

Signal&Systems : CA#2



تشخیص پلاک

سید مهدی حاجی سید حسین

شماره دانشجویی: 810100118

Contents

۳.....	مقدمه
۴.....	بخش اول (پلاک خارجی)
۱۵.....	جمع آوری دیتاست فارسی.
۱۶.....	بخش دوم (پلاک فارسی)
۲۵.....	بخش سوم (تشخیص پلاک از عکس جلوبند)

First Words:

مقدمه و تشریح پروژه

در این پروژه باید سعی میشد تا با استفاده از ساده سازی پلاک و همچنین کورولیشن گیری میان پلاک و تمامی کاراکتر های موجود ، شماره پلاک را از روی پلاک تشخیص داد و آن را ثبت نمود .

در بخش اول پروژه به تشخیص پلاک های خارجی ، و در بخش دوم این پروژه به تشخیص پلاک ایرانی و همچنین در پارت آخر به تشخیص شماره پلاک از روی جلوبندی ماشین پرداختم .

Part 1:

تشخیص پلاک انگلیسی

در این بخش باید به تشخیص شماره پلاک خارجی از روی عکس پلاک میپردازیم.

قسمت های مطلوب پروژه برای گزارش :

Question 1:

در این قسمت با استفاده از تابع `uigetfile` اقدام به دریافت عکس پلاک نمودم.

```
1  clc;close all;clearvars;
2  % Question 1 :
3  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
4  [file,path]=uigetfile({'*.jpg;*.bmp;*.png;*.tif'},'Please Choose your PELAK ');
5  source = [path file];
6  % source = '/MATLAB Drive/untitled/image1.jpg';
7  picture=imread(source);
8  figure
9  subplot(1,2,1)
10 imshow(picture);
11
12
```

Question 2:

در این قسمت سایز عکس را به ابعاد 300 در 500 تغییر دادم.

```
15 % Question 2 :
16 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
17 picture=imresize(picture,[300 500]);
18 subplot(1,2,2)
19 imshow(picture);
20
21
22
```

Question 3 :

در این قسمت با استفاده از فرمول زیر کانال های رنگی قرمز ، سبز ، آبی را به کانال خاکستری تغییر رنگ دادم .

$$Gray_{channel} = 0.299 \times Red_{channel} + 0.578 \times Green_{channel} + 0.114 \times Blue_{channel}$$

```
24 % Question 3 :  
25 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
26 picture=mygrayfun(picture);  
27 figure  
28 imshow(picture)  
29
```

```
3 function [grayPic] = mygrayfun(pic)  
4 % This function make the pic to gray and white :  
5  
6 % Gray channel is : 0.299 * RedChannel + 0.578 * GreenChannel + 0.114 * BlueChannel  
7  
8     grayPic = 0.299 * pic(:,:,1) + 0.578 * pic(:,:,2) + 0.114 * pic(:,:,3);  
9  
10 end
```



Question 4:

در این پارت با نوشتن تابعی به اسم mybinaryfun اقدام به سیاه سفید نمودن عکس با یک Th مشخص نمودم .

```
33 % Question 4 :  
34 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
35 %T = graythresh(I) computes a global threshold T from grayscale image I, using Otsu's method [1].  
36 % Otsu's method chooses a threshold that minimizes the intraclass variance of the thresholded black and white pixels.  
37 % The global threshold T can be used with imbinarize to convert a grayscale image to a binary image.  
38 th = graythresh(image);  
39  
40  
41 picture= mybinaryfun (picture , 127);  
42 figure  
43 imshow(picture)  
44  
45
```

```
4 function [bpic] = mybinaryfun(grayPic,th)  
5 % This function make the gray pic to white and black using threshold  
6     bpic = grayPic > th;  
7     bpic = ~bpic ;  
8  
9 end
```



Question 5 :

نگاشت بدست آمده در بخش قبل میتواند حاوی مقداری نویز باشد ، برای تمیز کردن این نگاشت می توان سائز تکه های سیاه یا سفید متصل به یکدیگر را پیدا کرده و تکه های با سوائز کووکترا را حذف کرد.

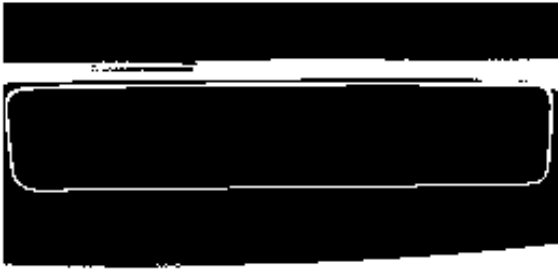
در بخش ۵ قسمت اول پررژه به پیاده سازی تابع bwareaopen پرداختم که ورودی آن عکس و یک حد کمینه برای نویز است . خروجی آن عکس تمیز شده است که مقادیر کمتر از ورودی را حذف و تمیز میکند .

```
47 % Question 5 :  
48 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
49 pixel_numbrer_for_noise = 300 ;  
50 picture_without_noise = myremovecom(picture , pixel_numbrer_for_noise);  
51 figure  
52  
53 subplot(1,2,1)  
54 imshow(picture_without_noise)  
55  
56 pixel_number_for_frame = 2500 ;  
57 frame = myremovecom(picture , pixel_number_for_frame);  
58  
59 subplot(1,2,2)  
60 imshow(frame)  
61  
62 picture = picture_without_noise - frame ;  
63 imshow(picture);  
64  
65  
66 figure  
67 imshow(picture)  
68
```

