

# سیگنال و سیستم ها

810100119

پروژه 2

سهیل حاجیان منش

استاد: سعید اخوان

**بخش اول)** عکس کد های نوشته شده مربوط به هر تابع و اسکریپت p1.m را در زیر قرار داده ام:

**Mygrayfun:**

```
function grayPic=mygrayfun(picture)
for i=1:size(picture,1)
    for j=1:size(picture,2)
        grayPic(i,j)=(0.299*picture(i,j,1))+
            (0.578*picture(i,j,2))+(0.114*picture(i,j,3));
    end
end
end
```

در دو حلقه تو در تو هر پیکسل از عکس را با استفاده از عملیات گفته شده به یک عنصر از ماتریس دوبعدی نگاشت میکنیم.

**Mybinaryfun:**

```
function binaryPic=mybinaryfun(picture,treshold)
for i=1:size(picture,1)
    for j=1:size(picture,2)
        if picture(i,j)>treshold
            binaryPic(i,j)=1;
        else
            binaryPic(i,j)=0;
        end
    end
end
end
end
```

باز هم در یک حلقه تو در تو هر خونه از ماتریس مربوط به عکس خاکستری را به عدد یک و یا صفر نگاشت میکنیم. (اگر بزرگتر از threshold بود یک و در غیر اینصورت صفر میگذاریم).

### Myremovecom:

در واقع روی تمام نقاط الگوریتم DFS میزنیم و تمام نقطه های همسایه را پیدا میکنیم و در متغیر objects ذخیره میکنیم. سپس آن ابعکت هایی را که تعداد پیکسل هایشان کمتر از n میباشد را از objects حذف میکنیم. به این ترتیب تصویر تمیز میشود.

```
function cleanPic=myremovecom(picture,n)
    [row,column]=find(picture==1);
    points=[row';column'];
    objNumber=1;
    while size(points,2)>0
        startPoint=points(:,1);
        points(:,1)=[];
        [points,adjPoints]=findAdjencyPoints(startPoint,points);
        currentObject=[startPoint adjPoints];
        while size(adjPoints,2)>0
            newAdjs=[];
            for i=1:size(adjPoints,2)
                [points,newpoints]=findAdjencyPoints(adjPoints(:,i),points);
                newAdjs=[newAdjs newpoints];
            end
            currentObject=[currentObject newAdjs];
            adjPoints=newAdjs;
        end
        objects{objNumber}=currentObject;
        objNumber=objNumber+1;
    end
    for i=1:objNumber-1
        if(size(objects{i},2)<n)
            index=sub2ind(size(picture),objects{i}(1,:),objects{i}(2,:))
            picture(index)=0;
        end
    end
    cleanPic=picture;
end
```

### Mysegmentation:

در این قسمت هم مشابه تابع قبلی با استفاده از DFS object های مختلف موجود در تصویر سیاه سفید را می یابیم . ماتریس label با ابعاد برابر تصویر سیاه و سفید ایجاد میکنیم و ایندکس های مربوط به هر object را برابر شماره آن object قرار میدهیم. تعداد کل object های موجود در تصویر و همچنین متغیر label را به عنوان خروجی تابع درنظر میگیریم.

```

function [label,numbers]=mysegmentation(picture)
    [row,column]=find(picture==1);
    points=[row';column'];
    objNumber=1;
    while size(points,2)>0
        startPoint=points(:,1);
        points(:,1)=[];
        [points,adjPoints]=findAdjencyPoints(startPoint,points);
        currentObject=[startPoint adjPoints];
        while size(adjPoints,2)>0
            newAdjs=[];
            for i=1:size(adjPoints,2)
                [points,newpoints]=findAdjencyPoints(adjPoints(:,i),points);
                newAdjs=[newAdjs newpoints];
            end
            currentObject=[currentObject newAdjs];
            adjPoints=newAdjs;
        end
        objects{objNumber}=currentObject;
        objNumber=objNumber+1;
    end
    label=zeros(size(picture));
    for i=1:objNumber-1
        index=sub2ind(size(picture),objects{i}(1,:),objects{i}(2,:))
        label(index)=i;
    end
    numbers=objNumber-1;
end

