

ارشیا عطایی نایینی ۸۱۰۱۰۰۲۵۲

گزارش تمرین کامپیوتری دوم

(بخش اول)

۱- ابتدا عکس را گرفته و در ماتریس `picture` ذخیره می‌کنیم.

۲- تصویر را با ابعاد گفته شده `resize` می‌کنیم.

۳- با استفاده از مقدارهای گفته شده عکس را خاکستری می‌کنیم.

```
function [gray_pic] = mygrayfun(picture)
    gray_pic = 0.299 * picture(:,:,1) + 0.578 * picture(:,:,2) + 0.114 * picture(:,:,3);
end
```

۴- در این بخش با استفاده از عکس خاکستری شده، آن را باینری می‌کنیم به طوری که سفیدها برابر ۱ و سیاه‌ها برابر ۰ هستند.

```
function [binary_pic] = mybinaryfun(picture, threshold)
    binary_pic = im2double(picture) > threshold;
end
|
```

تصاویر این ۴ قسمت به شرح زیر است:



۵- برای این بخش و پیدا کردن مولفه‌ها الگوریتم **bfs** را بر روی آن اجرا می‌کنیم به این صورت که هر پیکسلی که تا الان در مولفه‌ای قرار نگرفته است را در یک صف ریخته و هر دفعه سر صف را برداشته و همسایه‌های آن را به صف اضافه می‌کنیم.

```
function [rem_pic] = myremovecom(picture, n)
    dis = zeros([300, 500]);
    ischange = zeros([300, 500]);
    cnt = 0;
    INF = 1000000;
    dis(:, :) = INF;
    delta_x = [-1, 0, 1, -1, 1, -1, 0, 1];
    delta_y = [-1, -1, -1, 0, 0, 1, 1, 1];
    que = zeros([2, 300*500]);
    for i=1:300
        for j=1:500
            end
            if (dis(i, j) == INF)
                pnt = 1;
                sz = 1;
                dis(i, j) = 0;
                que(1, 1) = i;
                que(2, 1) = j;
                while (pnt <= sz)
                    x = que(1, pnt);
                    y = que(2, pnt);
                    pnt = pnt + 1;
                    for k=1:8
                        nx = x + delta_x(k);
                        ny = y + delta_y(k);
                        if (isval(nx, ny) == 1 && dis(nx, ny) > dis(x, y) + 1 && picture(x, y) == picture(nx, ny))
                            dis(nx, ny) = dis(x, y) + 1;
                            que(1, sz + 1) = nx;
                            que(2, sz + 1) = ny;
                            sz = sz + 1;
                        end
                    end
                end
            end
        end
    end
end
```

```

        pnt = pnt + 1;
        for k=1:8
            nx = x + delta_x(k);
            ny = y + delta_y(k);
            if (isval(nx, ny) == 1 && dis(nx, ny) > dis(x, y) + 1 && picture(x, y) == picture(nx, ny))
                dis(nx, ny) = dis(x, y) + 1;
                que(1, sz + 1) = nx;
                que(2, sz + 1) = ny;
                sz = sz + 1;
            end
        end
    end
    if (sz < n)
        for k=1:sz
            ischange(que(1, k), que(2, k)) = 1;
        end
    end
end
end
end
for i=1:300
    for j=1:500
        if (ischange(i, j) == 1)
            rem_pic(i, j) = 0;
        else
            rem_pic(i, j) = picture(i, j);
        end
    end
end
end
end
end

```

۶- برای این بخش نیز دقیقا از همان الگوریتم قبلی استفاده کردیم با این تفاوت که فقط بر روی نقاط سفید گروه‌بندی انجام می‌شود و در نهایت لیبلینگ انجام می‌شود.

```

function [L, Ne] = mysegmentation(picture)
    size = 8;
    dis = zeros([300, 500]);
    INF = 1000000;
    dis(:, :) = INF;
    %delta_x = [0, -1, 0, 1, -1, 1, -1, 1];
    %delta_y = [-1, 0, 1, 0, -1, -1, 1, 1];
    delta_x = [-1, 0, 1, -1, 1, -1, 0, 1];
    delta_y = [-1, -1, -1, 0, 0, 1, 1, 1];
    que = zeros([2, 300*500]);
    Ne = 0;
    L = zeros([300, 500]);
    for j=1:500
        for i=1:300
            if (dis(i, j) == INF && picture(i, j) == 1)
                pnt = 1;
                sz = 1;
                dis(i, j) = 0;
                que(1, 1) = i;
                que(2, 1) = j;
                while (pnt <= sz)
                    x = que(1, pnt);
                    y = que(2, pnt);
                    % x
                    % y
                    pnt = pnt + 1;
                    for k=1:size
                        nx = x + delta_x(k);
                        ny = y + delta_y(k);
                        if (isval(nx, ny) == 1 && dis(nx, ny) > dis(x, y) + 1 && picture(x, y) == picture(nx, ny))
                            dis(nx, ny) = dis(x, y) + 1;
                            que(1, sz + 1) = nx;
                            que(2, sz + 1) = ny;
                            sz = sz + 1;
                        end
                    end
                end
                Ne = Ne + 1;
                for k=1:sz
                    L(que(1, k), que(2, k)) = Ne;
                end
            end
        end
    end
end
end
end
end

