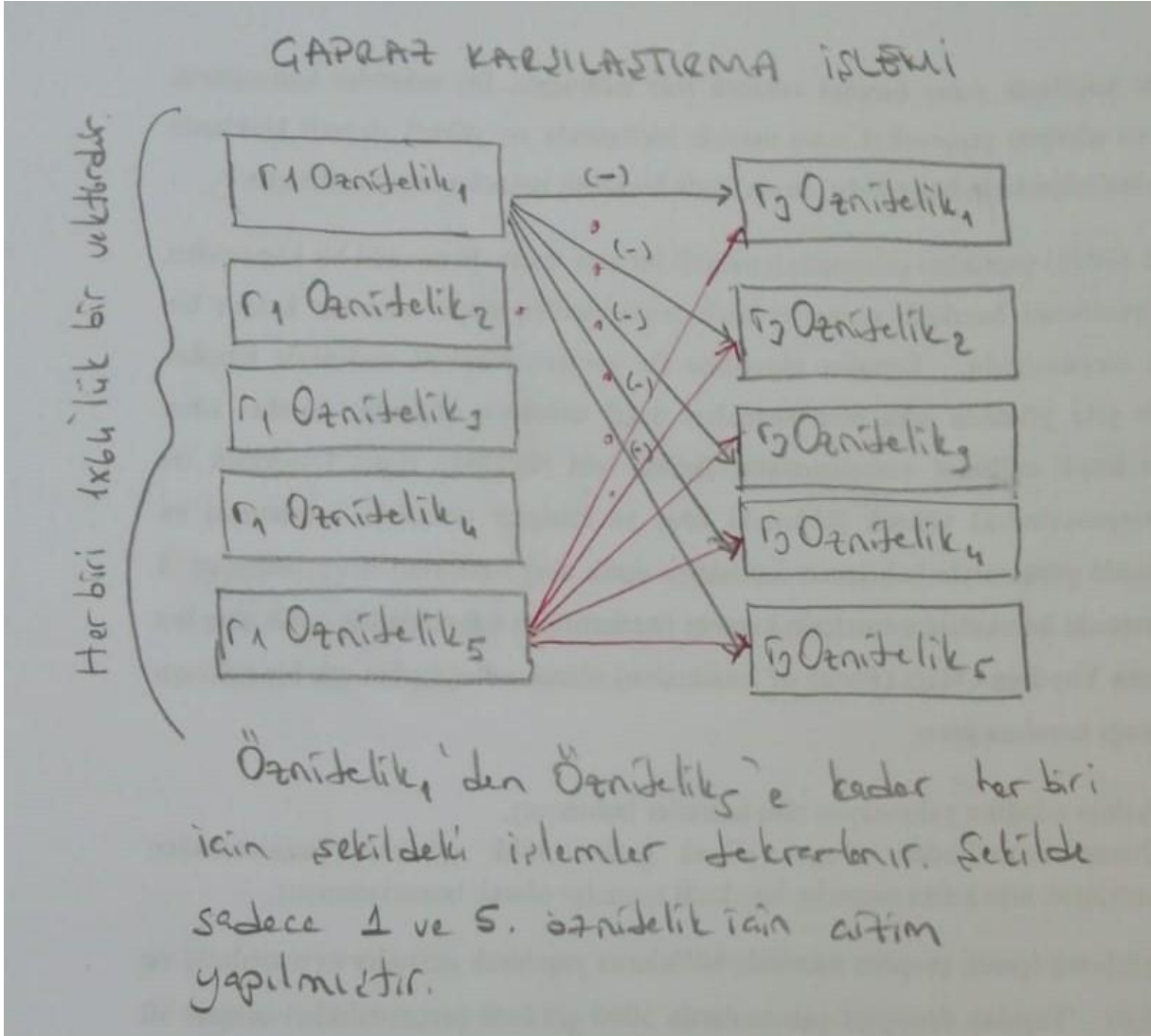


TEKNOLOJİ FAKÜLTESİ MEKATRONİK MÜHENDİSLİĞİ
GÖRÜNTÜ İŞLEME DERSİ VİZE YANITLARI (30.05.2018)

1.) 'headquarters-2.jpg' ve 'headquarters.jpg' isimli iki resim dosyası medyan filtresinden geçirildikten sonra SURF yöntemi kullanılarak en güçlü beşer öznetelikleri çıkartılacaktır. Buna göre:

- Elde edilen öznetelik vektörleri arasında çapraz karşılaştırma yaparak birbirine en benzer öznetelik çiftini bulunuz.
- Sonrasında en benzer öznetelik çiftinin resim üzerindeki lokasyonları (x,y koordinatları) arasındaki vektörel uzaklığı hesaplayınız. (40p)

$$\text{Vektörel Uzaklık} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$





AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ



Yanıt:

```
% 1. resmin öznitelikleri çıkartılıyor.
clear;clc;close all;
sayi=5; %en güçlü öznitelik sayısı
r1=imread('headquarters.jpg');
r1=rgb2gray(r1);
r1=medfilt2(r1);
r1points=detectSURFFeatures(r1);
r1Fet=extractFeatures(r1,r1points.selectStrongest(sayi));
figure;imshow(r1);
hold on;
plot(r1points.selectStrongest(sayi));
%%
% 2. resmin öznitelikleri çıkartılıyor.
r3=imread('headquarters-2.jpg');
r3=rgb2gray(r3);
r3=medfilt2(r3);
r3points=detectSURFFeatures(r3);
r3Fet=extractFeatures(r3,r3points.selectStrongest(sayi));
figure;imshow(r3);
hold on;
plot(r3points.selectStrongest(sayi));
%% Birbirine en yakın öznitelikleri bulmaya çalışıyoruz.
%Çapraz karşılaştırma yapılacak, yani bir resimdeki her öznitelik diğer
% resimdeki özniteliklerin tamamıyla karşılaştırılacak.
k=1;
for i=1:sayi
    for j=1:sayi
        %Önce farkOznitelikler tablosu oluşturuldu. Burada her bir satırda
        öznitelik
        %vektörleri arasındaki fark vektörü bulunur.
        farkOznitelikler(k,:)=abs(r1Fet(i,:)-r3Fet(j,:));

        %Burada her bir fark vektörü skaler bir değere dönüştürülür. Bu
        %işlemi farkTablosu(k,1)=sum(abs(r1Fet(i,:)-r3Fet(j,:))); şeklinde
        %tek satırda da yapabiliydik.
        farkTablosu(k,1)=sum(farkOznitelikler(k,:));

        %Burada i ve j alınıyor çünkü minimum farkı bulduğumuzda o farkın
        %oluşturduğu özniteliğin numaralarını (sıralarını) bilmemiz gerekir.
        farkTablosu(k,2)=i;
        farkTablosu(k,3)=j;
        k=k+1;
    end
end
end
%%
[minn,indis]=min(farkTablosu(:,1)); % en küçük farka sahip olan olan
özniteliğin değeri ve indisi bulundu.
fprintf('En küçük değer:%f Sırası:%d r1 sırası:%d r3
sırası:%d\n',minn,indis,farkTablosu(indis,2),farkTablosu(indis,3));

% minimum farka sahip öznitelik çiftinin buldukları resimlerdeki
% satır,sutun lokasyonları hesaplanıyor.
r1x=r1points.Location(farkTablosu(indis,2),1);
r1y=r1points.Location(farkTablosu(indis,2),2);
r3x=r3points.Location(farkTablosu(indis,3),1);
r3y=r3points.Location(farkTablosu(indis,3),2);
```



AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ



```
%Burada lokasyonlar arası vektörel uzaklık hesaplanıyor.  
mesafe=sqrt((r1x-r3x)^2+(r1y-r3y)^2);  
fprintf('Mesafe=%f\n',mesafe)
```

2.) 'videoA.mp4' ve 'videoB.mp4' isimli iki adet video dosyası bulunmaktadır. 'videoA.mp4' dosyasının toplam çerçeve sayısı:250, 'videoB.mp4' dosyasının toplam çerçeve sayısı:350'dir. Buna göre:

- Her bir video dosyasının tüm çerçevelerinin renk kanalları ayrı ayrı medyan filtresinden geçirilecek,
- Sonra bu kanallar birleştirilip RGB resim çerçeveleri tekrar oluşturulacak,
- Birleştirilen RGB resim çerçeveleri mantıksal OR işlemine tabi tutulacak,
- Son olarak 'OR' lanan görüntü çerçeveleri bir figüre penceresinden ardışık olarak oynatılacaktır. (40p)