```

در ادامه کد که در زیر قرار داده ام و در فایل **p1.m** موجود است، **mapset** را از روی فایل موجود در پوشه میخوانیم و سپس به ازای هر **object** یک حلقه روی تمامی حروف و اعداد میزنیم و **correlation** آن را با تمامی حروف حساب میکنیم و سپس ماکزیمم **correlation** هارا حساب کرده و اگر از مقدار کفی که تعیین کرده ایم بیشتر بود آن عنصر از **mapset** را بعنوان عضوی از پلاک به جواب اضافه میکنیم.

```

clc
close all;
clear;
[fileName,path]=uigetfile({'*.jpeg;*.jpg;*.bmp;*.png;*.tif'}, 'Choose an Image');
picture=imread([path fileName]);
picture=imresize(picture,[300 500]);
picture=mygrayfun(picture);
picture=~mybinaryfun(picture,100);
cleanPic=myremovecom(picture,350);
backGround=myremovecom(picture,2300);
picture=logical(cleanPic-backGround);
[label,numberOfObjects]=mysegmentation(picture);
figure;
imshow(picture);
folderPath='p1\MapSet';
files=dir(folderPath);
mapSet=cell(2,size(files,1)-2);
for i=3:size(files)
    mapSet{1,i-2}=imread(fullfile(folderPath,files(i).name));
    mapSet{2,i-2}=files(i).name(1);
end
resultString=[];
minCorrVal=0.6;
for i=1:numberOfObjects
    [row,col]=find(label==i);
    currentObject=picture(min(row):max(row),min(col):max(col));
    currentObject=imresize(currentObject,[42 24]);
    corrVals=zeros(1,size(mapSet,2));
    for j=1:size(mapSet,2)
        corrVals(j)=corr2(currentObject,mapSet{1,j});
    end
    [maxCorrVal,idx]=max(corrVals);
    if(maxCorrVal>minCorrVal)
        resultString=[resultString mapSet{2,idx}];
    end
end
fprintf('%s',resultString);
file = fopen('PlateNumber.txt', 'wt');
fprintf(file, '%s\n',resultString);
fclose(file);

```

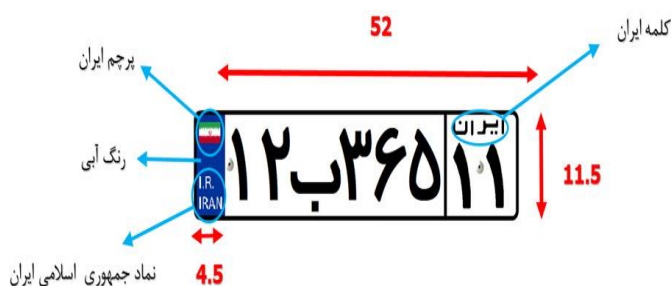
**بخش دوم)** در این قسمت mapSet حروف و اعداد فارسی را در کنار کد قرار داده و از آنها استفاده میکنیم باقی کد مشابه قسمت قبل است و تنها در بعضی جاها از تابع های آماده متلب استفاده میکنیم .

```
clc
close all;
clear;
[fileName,path]=uigetfile({'*.jpeg;*.jpg;*.bmp;*.png;*.tif'}, 'Choose an Image');
picture=imread([path fileName]);
picture=imresize(picture,[300 500]);
picture=rgb2gray(picture);
threshold=graythresh(picture)*255;
picture=~mybinaryfun(picture,threshold-4);
cleanPic=bwareaopen(picture,200);
backGround=bwareaopen(picture,7000);
picture=logical(cleanPic-backGround);
[label,numberOfObjects]=bwlabel(picture);
folderPath='p2\MapSet';
files=dir(folderPath);
mapSet=cell(2,size(files,1)-2);
for i=3:size(files)|
    mapSet{1,i-2}=imread(fullfile(folderPath,files(i).name));
    mapSet{2,i-2}=files(i).name(1);
end
resultString=[];
minCorrVal=0.7;
for i=1:numberOfObjects
    [row,col]=find(label==i);
    currentObject=picture(min(row):max(row),min(col):max(col));
    currentObject=imresize(currentObject,[42 24]);
    corrVals=zeros(1,size(mapSet,2));
    for j=1:size(mapSet,2)
        corrVals(j)=corr2(currentObject,mapSet{1,j});
    end
    [maxCorrVal,idx]=max(corrVals);
    if(maxCorrVal>minCorrVal)
        resultString=[resultString mapSet{2,idx}];
    end
end
fprintf('%s',resultString);
file = fopen('PlateNumber.txt', 'wt');
fprintf(file, '%s\n',resultString);
fclose(file);
```