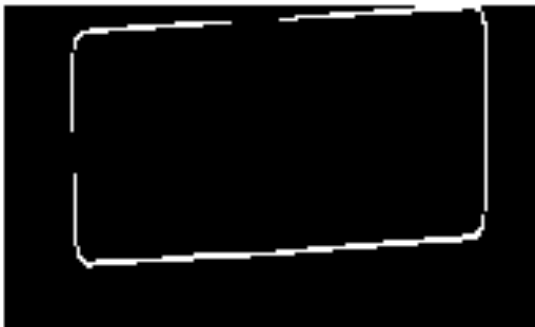
```

1 function [newPIC , OBJECT_NEW]=myremovecom(picture,n)
2 [row,column]=find(picture==1);
3 POINTS=[row';column'];
4 current_obj_num=1;
5 POINTS_NUM=size(POINTS,2);
6
7 while POINTS_NUM>0
8
9     point_ini=POINTS(:,1);
10    POINTS(:,1)=[];
11    [POINTS,newpoints]=close_points(point_ini,POINTS);
12    current_obj=[point_ini newpoints];
13    newpoints_len=size(newpoints,2);
14
15    while newpoints_len>0
16        newpoints2=[];
17        for i=1:newpoints_len
18            [POINTS,newpoints1]=close_points(newpoints(:,i),POINTS);
19            newpoints2=[newpoints2 newpoints1];
20        end
21        current_obj=[current_obj newpoints2];
22        newpoints=newpoints2;
23        newpoints_len=size(newpoints,2);
24    end
25
26    OBJECT{current_obj_num}=current_obj;
27    current_obj_num=current_obj_num+1;
28    POINTS_NUM=size(POINTS,2);
29 end
30 % OBJECT_NEW
31 z=1;
32 current_obj_num=current_obj_num-1;
33 for i=1:current_obj_num
34     if size(OBJECT{i},2)>n
35         OBJECT_NEW{z}=OBJECT{i};
36         z=z+1;
37     end
38 end
39
40 newPIC=zeros(size(picture));
41 for i=1:size(OBJECT_NEW , 2 )
42     ind=sub2ind(size(picture),OBJECT_NEW{i}(1,:),OBJECT_NEW{i}(2,:));
43     newPIC(ind)=1;
44 end

```

UP14 CB7145



DL5CH 8855

Question 6 :

حالا باید اقدام به شماره گذاری اجزای تصویر (component) های آن بکنیم .

الگوریتم این قسمت بدین صورت است که ستون هایی که مقدار همه ی آن ها صفر نیست را در نظر میگیریم ، و حالا باید به دنبال زیر آرایه ای از این آرایه باشیم تا بتوانیم بگویم هر جز تصویر از چه ستونی تا ستونی است . حالا که ستون ابتدایی و انتهایی هر جز را پیدا کردیم کافیهست که مقدار غیر صفر در این ستون هارا به شماره ایندکس جز مقدار دهی کنیم .

```
69 % Question 6 :  
70 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
71 [L,Ne] = mysegmentation(picture);  
72  
73 % [LL,NNe]=bwlabel(picture);  
74 % isequal(L , LL)  
75 propied=regionprops(L, 'BoundingBox');  
76 hold on  
77 for n=1:size(propied,1)  
78     rectangle('Position',propied(n).BoundingBox,'EdgeColor','r','LineWidth',2)  
79 end  
80 hold off  
81  
82
```

```

1 function [LabeledPic , Ne]=mysegmentation(picture)
2     arr = sum(picture);
3     %find subarray that are not zero in them
4     subarrayStart = 0;
5     subarrayEnd = 0;
6     insideSubarray = false;
7     subarrayIndices = [];
8     for i = 1:length(arr)
9         if arr(i) == 0
10             if insideSubarray
11                 subarrayEnd = i - 1;
12                 subarrayIndices = [subarrayIndices; subarrayStart, subarrayEnd];
13                 insideSubarray = false;
14             end
15             else
16                 if ~insideSubarray
17                     subarrayStart = i;
18                     insideSubarray = true;
19                 end
20             end
21         end
22
23         if insideSubarray
24             subarrayEnd = length(arr);
25             subarrayIndices = [subarrayIndices; subarrayStart, subarrayEnd];
26         end
27         % Display the starting and ending indices of subarrays without zeros
28         Ne = size(subarrayIndices , 1);
29         LabeledPic=zeros(size(picture));
30
31         for i=1 :Ne
32             theStart = subarrayIndices(i , 1 );
33             theEnd = subarrayIndices(i , 2);
34
35             % Extract the submatrix from column a to column b
36             %picture(: , theStart:theEnd)
37             [row,column]=find(picture(: , theStart:theEnd)==1);
38
39
40             ind=sub2ind(size(picture),row',(column+theStart-1)');
41             LabeledPic(ind)=i;
42
43         end
44     end

```



Question 7:

در این قسمت با استفاده از correlation گیری با دیتاست حروف و اعداد انگلیسی اقدام به تصمیم گیری راجع به هر یک از حروف پلاک میکنیم.

```
85 % Question 7 :
86 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
87 load TRAININGSET;
88 totalLetters=size(TRAIN,2);
89
90
91 figure
92 final_output=[];
93 t=[];
94 for n=1:Ne
95     [r,c]=find(L==n);
96     Y=picture(min(r):max(r),min(c):max(c));
97
98     Y=imresize(Y,[42,24]);
99
100
101
102
103     ro=zeros(1,totalLetters);
104     for k=1:totalLetters
105         ro(k)=corr2(TRAIN{1,k},Y);
106     end
107
108     [MAXRO,pos]=max(ro);
109     if MAXRO>.45
110         out=cell2mat(TRAIN(2,pos));
111         final_output=[final_output , out];
112     end
113 end
114
115 final_output
116
```

Question 8:

در آخرین قسمت یعنی هشتمین قسمت ، باید عبارت خوانده شده از روی پلاک را درون یک فایل txt. ذخیره سازی کنیم تا بتوان آن را مشاهده نمود .

```
116  
117 % Question 8 :  
118 %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%  
119 file = fopen('number_Plate.txt', 'wt');  
120 fprintf(file, 'Car tag is equal to : \n%s\n', final_output);  
121 fclose(file);  
122  
123
```

```
1 Car tag is equal to :  
2 UP14CB7145  
3
```

```
1 Car tag is equal to :  
2 DL5CH8855  
3
```

Concolution :

همان طور که در دو پلاک نمونه بالا مشاهده شد با دقت ۱۰۰٪ دو پلاک بالا شناسایی شده اند .