Yanıt:

```
clc;clear;close all;  
filename1='video1.mp4'; %275 çerçeve 1. video  
mov1=VideoReader(filename1);  
filename2='video2.mp4';%444 çerçeve 2. video  
mov2=VideoReader(filename2);  
artis=2; % kaç çerçevede bir okunacağını belirliyoruz.  
%%  
%okuma işlemi for döngüsünde başlıyor.  
for k=1:artis:mov1.NumberOfFrames;  
r1=read(mov1,k); %k. çerçeveyi oku  
r1Medyan_R=medfilt2(r1(:,:,1));  
r1Medyan_G=medfilt2(r1(:,:,2));  
r1Medyan_B=medfilt2(r1(:,:,3));  
r1=cat(3,r1Medyan_R,r1Medyan_G,r1Medyan_B);  
r2=read(mov2,k);%2. video çerçeveleri filtreleniyor  
r2Medyan_R=medfilt2(r2(:,:,1));  
r2Medyan_G=medfilt2(r2(:,:,2));  
r2Medyan_B=medfilt2(r2(:,:,3));  
r2=cat(3,r2Medyan_R,r2Medyan_G,r2Medyan_B);  
r=bitor(r1,r2);% 1. ve 2. video çerçeveleri mantıksal OR'lanıyor..  
figure(1);  
imshow(r);  
fprintf('Frame No=%d \n',k);  
end
```



AKÜ TEKNOLOJİ FAKÜLTESİ
MEKATRONİK MÜHENDİSLİĞİ BÖLÜMÜ



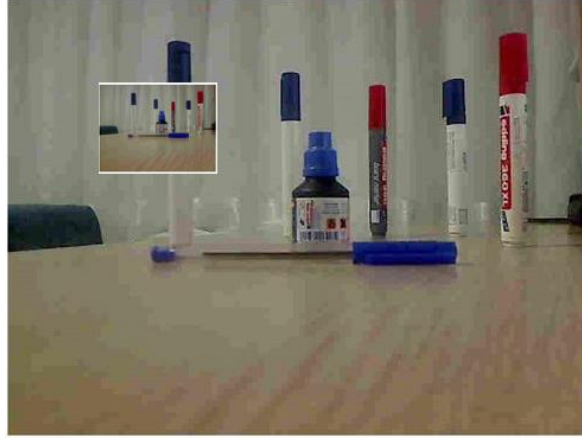
3.) olcekleYerlestir.m isimli bir fonksiyon yazılacaktır. Bu fonksiyon parametre olarak verilen bir görüntü matrisini verilen ölçekte küçülterek yine aynı resimde bulunan bir koordinata yerleştirmektedir. Yerleştirme sırasında taşmalar için önlem alınacaktır. Bu fonksiyonun 4 adet parametresi bulunmaktadır. Bunlar:

x : görüntü matrisi,
olcek : küçültme oranı,
satir : küçültülen resmin yerleştirileceği konumun satır numarası
sutun : küçültülen resmin yerleştirileceği konumun sütun numarası

Sonuçta çalıştırılacak fonksiyon ve oluşan görüntünün bir örneği aşağıda verilmiştir. (40p)

>>Y=olcekleYerlestir(x, 0.2, 80, 80);

>>imshow(Y);



```
function y = FinalS_3( x,olckle,satir,sutun )
z=imresize(x,olckle);
if size(x,3)>1 %rgb mi yoksa gri seviye mi kontrol ediliyor.
if satir+size(z,1)<size(x,1) && sutun+size(z,2)<size(x,2)%taşma var mı diye bakılıyor.
x(satir:satir+size(z,1)-1,sutun:sutun+size(z,2)-1,1)=z(:, :, 1);%taşma yok ise her renk kanalı için aktarma yapılıyor.
x(satir:satir+size(z,1)-1,sutun:sutun+size(z,2)-1,2)=z(:, :, 2);%taşma yok ise her renk kanalı için aktarma yapılıyor.
x(satir:satir+size(z,1)-1,sutun:sutun+size(z,2)-1,3)=z(:, :, 3);%taşma yok ise her renk kanalı için aktarma yapılıyor.
else
fprintf('Taşma meydana gelmiştir.\n');
end
y=x;
else
%resim gri seviye ise tek kanal aktarılıyor.
if satir+size(z,1)<size(x,1) && sutun+size(z,2)<size(x,2)
x(satir:satir+size(z,1)-1,sutun:sutun+size(z,2)-1)=z;
else
fprintf('Taşma meydana gelmiştir.\n');
end
y=x;
end
end
```