```

۷- در این بخش ابتدا map set را می‌خوانیم و در TRAININGSET میریزیم حال با استفاده از correlation

تطابق‌ها را بدست می‌آوریم.

تصویر کلی اسکریپت p1.m به شرح زیر است:

```
clc
close all;
clear;
% SELECTING THE TEST DATA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[file,path]=uigetfile({'*.jpg;*.bmp;*.png;*.tif'},'Choose an image');
s=[path,file];
picture=imread(s);
figure
subplot(2,2,1)
imshow(picture)
picture=imresize(picture,[300 500]);
subplot(2,2,2)
imshow(picture)

picture = mygrayfun(picture);
subplot(2, 2, 3)
imshow(picture)

picture = mybinaryfun(picture, 0.455);
picture = logical(picture);
subplot(2, 2, 4)
imshow(picture)
picture = ~picture;

figure
imshow(picture)

picture = myremovecom(picture, 300);
figure
background = myremovecom(picture, 2500);
picture = picture - background;
imshow(picture)

[L, Ne] = mysegmentation(picture);
[Lm,Nem]=bwlabel(picture);
figure
imshow(picture)
propied=regionprops(L,'BoundingBox');
hold on
for n=1:size(propied,1)
    rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','LineWidth',2)
    propied(n).BoundingBox;
end
hold off

load TRAININGSET;
totalLetters=size(TRAIN,2);

final_output=[];
t=[];
figure
for n=1:Ne
    [r,c]=find(L==n);
    Y=picture(min(r):max(r),min(c):max(c));
    imshow(Y)
    Y=imresize(Y,[42,24]);
    imshow(Y)
    pause(0.2)

    ro=zeros(1,totalLetters);
    for k=1:totalLetters
        ro(k)=corr2(TRAIN{1,k},Y);
    end

    [MAXRO,pos]=max(ro);
    if MAXRO>.45
        out=cell2mat(TRAIN{2,pos});
        final_output=[final_output out];
    end
end
fprintf("%s\n", final_output);
file = fopen('eng_res.txt', 'wt');
fprintf(file, "%s\n", final_output);
fclose(file);
winopen('eng_res.txt')
```



نتیجه بدست آمده:

DL5CH8855



نتیجه بدست آمده:

DL2CAo01311



نتیجه بدست آمده:

UP14CB7145

بخش دوم)

این بخش دقیقا مانند بخش سابق است با این تفاوت که n ورودی به تابع myremovecom برای دقت بهتر تغییر کرده است.

تصویر اسکرپت p2.m به شرح زیر است:

```

clc
close all;
clear;
% SELECTING THE TEST DATA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[file,path]=uigetfile({'*.jpg;*.bmp;*.png;*.tif'},'Choose an image');
s=[path,file];
picture=imread(s);
figure
subplot(2,2,1)
imshow(picture)
picture=imresize(picture,[300 500]);
subplot(2,2,2)
imshow(picture)

picture = mygrayfun(picture);
subplot(2, 2, 3)
imshow(picture)
picture = mybinaryfun(picture, 0.455);
picture = logical(picture);
subplot(2, 2, 4)
imshow(picture)
picture = ~picture;
figure
imshow(picture)

picture = myremovecom(picture, 500);
figure
background = myremovecom(picture,3500);
picture = picture - background;
imshow(picture)

[L, Ne] = mysegmentation(picture);
[Lm,Nem]=bwlabel(picture);
figure
imshow(picture)
propied=regionprops(L, 'BoundingBox');
hold on
for n=1:size(propied,1)
    rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','LineWidth',2)
    propied(n).BoundingBox;
end
hold off

```

```

figure
final_output=[];
t=[];
for n=1:Ne
    [r,c]=find(L==n);
    Y=picture(min(r):max(r),min(c):max(c));
    imshow(Y)
    pause(0.2)

    ro=zeros(1,totalLetters);
    for k=1:totalLetters
        [numRow, numCol] = size(PER_TRAIN{1, k});
        Y = imresize(Y, [numRow, numCol]);
        ro(k)=corr2(PER_TRAIN{1,k},Y);
    end

    [MAXRO,pos]=max(ro);
    if MAXRO>.45
        out=cell2mat(PER_TRAIN(2,pos));
        final_output=[final_output out];
    end
end
fprintf("%s\n", final_output);
file = fopen('per_res.txt', 'wt');
fprintf(file, "%s\n", final_output);
fclose(file);
winopen('per_res.txt')