Data Integration :

جمع آوری حروف و اعداد فارسی از روی پلاک ماشین ها

ما برای تشخیص پلاک فارسی نیاز به داده هایی از حروف و اعداد پلاک فارسی داریم

. برای این کار از پلاک ماشینهای مختلف عکس گرفته ام و یا از اینترنت آنها را پیدا کرده ام و اعداد و حروف مورد نظر را جدا کرده و اندازه همه آنها را به ابعاد ۵۰ در ۶۰ تغییر داده ام . و با تغییر ابعاد کورولیشن به این میزان اقدام به تصمیم گیری میکنم .

حروف جمع آوری شده شامل :

ب ، ج ، س ، ص ، ط ، ق ، ل ، م ، ن ، ه ، و ، ی

۰ ، ۱ ، ۲ ، ۳ ، ۴ ، ۵ ، ۶ ، ۷ ، ۸ ، ۹ ، ۰

می باشند .

Part 2:

تشخیص پلاک فارسی

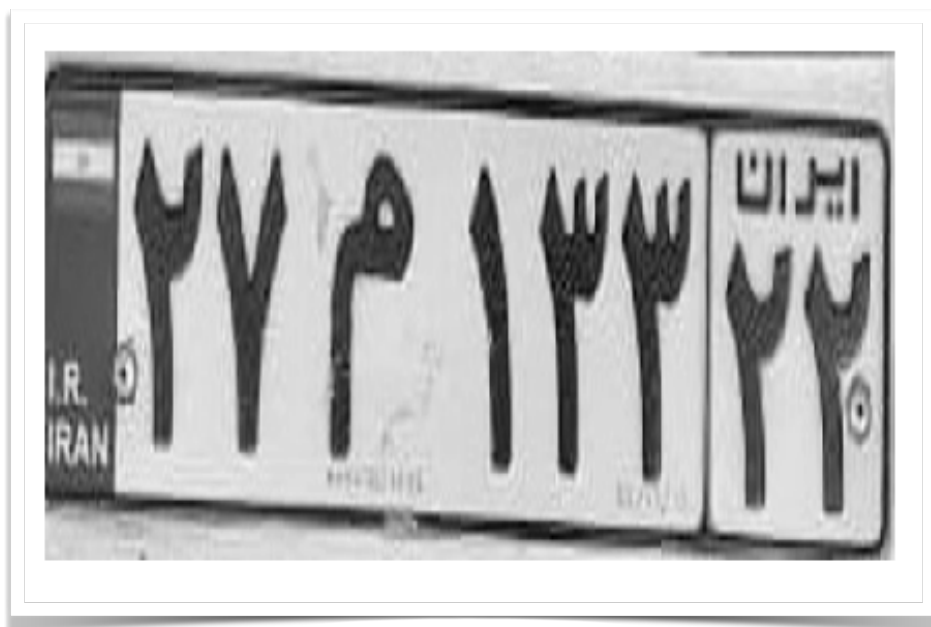
در این قسمت مشابه قسمت قبل از توابع نوشته شده توسط خودم و نه توابع داخلی متلب استفاده نموده ام تا با استفاده از دیتاست فارسی بتوان پلاک های ماشین فارسی را تشخیص داد.

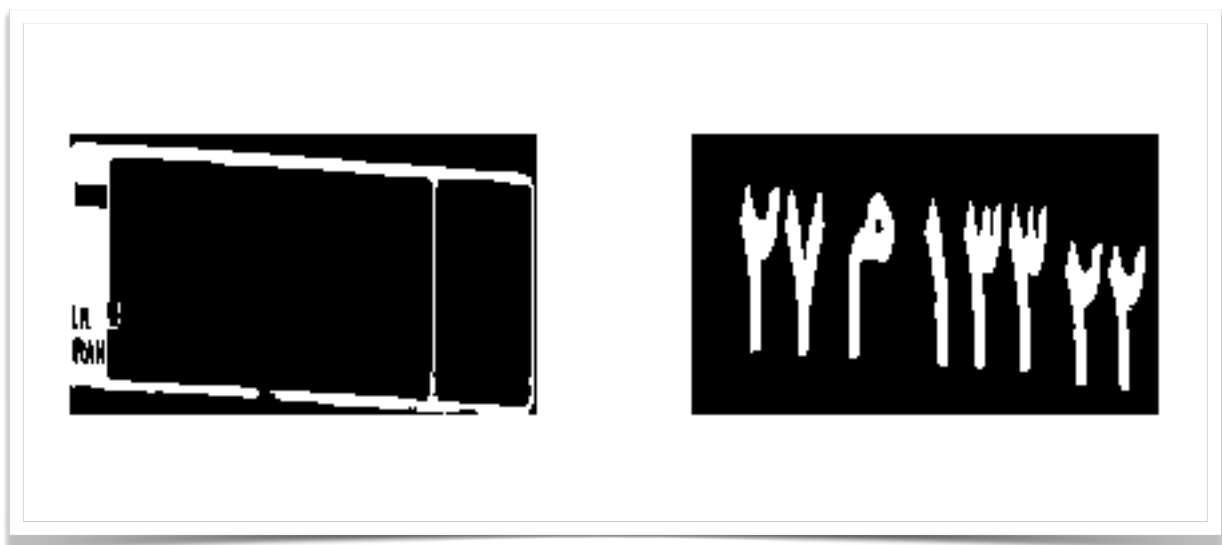
تنها نکته ای که حائز اهمیت است این است که در متلب آنلاین به دلیل نبود فونت فارسی ، حرف الفبای پلاک در آخر متن تشخیصی نوشته میشود و در ابتدای کار تمام اعداد نوشته شده اند .