**بخش سوم)** ابتدا با دستور **uigetfile** ابتدا عکس را میگیریم. سپس به دنبال آبی های موجود در

شکل میگردیم و از آنجایی که ممکن است چند مستطیل که رنگ آبی دارند تشخیص دهد و ما به دنبال آبی خاصی که در پلاک می باشد هستیم با گذاشتن **threshold** رنگ آبی پالک را که خاص تر و کمی پر رنگ تر است را فیلتر می کنیم.

سپس ما بخش آبی پلاک را در متغیر **bluestrip** نگه میداریم. این متغیر به ماتریس یک در چهار است که به صورت **[left up width height]** نگه داری میشود با استفاده از نسبت های پلاک که مشابه شکل زیر است :



البته کمی عدد هارا بیشتر میگیریم و همچنین از چپ و راست مقداری کم میکنیم تا از خطای ممکنه در شناسایی پلاک جلوگیری کنیم.

مراحل گفته شده در شکل از ابتدا تا انتهای **If** اول میباشد.

سپس در باقی بخش ها مشابه قسمت های **1** و **2** پلاک پیدا شده را حروفش را شناسایی میکنیم.

کد سوال را در صفحه بعد آورده ام.

```

[fileName,path]=uigetfile({'*.jpeg;*.jpg;*.bmp;*.png;*.tif'}, 'Choose an Image');
picture=imread([path fileName]);
diffImg = imsubtract(picture(:,:,3), rgb2gray(picture));
%Use a median filter to filter out noise
diffImg = medfilt2(diffImg, [3 3]);
diffImg = imbinarize(diffImg,0.28);
diffImg = bwareaopen(diffImg,300);
bw = bwlabel(diffImg, 8);
stats = regionprops(bw, 'BoundingBox', 'Centroid');
if ~isempty(stats)
    for object = 1:length(stats)
        blueStrip = stats(object).BoundingBox;
        blueStrip(1)=blueStrip(1)-20;
        blueStrip(2) = blueStrip(2)-250;
        blueStrip(3)=blueStrip(3)*16;
        blueStrip(4)=blueStrip(4)+300;
    end
    I2 = imcrop(picture,blueStrip);
end
picture=I2;
picture=imresize(picture,[300 500]);
picture=rgb2gray(picture);
threshold = graythresh(picture);
picture =~imbinarize(picture,threshold);
picture = bwareaopen(picture,220);
background=bwareaopen(picture,7000);
picture=picture-background;
[label,numbreOfObjects]=bwlabel(picture);
folderPath='p2\MapSet';
files=dir(folderPath);
mapSet=cell(2,size(files,1)-2);
for i=3:size(files)
    mapSet{1,i-2}=imread(fullfile(folderPath,files(i).name));
    mapSet{2,i-2}=files(i).name(1);
end
resultString=[];
for n=1:numbreOfObjects
    [r,c] = find(label==n);
    Y=picture(min(r):max(r),min(c):max(c));
    Y=imresize(Y,[42,24]);
    corrVals=zeros(1,size(mapSet,2));
    for k=1:size(mapSet,2)
        corrVals(k)=corr2(mapSet{1,k},Y);
    end
    [MaxCorr,idx]=max(corrVals);
    if MaxCorr>0.7
        resultString=[resultString cell2mat(mapSet(2,idx))];
    end
end
file = fopen('number_Plate.txt', 'wt');
fprintf(file,'%s\n',resultString);
fclose(file);

```