hsvhisteq

September 12, 2019

[107]:
```
import cv2
import numpy as np
from matplotlib import pyplot as plt
# Load the image
img = cv2.imread("1.jpg")
print(img.shape)
plt.title('Original BGR image')
plt.imshow(img)
plt.xticks([]), plt.yticks([])
plt.show()
org_img=img
```

```
(1920, 1080, 3)
```

[108]:
```
background= cv2.imread("back.jpg")
print(background.shape)
plt.title('Background image')
plt.imshow(background)
plt.xticks([]), plt.yticks([])
plt.show()
```

```
(1920, 1080, 3)
```

[109]:
```
#cv2.imshow('original_img',img)
per=50
width= int( (img.shape[1]*per) / 100)
height= int ((img.shape[0]*per) /100 )
dim=(width, height)
img = cv2.resize(img,dim, interpolation = cv2.INTER_AREA)
print(img.shape)
image=img
#cv2.imshow('Resized_img',img)
# Apply log transform
#c= 255/ (np.log(1+np.max(img)))
```

```
(960, 540, 3)
```

[110]:
```
#cv2.imshow('original_img',img)
per1=50
```

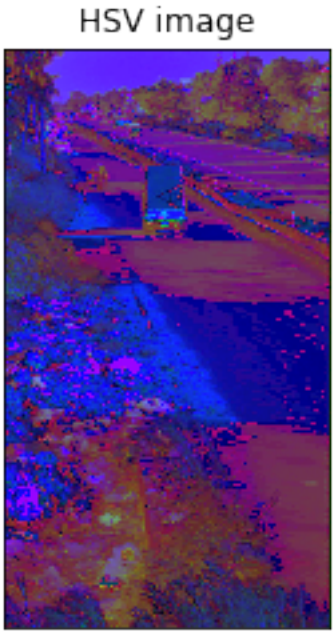
```
width1= int( (background.shape[1]*per) / 100)
height1= int ((background.shape[0]*per) /100 )
dim1=(width1, height1)
background = cv2.resize(background,dim1, interpolation = cv2.INTER_AREA)
print(background.shape)

#cv2.imshow('Resized_img',img)
# Apply log transform
#c= 255/ (np.log(1+np.max(img)))
```

```
(960, 540, 3)
```

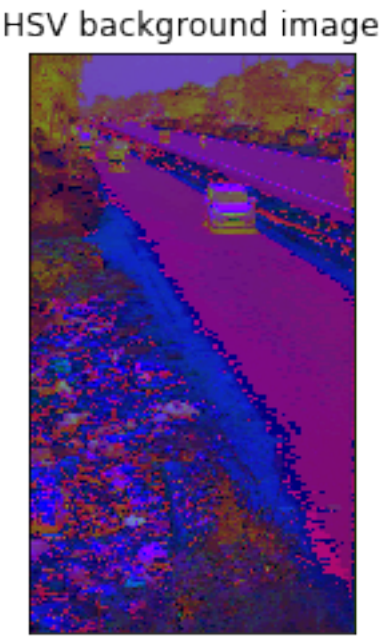
[111]:
```
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB)
plt.title('RGB image')
plt.imshow(img, cmap='gray')
plt.xticks([]), plt.yticks([])
plt.show()
image=img
```

[112]:
```
background = cv2.cvtColor(background, cv2.COLOR_BGR2RGB)
plt.title('RGB background image')
plt.imshow(background, cmap='gray')
plt.xticks([]), plt.yticks([])
plt.show()
```