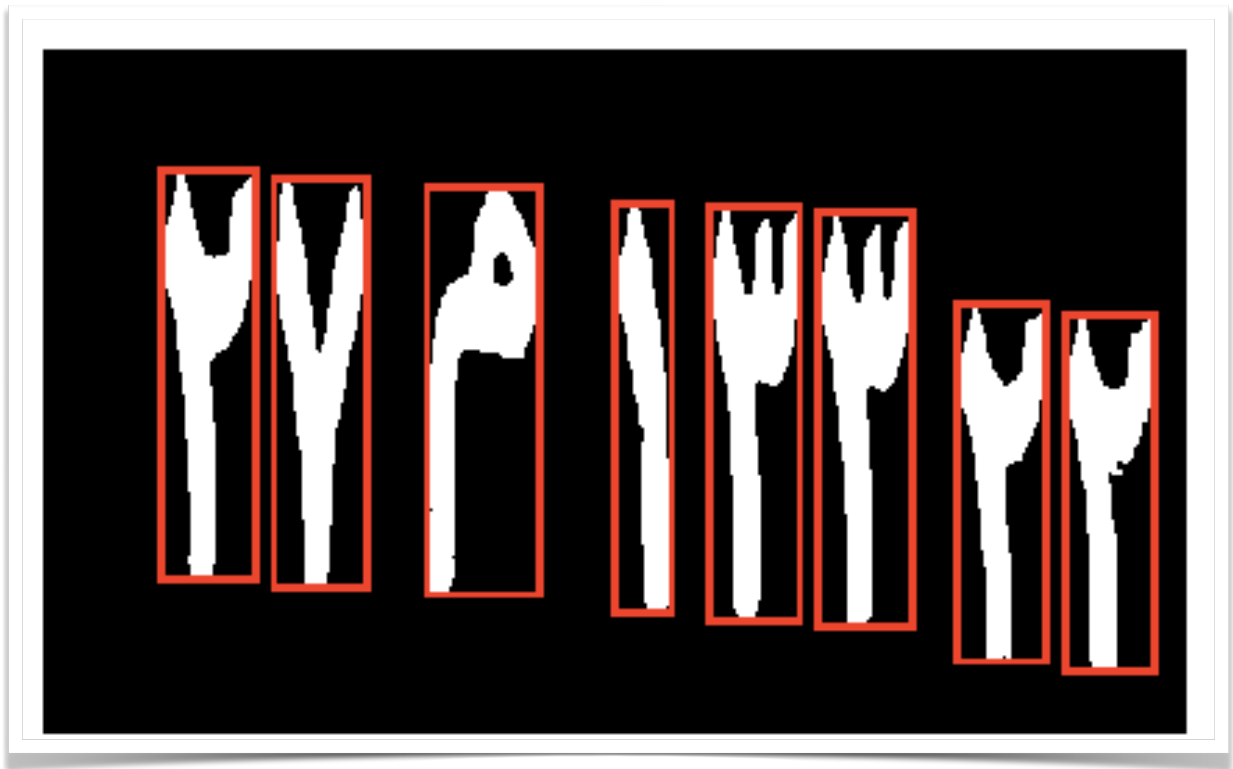
```
1      clc;clear;close all;
2      |
3      di=dir('persianDigits');
4      st={di.name};
5      nam=st(3:end);
6      len=length(nam);
7
8
9      TRAIN=cell(2,len);
10     for i=1:len
11         pic = imread(['persianDigits','/',cell2mat(nam(i))]);
12         %pic = im2gray(pic);
13         threshold = graythresh(pic);
14         pic = im2bw(pic,threshold);
15         TRAIN(1,i)={pic};
16         temp=cell2mat(nam(i));
17         TRAIN(2,i)={temp(1)};
18     end
19
20     save('PERSIANSET.mat','TRAIN');
21     clear;
22
```


Concolution :

نتایج قسمت دوم پروژه به شرح زیر اند :

