

شرح ساختار کلی پروژه :

- فایل های بخش اول پروژه :

- main_English.m : در این فایل کد اصلی پروژه قرار دارد که با اجرای این کد در ابتدا پنجره ای باز میشود که بایستی عکس یک پلاک دلخواه را انتخاب کنیم. در ادامه عکس را خاکستری میکنیم و از ۳ بعد به ۲ بعد کاهش میدهیم. سپس عکس را به یک عکس باینری تبدیل میکنیم. سپس background عکس را حذف میکنیم و با حذف سفیدی ها یا لکه های ریز عکس را تمیز میکنیم. سپس عکس را segment بندی کرده و به هر segment یک برچسب میزنیم. در ادامه با استفاده از correlation تصمیم گیری میکنیم که هر segment نشانگر چه حرفی است و در نهایت نتیجه را در فایل number_Plate.txt نوشته و نمایش میدهیم.

- training_loading_English.m : در این فایل با استفاده از عکس های موجود در پوشه English_mpset یک mapset میسازیم که معیار correlation گرفتن در فایل main_English.m همین mapset است.

کد های بخش اول :

```
1      clc;
2      clear;
3      close all;
4
5      di=dir('English_mapset');
6      st={di.name};
7      nam=st(3:end);
8      len=length(nam);
9
10
11     TRAIN=cell(2,len);
12     for i=1:len
13         TRAIN(1,i)={imread(['English_map_set','\ ',cell2mat(nam(i))])};
14         temp=cell2mat(nam(i));
15         TRAIN(2,i)={temp(1)};
16     end
17
18     save('TRAININGSET.mat','TRAIN');
19     clear;
```

training_loading_English.m

```

1      clc
2      close all;
3      clear;
4      load TRAININGSET;
5      totalLetters=size(TRAIN,2);
6
7      % SELECTING THE TEST DATA
8      %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
9      [file,path]=uigetfile({'*.jpg;*.bmp;*.png;*.tif'},'Choose an image');
10     s=[path,file];
11     picture=imread(s);
12     figure
13     subplot(1,2,1)
14     imshow(picture)
15     picture=imresize(picture,[300 500]);
16     subplot(1,2,2)
17     imshow(picture)
18
19     %RGB2GRAY
20     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
21     picture=rgb2gray(picture);
22     figure
23     subplot(1,2,1)
24     imshow(picture)
25
26     % THRESHOLDIG and CONVERSION TO A BINARY IMAGE
27     %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
28     threshold = graythresh(picture);
29     picture = ~im2bw(picture,threshold);
30     subplot(1,2,2)
31     imshow(picture)
32

```

```

33 % Removing the small objects and background
34 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
35 %%
36 figure
37 % picture = bwareaopen(picture,30); % removes all connected components
38 picture = bwareaopen(picture,200);
39 subplot(1,3,1)
40 imshow(picture)
41 background=bwareaopen(picture,3000);
42 subplot(1,3,2)
43 imshow(background)
44 picture2=picture-background;
45 subplot(1,3,3)
46 imshow(picture2)
47 % picture2=bwareaopen(picture2,200);
48 % subplot(1,4,4)
49 % imshow(picture2)
50 %%
53 % Labeling connected components
54 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
55 figure
56 imshow(picture2)
57 [L,Ne]=bwlabel(picture2);
58 propied=regionprops(L,'BoundingBox');
59 hold on
60 for n=1:size(propied,1)
61     rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','LineWidth',2)
62
63 end
64 hold off