```

اجرا بر روی تست:



نتیجه:

۲۷۱۳۳۳۲م

بخش سوم)

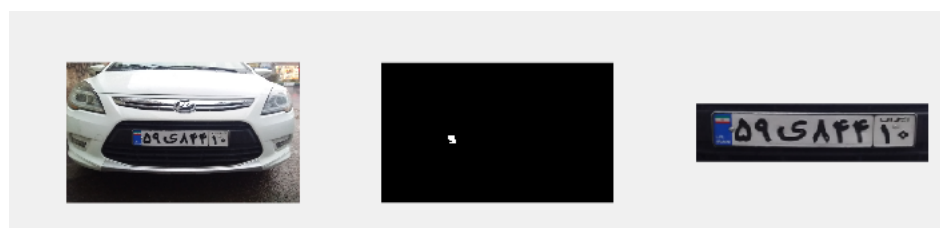
الگوریتم به این صورت کار می کند که عکس را به این روش به عکس باینری تبدیل می کند:

اگر پیکسلی ملایم به آبی باشد (با **threshold** برای رنگ آبی قابل تغییر است)، آن را سفید کرده و در غیر اینصورت سیاه می کنیم و سپس مولفه های خیلی کوچک و خیلی بزرگ سفید را حذف کرده و بین باقی مانده آنی که مایل به وسط است را به عنوان رنگ آبی چپ پلاک در نظر می گیریم. حال اینگونه پلاک را استخراج کرده و در ادامه کد قسمت قبل را اجرا می کنیم.

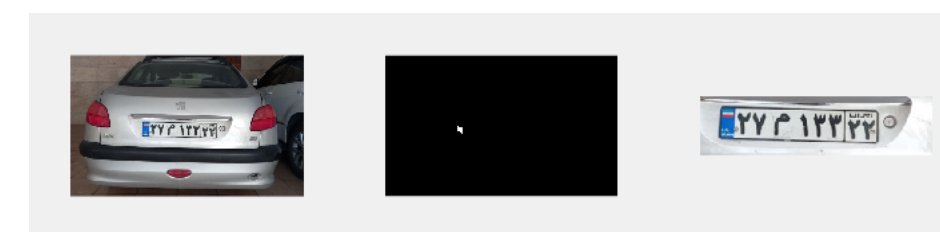
تصویر اسکرپیت پیدا کردن پلاک:

```
clc
close all;
clear;
% SELECTING THE TEST DATA
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
[file,path]=uigetfile({'*.jpg;*.bmp;*.png;*.tif'}, 'Choose an image');
s=[path,file];
picture=imread(s);
figure
picture=imresize(picture,[300 500]);
imshow(picture)
figure
pic_blue = delete_unblue(picture);
imshow(pic_blue)
pic_blue = bwareaopen(pic_blue, 100);
background = bwareaopen(pic_blue, 1000);
pic_blue = pic_blue - background;
figure
imshow(pic_blue)
[L, Ne] = bwlabel(pic_blue);
[r, c] = find(L == 1);
r1 = min(r);
r2 = max(r);
c1 = min(c);
c2 = max(c);
delta_x = 20;
delta_y = 200;
r1 = r1 - delta_x;
r2 = r2 + delta_x;
c1 = c1 - 20;
c2 = c2 + delta_y;
picture2 = picture(r1:r2, c1:c2, :);
figure
imshow(picture2)
picture = picture2;
picture=imresize(picture,[300 500]);
```

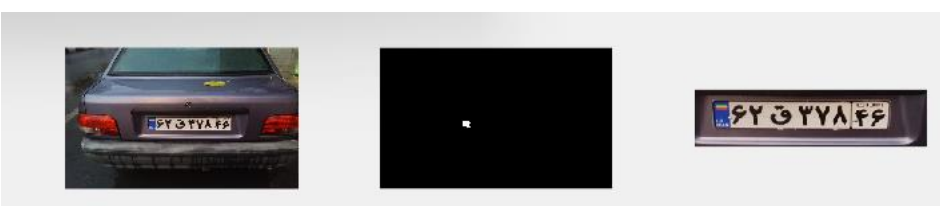
تست آن بر روی نمونه‌ها:



ی ۵۹۸۴۴۱۰



م ۲۷۹۱۳۳۳۲



ق ۶۳۳۷۸۴۶