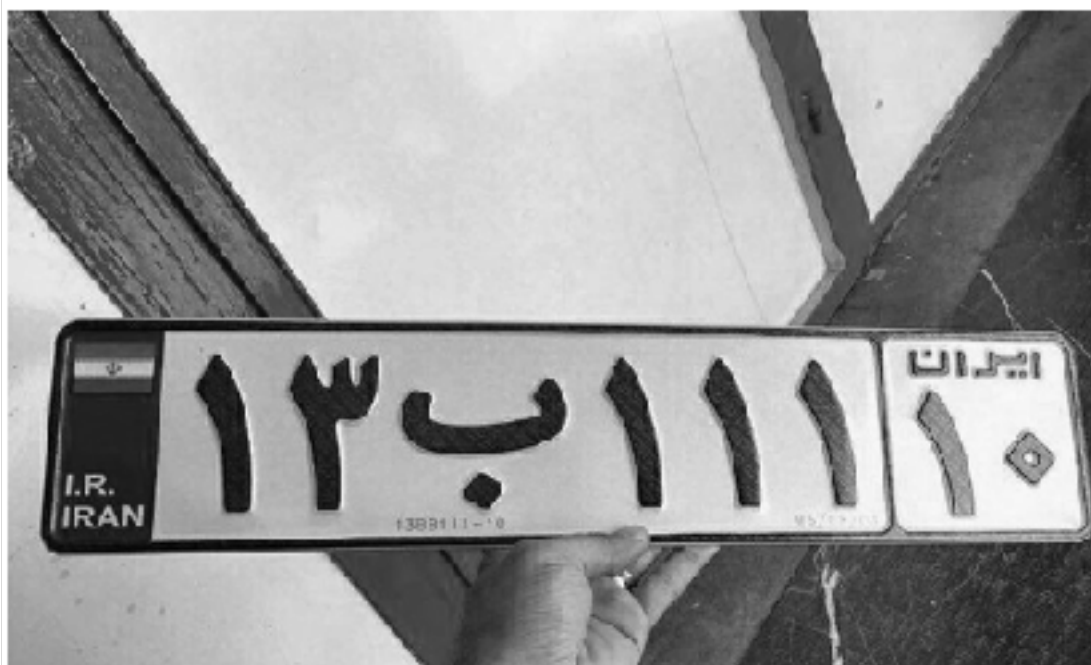


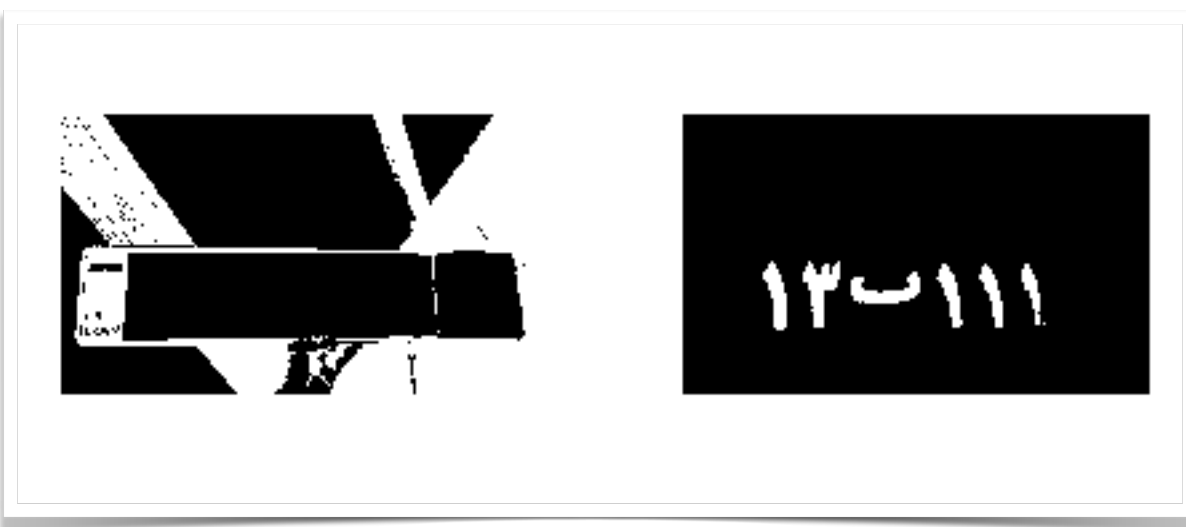


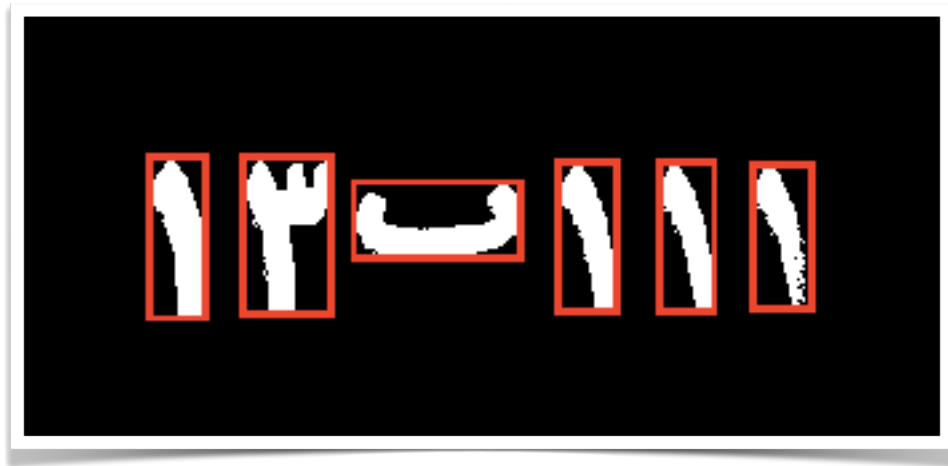


1	Car tag is equal to :
2	3713322 _p
3	

Second Sample :







1	Car tag is equal to :
2	13111
3	

Third Sample :





1	Car tag is equal to :
2	8716112 _S
3	

Part 3 :

تشخیص پلاک فارسی از روی عکس جلوبندی

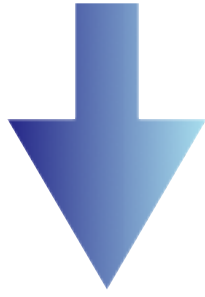
در این قسمت چندین ایده استفاده شده است :

- ❖ ابتدا تصویر را خاکستری و سپس سیاه سفید میکنیم
 - ❖ سپس 1/3 بالایی عکس را crop out میکنیم .
 - ❖ 1/5 راست عکس را نیز crop out میکنیم .
 - ❖ 1/5 چپ عکس را هم crop out میکنیم .
 - ❖ حالا مشابه قسمت های قبل نویز های کوچک و نویز های خیلی بزرگ را حذف میکنیم .
 - ❖ حالا اجزای مانده را با استفاده از bwlable شماره گذاری میکنیم .
 - ❖ سپس با توجه به بررسی هایم نسبت دقیق طول به عرض پلاک که 6 است را لحاظ میکنیم . به این صورت که مستطیل هایی که نسبت طول به عرض آنها میان 4 تا 6 است را از تصویر نگه میداریم و در فولدر JelobandiOUT ذخیره میکنیم .
 - ❖ حالا مشابه قسمت دوم پروژه عکس خروجی در فولدر JelobandiOUT با روش کولولیشین گیری روی دیتاست فارسی ، حدس میزنیم .
- دقت شود که خطوط ۱۶۴ تا ۱۷۰ با توجه به پلاک میتواند uncommnet یا commnet شود تا دقت بالا پایین شود .