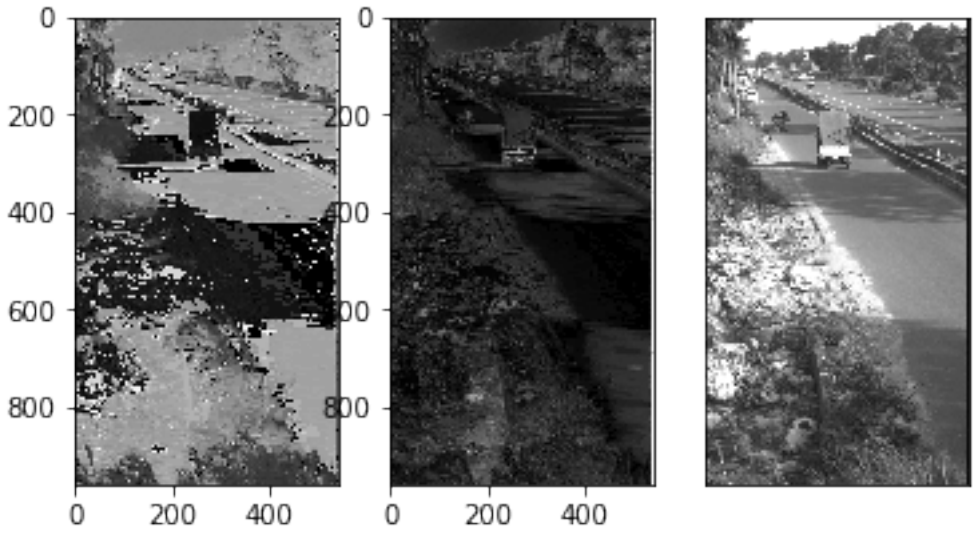
[113]:
```
img = cv2.cvtColor(img, cv2.COLOR_RGB2HSV)
plt.title('HSV image')
plt.imshow(img, cmap='gray')
plt.xticks([]), plt.yticks([])
plt.show()
```



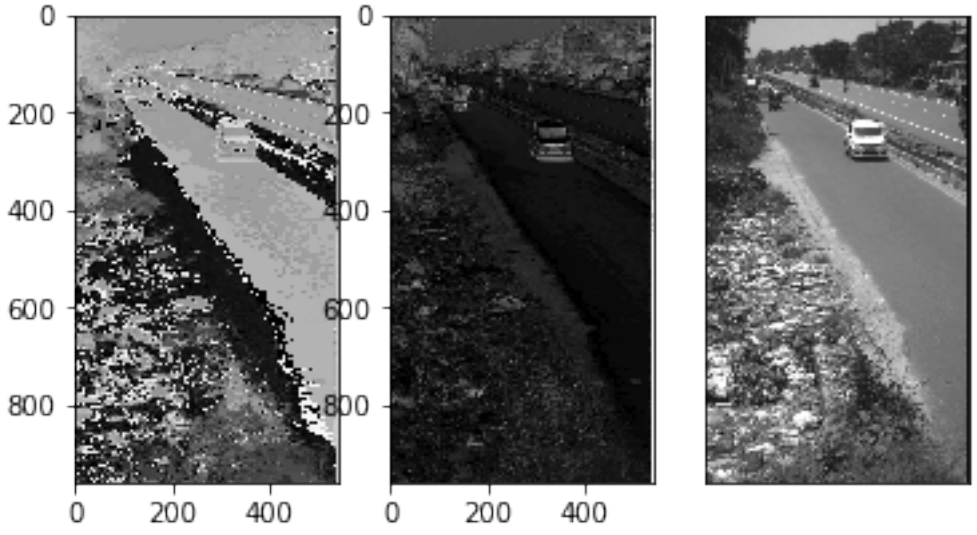
[114]:
```
background= cv2.cvtColor(background, cv2.COLOR_RGB2HSV)
plt.title('HSV background image')
plt.imshow(background, cmap='gray')
plt.xticks([]), plt.yticks([])
plt.show()
```



[115]:
```
H, S, V = cv2.split(img)
plt.title('HSV components image')
plt.subplot(131)
plt.imshow(H, cmap='gray')
plt.subplot(132)
plt.imshow(S, cmap='gray')
plt.subplot(133)
plt.imshow(V, cmap='gray')
plt.xticks([]), plt.yticks([])
plt.show()
```



[116]:
```
BH, BS, BV= cv2.split(background)
plt.title('HSV components background image')
plt.subplot(131)
plt.imshow(BH, cmap='gray')
plt.subplot(132)
plt.imshow(BS, cmap='gray')
plt.subplot(133)
plt.imshow(BV, cmap='gray')
plt.xticks([]), plt.yticks([])
plt.show()
```



