import cv2 as cv

from matplotlib import pyplot as plt

img = cv.imread('gradient.jpg',0)



ret,thresh1 = cv.threshold(img,127,255,cv.THRESH\_BINARY)

ret,thresh2 = cv.threshold(img,127,255,cv.THRESH\_BINARY\_INV)

ret,thresh3 = cv.threshold(img,127,255,cv.THRESH\_TRUNC)

ret,thresh4 = cv.threshold(img,127,255,cv.THRESH\_TOZERO)

ret,thresh5 = cv.threshold(img,127,255,cv.THRESH\_TOZERO\_INV)

titles = ['Original Image','BINARY','BINARY\_INV','TRUNC','TOZERO','TOZERO\_INV']

images = [img, thresh1, thresh2, thresh3, thresh4, thresh5]

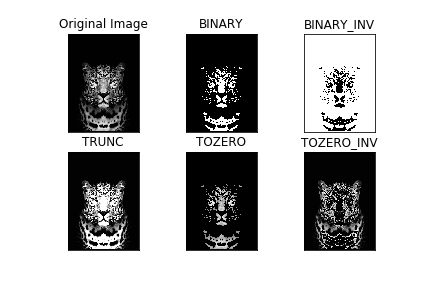
for i in range(6):

plt.subplot(2,3,i+1),plt.imshow(images[i],'gray')

plt.title(titles[i])

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

plt.show()



img2 = cv.imread('gradient.jpg',0)

img2 = cv.medianBlur(img2,5)

ret,th1 = cv.threshold(img2,127,255,cv.THRESH\_BINARY)

th2 = cv.adaptiveThreshold(img2,255,cv.ADAPTIVE\_THRESH\_MEAN\_C,\

cv.THRESH\_BINARY,11,2)

th3 = cv.adaptiveThreshold(img2,255,cv.ADAPTIVE\_THRESH\_GAUSSIAN\_C,\

cv.THRESH\_BINARY,11,2)

titles = ['Original Image', 'Global Thresholding (v = 127)',

'Adaptive Mean Thresholding', 'Adaptive Gaussian Thresholding']

images = [img2, th1, th2, th3]

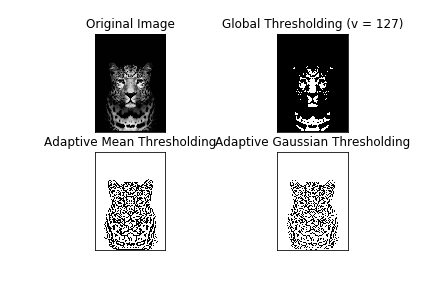
for i in range(4):

plt.subplot(2,2,i+1),plt.imshow(images[i],'gray')

plt.title(titles[i])

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

plt.show()



img3 = cv.imread('gradient.jpg',0)

# global thresholding

ret1,th1 = cv.threshold(img3,127,255,cv.THRESH\_BINARY)

# Otsu's thresholding

ret2,th2 = cv.threshold(img3,0,255,cv.THRESH\_BINARY+cv.THRESH\_OTSU)

# Otsu's thresholding after Gaussian filtering

blur = cv.GaussianBlur(img3,(5,5),0)

ret3,th3 = cv.threshold(blur,0,255,cv.THRESH\_BINARY+cv.THRESH\_OTSU)

# plot all the images and their histograms

images = [img, 0, th1,

img, 0, th2,

blur, 0, th3]

titles = ['Original Noisy Image','Histogram','Global Thresholding (v=127)',

'Original Noisy Image','Histogram',"Otsu's Thresholding",

'Gaussian filtered Image','Histogram',"Otsu's Thresholding"]

for i in range(3):

plt.subplot(3,3,i\*3+1),plt.imshow(images[i\*3],'gray')

plt.title(titles[i\*3]), plt.xticks([]), plt.yticks([])

plt.subplot(3,3,i\*3+2),plt.hist(images[i\*3].ravel(),256)

plt.title(titles[i\*3+1]), plt.xticks([]), plt.yticks([])

plt.subplot(3,3,i\*3+3),plt.imshow(images[i\*3+2],'gray')

plt.title(titles[i\*3+2]), plt.xticks([]), plt.yticks([])

plt.show()

