**直方圖均衡化**

直方圖均衡化的3種清況:

1.灰度圖像長條圖均衡化  
2.彩色圖像長條圖均衡化  
3.YUV長條圖均衡化

(RGB 色彩值： R = 0~255、G = 0~255、B = 0~255)  
(YUV 色階值： Y = 16~235、U = 16~235、V = 16~235)

1. **灰度圖像長條圖均衡化**

**=>**對長條圖均衡化主要使用opencv提供的一個equalizeHist()方法，將灰度圖像作為參數傳進equalizeHist()方法即可。

|  |
| --- |
| import cv2  import numpy as np  img = cv2.imread("lena.jpg", 1)  # 轉為灰階圖片  gray = cv2.cvtColor(img, cv2.COLOR\_BGR2GRAY)  cv2.imshow("src", gray)  # 灰階圖片直方圖均衡  dst = cv2.equalizeHist(gray)  cv2.imshow("dst", dst)  cv2.waitKey(0) |



1. **彩色圖像長條圖均衡化**

**=>**彩色圖像的長條圖均衡化和灰度圖像略有不同,需要將彩色圖像先用split()方法,將三個通道拆分,然後分別進行均衡化.最後使用merge()方法將均衡化之後的三個通道進行合併。

|  |
| --- |
| import cv2  import numpy as np  img = cv2.imread("lena.jpg", 1)  cv2.imshow("src", img)  # 彩色圖像均衡化,需要分解通道 對每一個通道均衡化  (b, g, r) = cv2.split(img)  bH = cv2.equalizeHist(b)  gH = cv2.equalizeHist(g)  rH = cv2.equalizeHist(r)  # 合併每一個通道  result = cv2.merge((bH, gH, rH))  cv2.imshow("dst", result)  cv2.waitKey(0) |

1. **YUV長條圖均衡化**

|  |
| --- |
| import cv2  import numpy as np  img = cv2.imread("image0.jpg", 1)  imgYUV = cv2.cvtColor(img, cv2.COLOR\_BGR2YCrCb)  cv2.imshow("src", img)  channelsYUV = cv2.split(imgYUV)  channelsYUV[0] = cv2.equalizeHist(channelsYUV[0])  channels = cv2.merge(channelsYUV)  result = cv2.cvtColor(channels, cv2.COLOR\_YCrCb2BGR)  cv2.imshow("dst", result)  cv2.waitKey(0) |