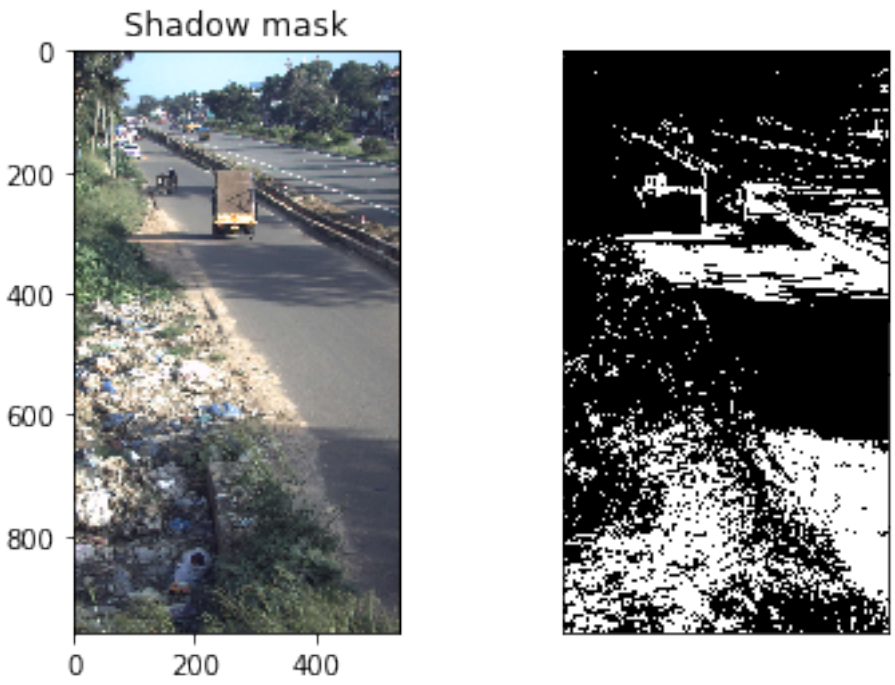
[117]:
```
alpha1 = 0.2 ; beta1 = 0.8 ; thetas= 1; thetah= 1
s=V
print(s.shape)
```

```
(960, 540)
```

[118]:
```
import sys
V = np.float32(V)/255
BV= np.float32(BV)/255
S = np.float32(S)/255
BS= np.float32(BS)/255
H = np.float32(H)/255
BH= np.float32(BH)/255
s= V.copy()
```

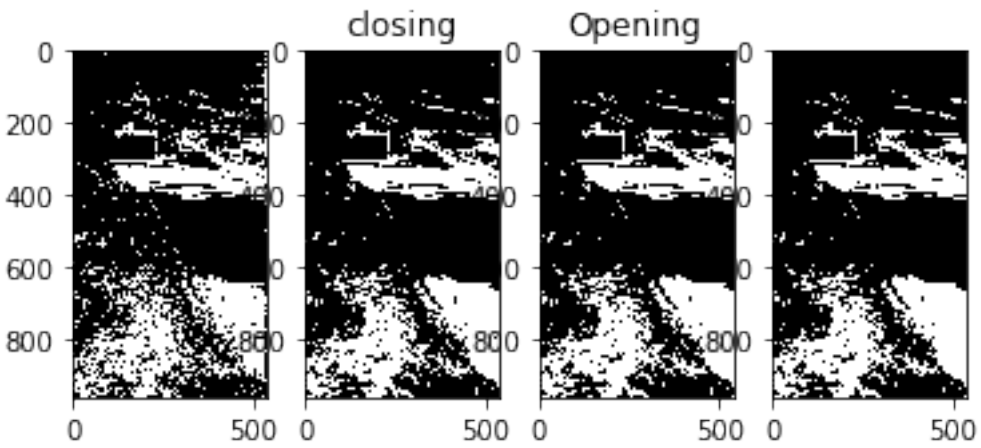
[119]:
```
for i in range(H.shape[0]):
    for j in range(H.shape[1]):
        try:
            x= V[i,j]/ BV[i,j]
        except ZeroDivisionError:
            x = 0

        if ( (alpha1 <= x <= beta1) and (abs(S[i,j]-BS[i,j])<= thetas) and␣
 ↪(abs(H[i,j]-BH[i,j])<=thetah)) :
            s[i,j]=1
        else:
            s[i,j]=0

s=s*255
```

