def mean\_filter\_rgb(img,k\_sizes):

height,width,deep=img.shape

output=np.zeros\_like(img)

radius=k\_sizes//2

for i in range(radius,height-radius):

for j in range(radius,width-radius):

for c in range(deep):

average=np.mean(img[i-radius:i+radius+1,j-radius:j+radius+1,c])

output[i,j,c]=average

return output

def Sobel\_N(img,choose):

height,width=img.shape

#output=np.zeros\_like(img) 错误原因

output=np.zeros((height,width))

k\_x=np.array(

[[-1,0,1],

[-2,0,2],

[-1,0,1]])

k\_y=np.array(

[[1,2,1],

[0,0,0],

[-1,-2,-1]])

img\_add=np.pad(img,1,'constant')

img\_add

for i in range(2,height+1):

for j in range(2,width+1):

window=img\_add[i-1:i+2,j-1:j+2]

if choose==0:

output[i-1,j-1]=np.sum(window\*k\_x)

elif choose==1:

output[i-1,j-1]=np.sum(window\*k\_y)

#print(window)

#print(output[i-1,j-1])

#print(np.sum(window\*k\_x))

#print(output)

#return 0

return output