

```

1 from PIL import Image
2 import numpy as np
3 import matplotlib.pyplot as plt
4
5 im1 = Image.open("Original.jpg").resize((255,255))
6 im2 = Image.open("alam2.jpg").resize((255,255))
7 arr1 = np.array(im1.copy())
8 arr2 = np.array(im2.copy())
9 arr3 = np.array(im2.copy())
10
11 keys = np.arange(256)
12 hist1R = np.zeros((256), dtype= np.uint32)
13 hist1G = np.zeros((256), dtype= np.uint32)
14 hist1B = np.zeros((256), dtype= np.uint32)
15
16 hist2R = np.zeros((256), dtype= np.uint32)
17 hist2G = np.zeros((256), dtype= np.uint32)
18 hist2B = np.zeros((256), dtype= np.uint32)
19
20 hist3R = np.zeros((256), dtype= np.uint32)
21 hist3G = np.zeros((256), dtype= np.uint32)
22 hist3B = np.zeros((256), dtype= np.uint32)
23
24 for y in range(arr1.shape[0]):
25     for x in range(arr1.shape[1]):
26         hist1R[arr1[y,x,0]] = hist1R[arr1[y,x,0]]+1;
27         hist1G[arr1[y,x,1]] = hist1G[arr1[y,x,1]]+1;
28         hist1B[arr1[y,x,2]] = hist1B[arr1[y,x,2]]+1;
29
30         hist2R[arr2[y,x,0]] = hist2R[arr2[y,x,0]]+1;
31         hist2G[arr2[y,x,1]] = hist2G[arr2[y,x,1]]+1;
32         hist2B[arr2[y,x,2]] = hist2B[arr2[y,x,2]]+1;
33
34 cdf1R = hist1R.cumsum()
35 cdf1R_norm = cdf1R * float(hist1R.max()) / cdf1R.max()
36
37 cdf1G = hist1G.cumsum()
38 cdf1G_norm = cdf1G * float(hist1G.max()) / cdf1G.max()
39
40 cdf1B = hist1B.cumsum()
41 cdf1B_norm = cdf1B * float(hist1B.max()) / cdf1B.max()
42
43
44 cdf2R = hist2R.cumsum()
45 cdf2R_norm = cdf2R * float(hist2R.max()) / cdf2R.max()
46
47 cdf2G = hist2G.cumsum()
48 cdf2G_norm = cdf2G * float(hist2G.max()) / cdf2G.max()
49
50 cdf2B = hist2B.cumsum()
51 cdf2B_norm = cdf2B * float(hist2B.max()) / cdf2B.max()
52
53 lutR = np.zeros((256), dtype=np.uint32)
54 lutG = np.zeros((256), dtype=np.uint32)
55 lutB = np.zeros((256), dtype=np.uint32)
56
57 gjR = 0
58 gjG = 0
59 gjB = 0
60
61 for giR in range(255):
62     while(gjR<255 and cdf2R[gjR+1] < cdf1R[giR+1]):
63         gjR = gjR+1
64     lutR[giR+1] = gjR
65
66 for giG in range(255):
67     while(gjG<255 and cdf2G[gjG+1] < cdf1G[giG+1]):
68         gjG = gjG+1
69     lutG[giG+1] = gjG
70
71 for giB in range(255):
72     while(gjB<255 and cdf2B[gjB+1] < cdf1B[giB+1]):
73         gjB = gjB+1
74     lutB[giB+1] = gjB
75
76 for y in range(arr1.shape[0]):
77     for x in range(arr1.shape[1]):
78         tmpR = lutR[arr1[y,x,0]]
79         tmpG = lutG[arr1[y,x,1]]

```

```

80     tmpB = lutB[arr1[y,x,2]]
81     arr3[y,x] = [tmpR,tmpG,tmpB]
82     hist3R[tmpR] = hist3R[tmpR]+1
83     hist3G[tmpG] = hist3G[tmpG]+1
84     hist3B[tmpB] = hist3B[tmpB]+1
85
86 cdf3R = hist3R.cumsum()
87 cdf3R_norm = cdf3R * float(hist3R.max()) / cdf3R.max()
88
89 cdf3G = hist3G.cumsum()
90 cdf3G_norm = cdf3G * float(hist3G.max()) / cdf3G.max()
91
92 cdf3B = hist3B.cumsum()
93 cdf3B_norm = cdf3B * float(hist3B.max()) / cdf3B.max()
94
95 fig = plt.figure(1)
96 plt.bar(keys,hist1R)
97 plt.bar(keys,hist1G)
98 plt.bar(keys,hist1B)
99 plt.plot(cdf1R_norm, color = 'r')
100 plt.plot(cdf1G_norm, color = 'g')
101 plt.plot(cdf1B_norm, color = 'b')
102 fig.canvas.draw()
103 data1 = np.frombuffer(fig.canvas.tostring_rgb(), dtype=np.uint8)
104 data1 = data1.reshape(fig.canvas.get_width_height()[::-1] + (3,))
105 histImage1 = Image.fromarray(data1).resize((255,255))
106 histNp1 = np.array(histImage1)
107
108 fig = plt.figure(2)
109 plt.bar(keys,hist2R)
110 plt.bar(keys,hist2G)
111 plt.bar(keys,hist2B)
112 plt.plot(cdf2R_norm, color = 'r')
113 plt.plot(cdf2G_norm, color = 'g')
114 plt.plot(cdf2B_norm, color = 'b')
115 fig.canvas.draw()
116 data2 = np.frombuffer(fig.canvas.tostring_rgb(), dtype=np.uint8)
117 data2 = data2.reshape(fig.canvas.get_width_height()[::-1] + (3,))
118 histImage2 = Image.fromarray(data2).resize((255,255))
119 histNp2 = np.array(histImage2)
120
121 fig = plt.figure(3)
122 plt.bar(keys,hist3R)
123 plt.bar(keys,hist3G)
124 plt.bar(keys,hist3B)
125 plt.plot(cdf3R_norm, color = 'r')
126 plt.plot(cdf3G_norm, color = 'g')
127 plt.plot(cdf3B_norm, color = 'b')
128 fig.canvas.draw()
129 data3 = np.frombuffer(fig.canvas.tostring_rgb(), dtype=np.uint8)
130 data3 = data3.reshape(fig.canvas.get_width_height()[::-1] + (3,))
131 histImage3 = Image.fromarray(data3).resize((255,255))
132 histNp3 = np.array(histImage3)
133
134 Image.fromarray(np.hstack((np.vstack((histNp1,arr1)),np.vstack((histNp2,arr2)),np.vstack((histNp3,arr3))))).show()

```