[120]:
```
s = np.array(s,dtype=np.uint8)

plt.subplot(121)
plt.imshow(image, cmap = 'gray')
plt.title('Shadow mask')
plt.subplot(122)
plt.imshow(s, cmap= 'gray')


plt.xticks([]), plt.yticks([])
plt.show()
```



[121]:
```
plt.subplot(141)
plt.imshow(s, cmap= 'gray')

median = cv2.medianBlur(s, 5)
plt.subplot(142)
plt.imshow(median, cmap = 'gray')
# Closing - Good for removing noise
kernel = np.ones((3,3), np.uint8)
closing = cv2.morphologyEx(median, cv2.MORPH_CLOSE, kernel)
plt.title('closing')
plt.subplot(143)
plt.imshow(closing, cmap = 'gray')


# Opening - Good for removing noise
opening = cv2.morphologyEx(closing, cv2.MORPH_OPEN, kernel)
plt.title('Opening')
plt.subplot(144)
plt.imshow(opening, cmap = 'gray')
```

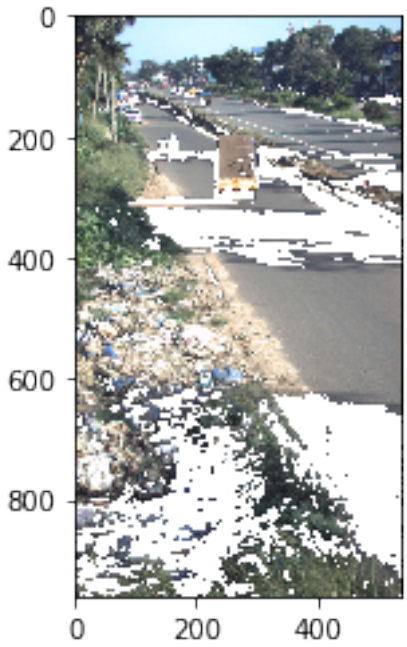
[121]:
```
<matplotlib.image.AxesImage at 0x7fdf685548d0>
```



[122]:
```
opening = np.array(opening,dtype=np.uint8)
print(opening.shape)
print(opening.dtype)
print(image.shape)
print(image.dtype)

result = image.copy()
height=result.shape[0]
width=result.shape[1]
blank_image = np.zeros((height,width,3), np.uint8)
print(blank_image.shape)
#result[opening!=0] = (0,0,0)

result[opening!=0] = (255,255,255)

plt.imshow(result, cmap = 'gray')
#plt.imshow(blank_image, cmap = 'gray')
```

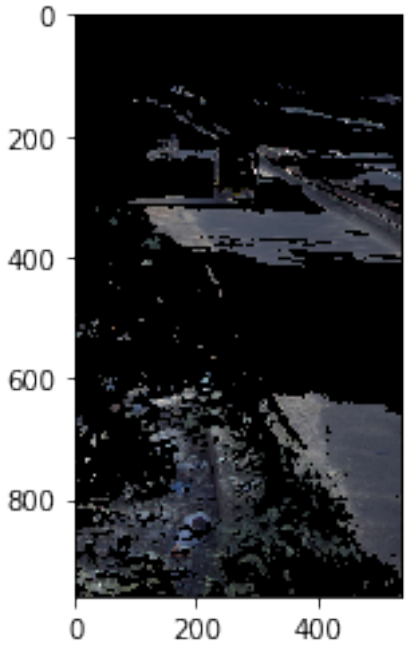
```
(960, 540)
uint8
(960, 540, 3)
uint8
(960, 540, 3)
```

[122]:
```
<matplotlib.image.AxesImage at 0x7fdf684ed208>
```



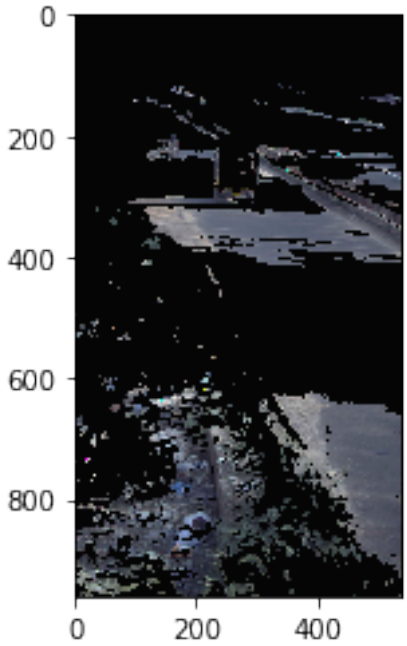
[123]:
```
blank_image[opening!=0] = image[opening!=0]
plt.imshow(blank_image, cmap = 'gray')
```