```

```

66 % Decision Making
67 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
68 figure
69 final_output=[];
70 t=[];
71 for n=1:Ne
72     [r,c] = find(L==n);
73     Y=picture2(min(r):max(r),min(c):max(c));
74     imshow(Y)
75     Y=imresize(Y,[42,24]);
76     imshow(Y)
77     pause(0.2)
78
79
80     ro=zeros(1,totalLetters);
81     for k=1:totalLetters
82         ro(k)=corr2(TRAIN{1,k},Y);
83     end
84     [MAXRO,pos]=max(ro);
85     if MAXRO>.45
86         out=cell2mat(TRAIN(2,pos));
87         final_output=[final_output out];
88     end
89 end
90
91 % Printing the plate
92 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
93 file = fopen('number_Plate.txt', 'wt');
94 fprintf(file,'%s\n',final_output);
95 fclose(file);
96 winopen('number_Plate.txt')

```

main_English.m

- فایل های بخش دوم پروژه :

- main_Farsi.m : کاملاً مشابه با بخش اول در این فایل کد اصلی پروژه قرار دارد که با اجرای این کد در ابتدا پنجره ای باز میشود که بایستی عکس یک پلاک دلخواه را انتخاب کنیم. در ادامه عکس را خاکستری میکنیم و از ۳ بعد به ۲ بعد کاهش میدهیم. سپس عکس را به یک عکس باینری تبدیل میکنیم. سپس background عکس را حذف میکنیم و با حذف سفیدی ها یا لکه های ریز ، عکس را تمیز میکنیم. در ادامه چون این بار یک عکس کامل از جلوبندی داریم و نه خود پلاک بایستی پلاک را در تصویر تشخیص دهیم. از آن جایی که عکس مان سیاه و سفید است و حروف پلاک در چنین تصویری سفید هستند. میتوان گفت ناحیه ای که پلاک در آن قرار دارد تغییرات زیادی از سیاه به سفید یا از سفید به سیاه دارد. پس ایده تشخیص ناحیه پلاک این است که ببینیم در چه سطر و ستونی از عکس ، تغییرات از سیاه به سفید یا از سفید به سیاه بیشتر است و این سطر و ستون در مکان مناسبی از تصویر است (چون ممکن است سطر یا ستون با بیشینه تغییرات در قسمت های بالا یا پایین عکس باشد). که یافتن مکان مناسب با آزمون و خطا به دست آمد. همچنین میزان این تغییرات را برای هر سطر و ستون جداگانه محاسبه میکنیم. در ادامه برای تشخیص بالا و پایین پلاک از بالای تصویر تا سطری که بیشترین تغییرات را داشته است دنبال اولین سطری میگردیم که میزان تغییراتش نزدیک به بیشینه تغییرات در یک سطر باشد و همچنین در جای مناسبی باشد (خیلی بالاتر از پلاک نباشد چون ممکن است در سطر های بالاتر از سطر با بیشینه تغییرات هم سطرهایی با این ویژگی داشته باشیم) که یافتن محدوده جای مناسب با آزمون و خطا به دست آمد. برای یافتن چپ و راست پلاک هم کاملاً مشابه با یافتن بالا و پایین عمل میکنیم. سپس عکس را segment بندی کرده و به هر segment یک برچسب میزنیم. در ادامه با استفاده از correlation تصمیم گیری میکنیم که هر segment نشانگر چه حرفی است و در نهایت نتیجه را در فایل number_Plate.txt نوشته و نمایش میدهیم.

- make_mapset.m : در قسمت فارسی mapset از ابتدا آماده نیست. به همین خاطر با استفاده از این فایل و عکس هایی که از پلاک ها گرفته ایم ، خودمان mapset را میسازیم. عملکرد این فایل هم به این شکل است که پلاک را سیاه و سفید و segment بندی میکند و در ادامه هر کدام از segment ها را به عکس هایی جداگانه تبدیل میکند. سپس خودمان از بین عکس های به دست آمده حروف و اعداد مناسب mpset را انتخاب میکنیم و باقی عکس ها را حذف میکنیم.