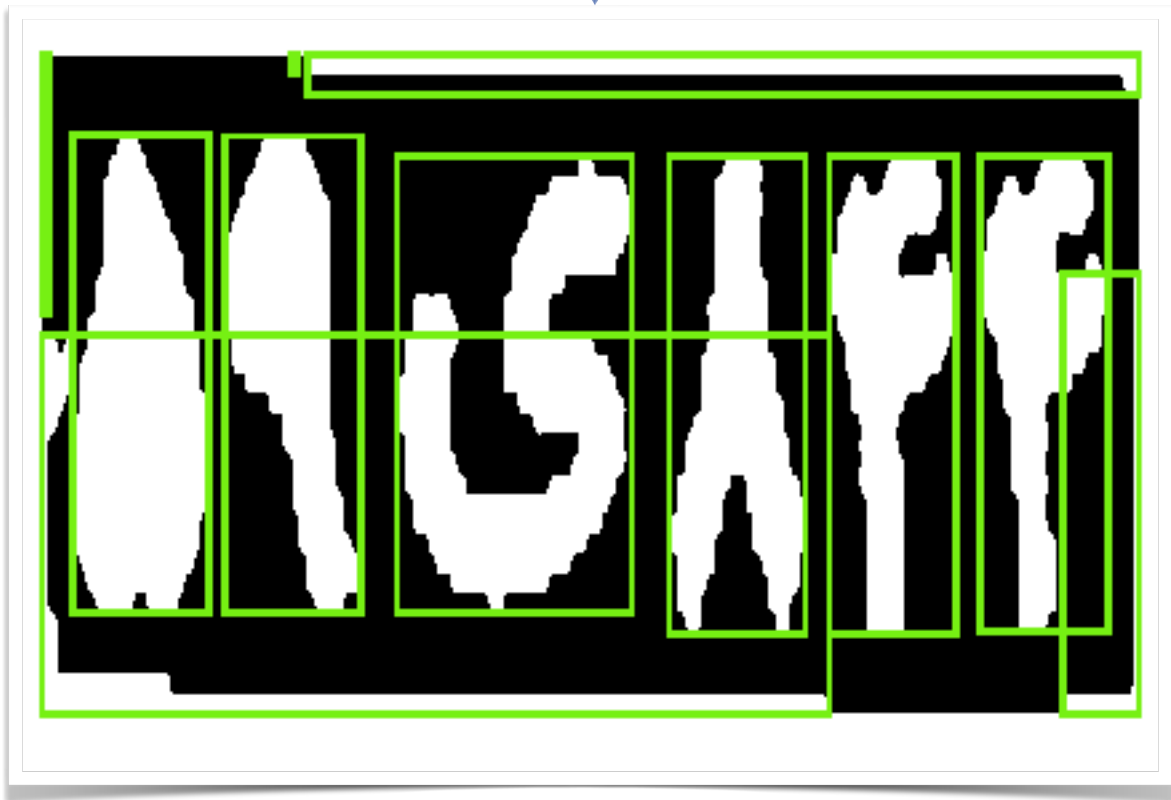
First Sample :







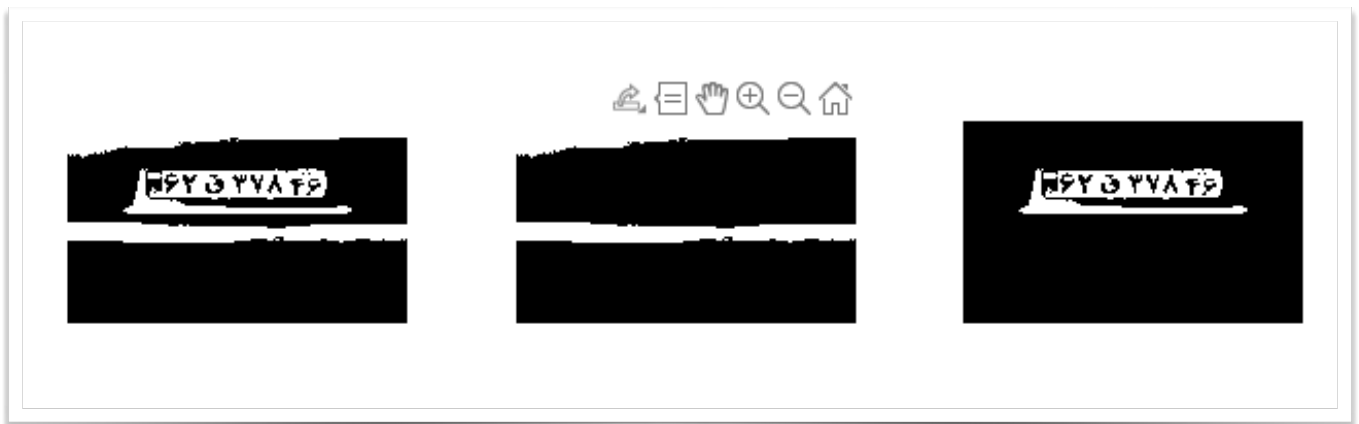
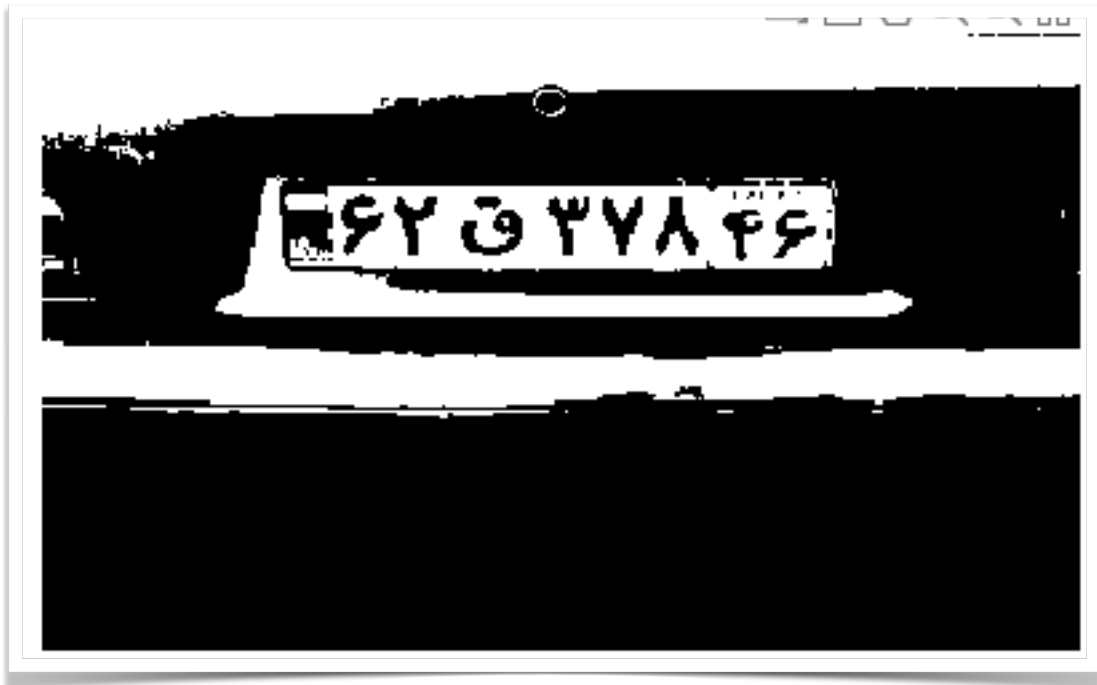