[123]:
```
<matplotlib.image.AxesImage at 0x7fdf684b92e8>
```



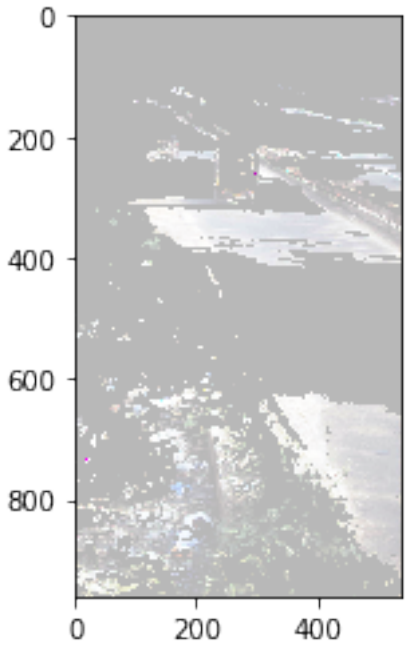
[131]:
```
blank_image1=np.uint8(np.double(blank_image)*1.2 + 5)
plt.imshow(blank_image1, cmap = 'gray')
```

[131]:
```
<matplotlib.image.AxesImage at 0x7fdf689676d8>
```



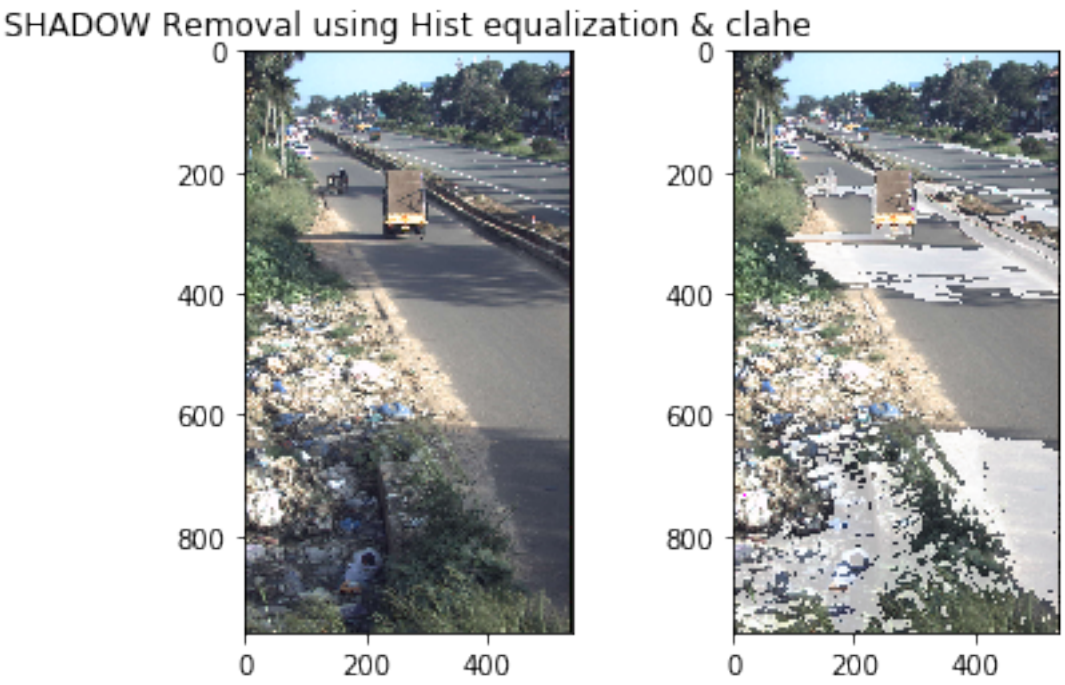
[132]:
```
equ = np.zeros((height,width,3), np.uint8)
equ[opening==0]=image[opening==0]
equ[:,:,0] = cv2.equalizeHist(blank_image1[:,:,0])
equ[:,:,1] = cv2.equalizeHist(blank_image1[:,:,1])
equ[:,:,2] = cv2.equalizeHist(blank_image1[:,:,2])
plt.imshow(equ, cmap = 'gray')
```