- training_loading_Farsi.m : در این فایل با استفاده از عکس های موجود در پوشه Farsi_mpset یک mapset میسازیم که معیار correlation گرفتن در فایل main_Farsi.m همین mapset است.

```

1  clc;
2  close all;
3  clear;
4
5  [file, path] = uigetfile('*.jpg;*.png;*.jpeg;*.bmp');
6  picture = imread([path, file]);
7  picture = rgb2gray(picture);
8  imshow(picture);
9  threshold = graythresh(picture);
10 picture = ~imbinarize(picture, threshold - 0.2);
11 imshow(picture);
12 imshow(picture);
13 picture = bwareaopen(picture, 60000);
14 imshow(picture);
15 imshow(picture);
16 background = bwareaopen(picture, 4400000);
17 imshow(background);
18 picture = picture - background;
19 imshow(picture);
20 imshow(picture);
21 [L,Ne]=bwlabel(picture);
22 propied=regionprops(L,'BoundingBox');
24 for n=1:size(propied,1)
25     rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','LineWidth',2)
26 end
27
28 for n=1:Ne
29     [r,c] = find(L==n);
30     Y=picture(min(r):max(r),min(c):max(c));
31     imshow(Y)
32     pathStr = sprintf('Farsi_mapset\\%d.bmp', n);
33     imwrite(Y, pathStr);
34 end

```

make_mapset.m

```

1  clc;
2  close all;
3  clear;
4
5  mapset_dir = dir('Farsi_mapset\');
6  col_name = {mapset_dir.name};
7  target_name = col_name(3:end);
8  target_len = length(target_name);
9
10 data_table = cell(2, target_len);
11 for i = 1:target_len
12     data_table(1, i) = {imresize(imread(['Farsi_mapset\' , target_name{i}]), [NaN, 64])};
13     data_table(2, i) = {extractBefore(target_name{i}, ".")};
14 end
15
16 save('Farsi_DATA_TABLE.mat', "data_table");
17 clear;

```

training_loading_Farsi.m

```

1  clc
2  close all;
3  clear;
4
5  load('Farsi_DATA_TABLE.mat');
6
7  [file, path] = uigetfile('*.jpg;*.png;*.jpeg;*.bmp');
8  picture = imread([path, file]);
9  picture = rgb2gray(picture);
10 imshow(picture);
11 threshold = graythresh(picture);
12 picture = ~imbinarize(picture, threshold-0.1);
13 imshow(picture);
14 width = 600;
15 length = 800;
16 picture = imresize(picture, [width, length]);
17 picture = bwareaopen(picture, 60);
18 imshow(picture);
19 picture = picture - bwareaopen(picture, 1670);
20 imshow(picture);

```

```

22     max_rows_changes = 0;
23     max_cols_changes = 0;
24     y_max_changes = 1;
25     x_max_changes = 1;
26     rows_changes_count = zeros(1, width);
27     cols_changes_count = zeros(1, length);
28
29     for i=1: width
30         changes_count = 0;
31         for j=1: length - 1
32             if picture(i, j + 1) ~= picture(i, j)
33                 changes_count = changes_count + 1;
34             end
35         end
36         rows_changes_count(i) = changes_count;
37         if changes_count > max_rows_changes && i > 340 && i < 500
38             max_rows_changes = changes_count;
39             y_max_changes = i;
40         end
41     end
42
43     y_down = width - 100;
44     y_top = 100;
45
46     x_right = length - 100;
47     x_left = 100;
48
49     for i=100: y_max_changes
50         if abs(max_rows_changes - rows_changes_count(i)) < 20 && y_max_changes - i < 50
51             y_top = i;
52             break;
53         end
54     end
55
56     for i=width - 100:-1: y_max_changes
57         if abs(max_rows_changes - rows_changes_count(i)) < 20 && i - y_max_changes < 50
58             y_down = i;
59             break;
60         end
61     end