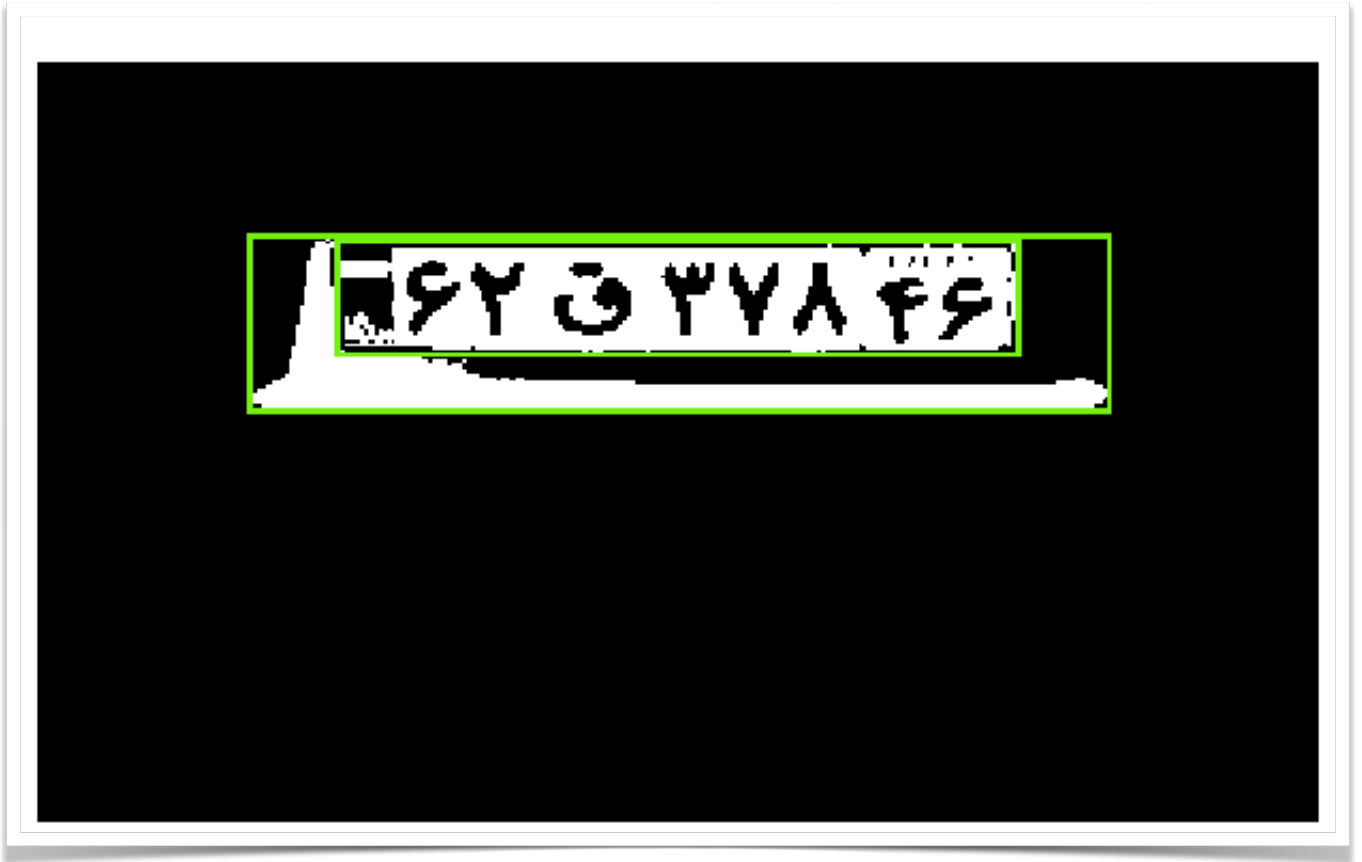


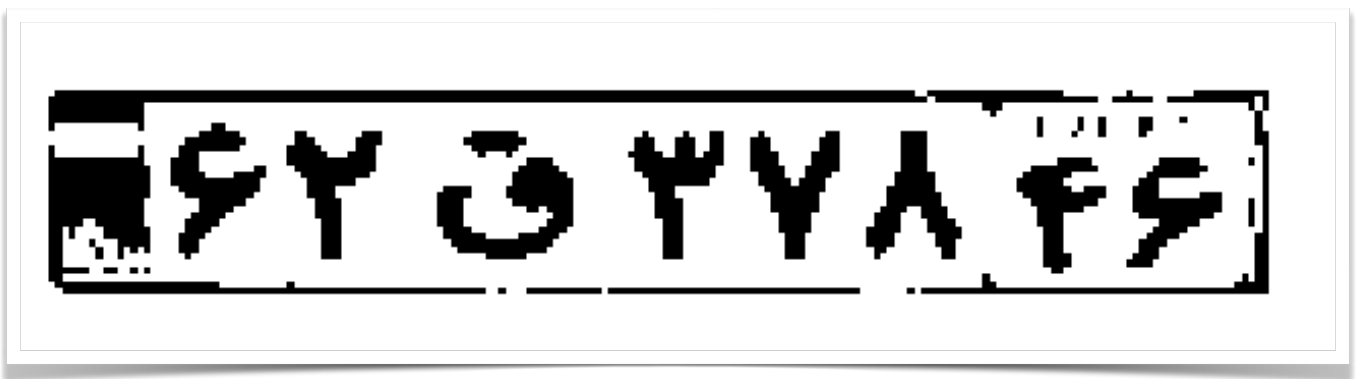
```
final_output =  
    '59844_'  
>>
```