[132]:
```
<matplotlib.image.AxesImage at 0x7fdf689a70f0>
```



[133]:
```
equ[opening==0]=image[opening==0]
plt.subplot(121)
plt.title('SHADOW Removal using Hist equalization & clahe')
plt.imshow(image, cmap = 'gray')
plt.subplot(122)
plt.imshow(equ, cmap = 'gray')
```

[133]:
```
<matplotlib.image.AxesImage at 0x7fdf686c9ba8>
```



[134]:
```
claheimg = np.zeros((height,width,3), np.uint8)
claheimg[opening==0]=image[opening==0]
clahe = cv2.createCLAHE(clipLimit=9.0,tileGridSize=(5, 5))

claheimg[:,:,0] = clahe.apply(blank_image1[:,:,0])
claheimg[:,:,1] = clahe.apply(blank_image1[:,:,1])
claheimg[:,:,2] = clahe.apply(blank_image1[:,:,2])
```

[135]:
```
claheimg[opening==0]=image[opening==0]

plt.subplot(121)
plt.title('SHADOW Removal using clahe')
plt.imshow(image, cmap = 'gray')
plt.subplot(122)
plt.title('SHADOW Removal using CLAHE ')
plt.imshow(claheimg, cmap = 'gray')
```

[135]:
```
<matplotlib.image.AxesImage at 0x7fdf68719e48>
```

[140]:
```
import os
path1 = "/home/user/Downloads/DAY2/SET7"
listing = os.listdir(path1)
for file in listing:
    print(file)
print(len(listing))
```

```
img2597950ANPR_L00.jpg
img2597952ANPR_L00.jpg
img2597891ANPR_L00.jpg
img2597951ANPR_L00.jpg
img2597863ANPR_L00.jpg
img2597849ANPR_L00.jpg
img2597841ANPR_L00.jpg
img2597897ANPR_L00.jpg
img2597932ANPR_L00.jpg
img2597818ANPR_L00.jpg
img2597866ANPR_L00.jpg
img2597824ANPR_L00.jpg
img2597829ANPR_L00.jpg
img2597959ANPR_L00.jpg
img2597955ANPR_L00.jpg
img2597819ANPR_L00.jpg
img2597880ANPR_L00.jpg
img2597873ANPR_L00.jpg
```

img2597928ANPR_L00.jpg
img2597961ANPR_L00.jpg
img2597943ANPR_L00.jpg
img2597895ANPR_L00.jpg
img2597867ANPR_L00.jpg
img2597815ANPR_L00.jpg
img2597935ANPR_L00.jpg
img2597942ANPR_L00.jpg
img2597962ANPR_L00.jpg
img2597858ANPR_L00.jpg
img2597965ANPR_L00.jpg
img2597912ANPR_L00.jpg
img2597970ANPR_L00.jpg
img2597908ANPR_L00.jpg
img2597944ANPR_L00.jpg
img2597953ANPR_L00.jpg
img2597901ANPR_L00.jpg
img2597933ANPR_L00.jpg
img2597898ANPR_L00.jpg
img2597948ANPR_L00.jpg
img2597885ANPR_L00.jpg
img2597874ANPR_L00.jpg
img2597830ANPR_L00.jpg
img2597837ANPR_L00.jpg
img2597969ANPR_L00.jpg
img2597903ANPR_L00.jpg
img2597947ANPR_L00.jpg
img2597957ANPR_L00.jpg
img2597843ANPR_L00.jpg
img2597968ANPR_L00.jpg
img2597822ANPR_L00.jpg
img2597914ANPR_L00.jpg
img2597860ANPR_L00.jpg
img2597864ANPR_L00.jpg
img2597875ANPR_L00.jpg
img2597811ANPR_L00.jpg
img2597832ANPR_L00.jpg
img2597915ANPR_L00.jpg
img2597877ANPR_L00.jpg
img2597958ANPR_L00.jpg
img2597911ANPR_L00.jpg
img2597920ANPR_L00.jpg
img2597896ANPR_L00.jpg
img2597910ANPR_L00.jpg
img2597900ANPR_L00.jpg
img2597937ANPR_L00.jpg
img2597876ANPR_L00.jpg
img2597834ANPR_L00.jpg