```



```

63     for j=1: length
64         changes_count = 0;
65         for i=1: width - 1
66             if picture(i + 1, j) ~= picture(i, j)
67                 changes_count = changes_count + 1;
68             end
69         end
70         cols_changes_count(i) = changes_count;
71         if changes_count > max_cols_changes && j > 300 && j < 400
72             max_cols_changes = changes_count;
73             x_max_changes = j;
74         end
75     end
76
77     for j=220: x_max_changes
78         if abs(max_cols_changes - cols_changes_count(j)) < 30 && x_max_changes - j < 230
79             x_left = j;
80             break;
81         end
82     end
83
84     for j=length - 200:-1: x_max_changes
85         if abs(max_cols_changes - cols_changes_count(j)) < 30 && j - x_max_changes < 300
86             x_right = j;
87             break;
88         end
89     end
90
91     delta_y = y_down - y_top;
92
93     if delta_y < 60
94         y_down = y_down + 80 - delta_y;
95     end
96
97     plate = picture(y_top:y_down,x_left:x_right);
98     imshow(plate);
99
100     [L,Ne] = bwlabel(plate);
101     propied = regionprops(L,'BoundingBox');
102     for n=1:size(propied,1)
103         rectangle('Position',propied(n).BoundingBox,'EdgeColor','g','Linewidth',2)
104     end

```

```

106 plate = bwareaopen(plate, 100);
107 imshow(plate);
108
109 final_output=[];
110 t=[];
111
112 totalLetters = size(data_table, 2);
113
114 file = fopen('number_Plate.txt', 'wt');
115
116 for n=1:Ne
117     [r, c] = find(L == n);
118     Y = plate(min(r):max(r),min(c):max(c));
119     imshow(Y)
120     imshow(Y);
121
122     ro = zeros(1, totalLetters);
123     for k=1: totalLetters
124         [row, col] = size(data_table{1,k});
125         Y = imresize(Y, [row, col]);
126         ro(k) = corr2(data_table{1,k},Y);
127     end
128     [MAXRO, pos] = max(ro);
129     if MAXRO>.45
130         out = cell2mat(data_table(2,pos));
131         final_output = [final_output out];
132         fprintf(file,'%s\n', out);
133     end
134 end
135
136 fclose(file);
137
138 winopen('number_Plate.txt')

```

main_Farsi.m

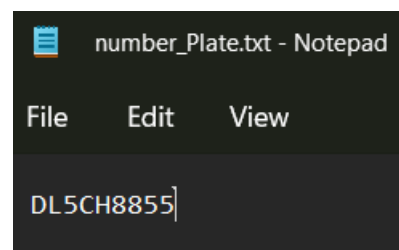
تست بخش اول :



ورودی



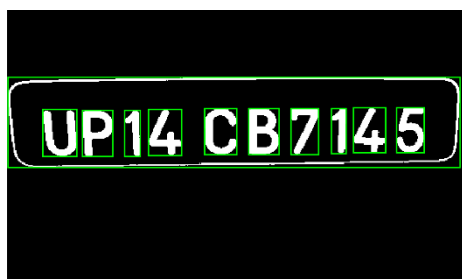
پلاک سگمنت بندی شده



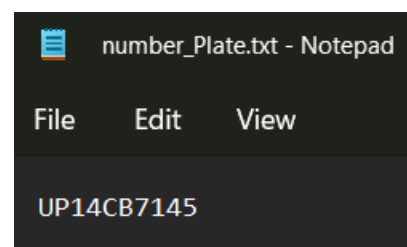
خروجی



ورودی



پلاک سگمنت بندی شده



خروجی

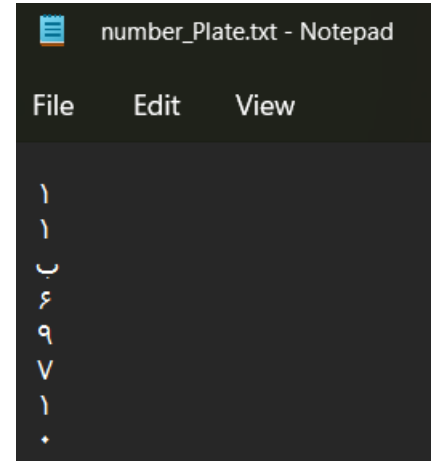
تست بخش دوم :



ورودی



پلاک سگمنت بندی شده



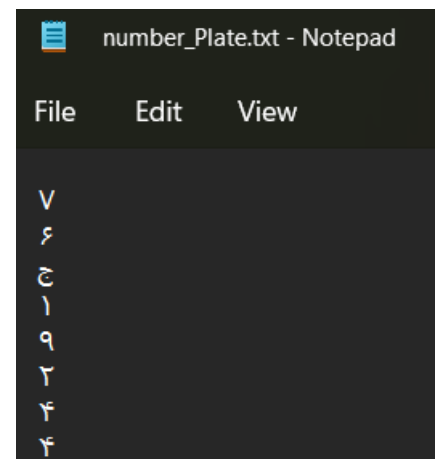
خروجی



ورودی



پلاک سگمنت بندی شده



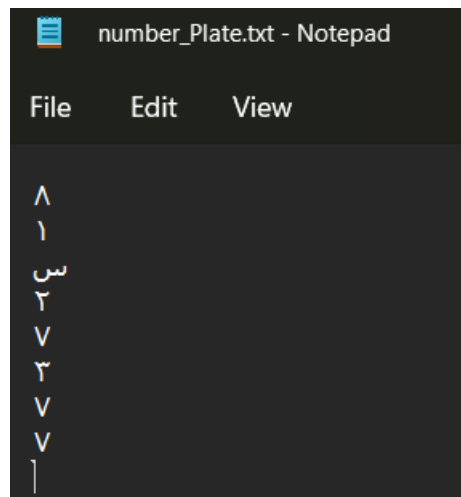
خروجی



ورودی



پلاک سگمنت بندی شده



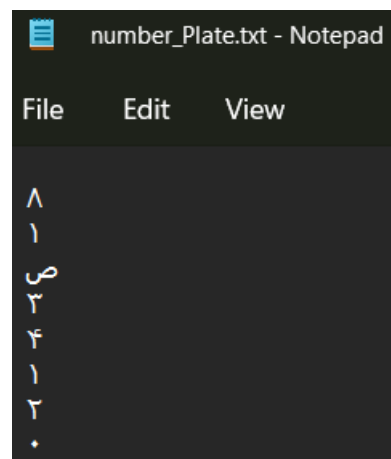
خروجی



ورودی



پلاک سگمنت بندی شده



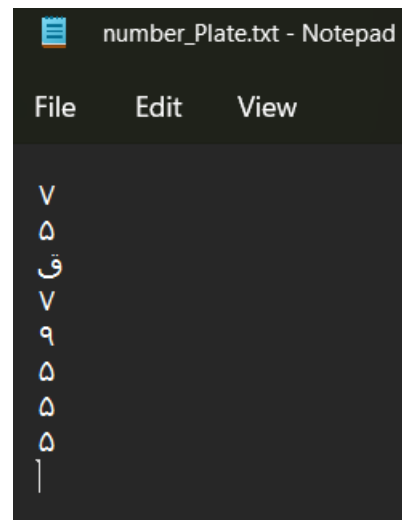
خروجی



ورودی



پلاک سگمنت بندی شده



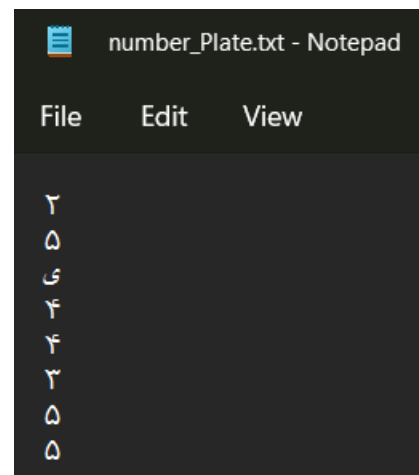
خروجی



ورودی



پلاک سگمنت بندی شده



خروجی