Second Sample :











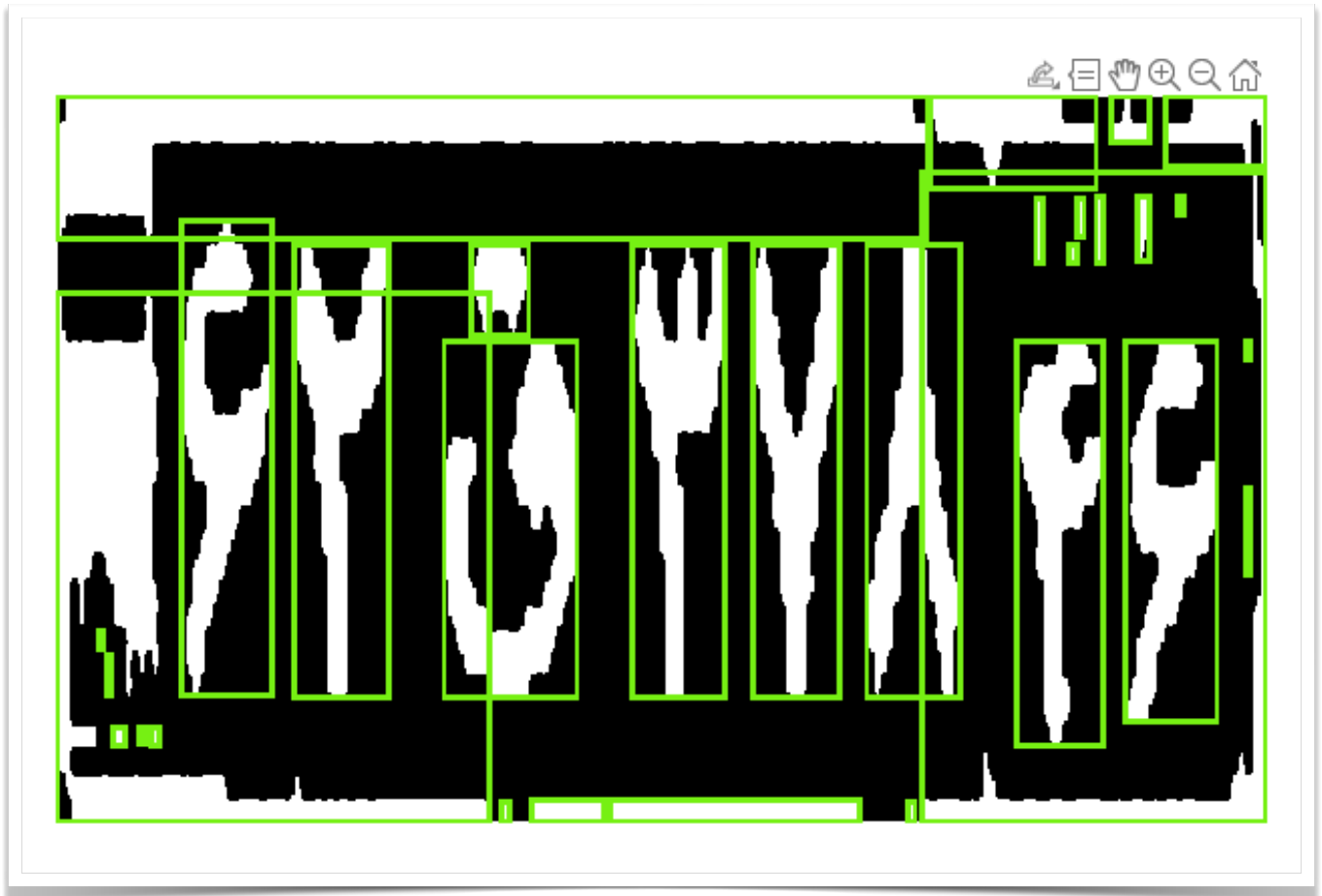




Figure 8





```
final_output =  
    '626378446ج'  
>>
```