img2597833ANPR_L00.jpg
img2597846ANPR_L00.jpg
img2597938ANPR_L00.jpg
img2597842ANPR_L00.jpg
img2597892ANPR_L00.jpg
img2597904ANPR_L00.jpg
img2597941ANPR_L00.jpg
img2597831ANPR_L00.jpg
img2597816ANPR_L00.jpg
img2597936ANPR_L00.jpg
img2597848ANPR_L00.jpg
img2597881ANPR_L00.jpg
img2597851ANPR_L00.jpg
img2597939ANPR_L00.jpg
img2597918ANPR_L00.jpg
img2597828ANPR_L00.jpg
img2597967ANPR_L00.jpg
img2597882ANPR_L00.jpg
img2597887ANPR_L00.jpg
img2597865ANPR_L00.jpg
img2597922ANPR_L00.jpg
img2597844ANPR_L00.jpg
img2597886ANPR_L00.jpg
img2597810ANPR_L00.jpg
img2597893ANPR_L00.jpg
img2597827ANPR_L00.jpg
img2597813ANPR_L00.jpg
img2597840ANPR_L00.jpg
img2597890ANPR_L00.jpg
img2597859ANPR_L00.jpg
img2597899ANPR_L00.jpg
img2597869ANPR_L00.jpg
img2597820ANPR_L00.jpg
img2597930ANPR_L00.jpg
img2597963ANPR_L00.jpg
img2597857ANPR_L00.jpg
img2597894ANPR_L00.jpg
img2597946ANPR_L00.jpg
img2597923ANPR_L00.jpg
img2597883ANPR_L00.jpg
img2597817ANPR_L00.jpg
img2597855ANPR_L00.jpg
img2597907ANPR_L00.jpg
img2597839ANPR_L00.jpg
img2597862ANPR_L00.jpg
img2597847ANPR_L00.jpg
img2597879ANPR_L00.jpg
img2597905ANPR_L00.jpg

img2597971ANPR_L00.jpg
img2597871ANPR_L00.jpg
img2597940ANPR_L00.jpg
img2597919ANPR_L00.jpg
img2597956ANPR_L00.jpg
img2597925ANPR_L00.jpg
img2597954ANPR_L00.jpg
img2597927ANPR_L00.jpg
img2597960ANPR_L00.jpg
img2597916ANPR_L00.jpg
img2597826ANPR_L00.jpg
img2597838ANPR_L00.jpg
img2597835ANPR_L00.jpg
img2597823ANPR_L00.jpg
img2597966ANPR_L00.jpg
img2597825ANPR_L00.jpg
img2597854ANPR_L00.jpg
img2597836ANPR_L00.jpg
img2597852ANPR_L00.jpg
img2597845ANPR_L00.jpg
img2597878ANPR_L00.jpg
img2597909ANPR_L00.jpg
img2597850ANPR_L00.jpg
img2597821ANPR_L00.jpg
img2597921ANPR_L00.jpg
img2597926ANPR_L00.jpg
img2597949ANPR_L00.jpg
img2597945ANPR_L00.jpg
img2597870ANPR_L00.jpg
img2597924ANPR_L00.jpg
img2597889ANPR_L00.jpg
img2597814ANPR_L00.jpg
img2597884ANPR_L00.jpg
img2597888ANPR_L00.jpg
img2597964ANPR_L00.jpg
img2597934ANPR_L00.jpg
img2597856ANPR_L00.jpg
img2597853ANPR_L00.jpg
img2597861ANPR_L00.jpg
img2597902ANPR_L00.jpg
img2597917ANPR_L00.jpg
img2597931ANPR_L00.jpg
img2597929ANPR_L00.jpg
img2597906ANPR_L00.jpg
img2597872ANPR_L00.jpg
img2597868ANPR_L00.jpg
img2597913ANPR_L00.jpg
img2597812ANPR_L00.jpg

```
162
```

[ ]: