|  |  |
| --- | --- |
| 교육 제목 | Gaussian, Bilateral Filter |
| 교육 일시 | 2021. 11. 12 |
| 교육 장소 | 영우글로벌러닝 |
| **교육 내용** | |
| 오전 | import numpy as np  import sys  import cv2  src=cv2.imread('fig/Hawkes.jpg', cv2.IMREAD\_GRAYSCALE)  cv2.imshow('src', src)  dst\_equal=cv2.equalizeHist(src)  dst\_normal=cv2.normalize(src, None, 0, 255, cv2.NORM\_MINMAX, -1)  cv2.imshow('dst\_equal', dst\_equal)  cv2.imshow('dst\_normal', dst\_normal)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/field.bmp', cv2.IMREAD\_COLOR)  src\_ycrcb=cv2.cvtColor(src, cv2.COLOR\_BGR2YCrCb)  y, cr, cb= cv2.split(src\_ycrcb)  y\_norm=cv2.normalize(y, None, 0, 255, cv2.NORM\_MINMAX, -1)  y\_equal=cv2.equalizeHist(y)  dst\_norm=cv2.merge([y\_norm, cr, cb])  dst\_equal=cv2.merge([y\_equal, cr, cb])  dst\_norm=cv2.cvtColor(dst\_norm, cv2.COLOR\_YCrCb2BGR)  dst\_equal=cv2.cvtColor(dst\_equal, cv2.COLOR\_YCrCb2BGR)  cv2.imshow('src', src)  cv2.imshow('dst\_norm', dst\_norm)  cv2.imshow('dst\_equal', dst\_equal)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/candies2.png')  src\_hsv=cv2.cvtColor(src, cv2.COLOR\_BGR2HSV)    cv2.imshow('src', src)  dst1=cv2.inRange(src, (0, 128, 0), (100, 255, 100))  dst2=cv2.inRange(src\_hsv, (50, 170, 0), (80, 255, 255))  cv2.imshow('dst1', dst1)  cv2.imshow('dst2', dst2)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/cropland.png')  x, y, w, h=cv2.selectROI(src)  src\_ycrcb=cv2.cvtColor(src, cv2.COLOR\_BGR2YCrCb)  crop=src\_ycrcb[y:y+h, x:x+w]  hist=cv2.calcHist([crop], [1,2], None, [128, 128], [0, 256, 0, 256])  backproj=cv2.calcBackProject([src\_ycrcb], [1,2], hist, [0, 256, 0, 256], 1)  dst=cv2.copyTo(src, backproj)  cv2.imshow('src', src)  cv2.imshow('backproj', backproj)  cv2.imshow('dst', dst)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  kernel\_3=np.ones((3,3), np.float32)/9  kernel\_5=np.ones((5,5), np.float32)/25  src\_mean\_filter3=cv2.filter2D(src, -1, kernel\_3) #, (-1, -1), 0)  src\_mean\_filter5=cv2.filter2D(src, -1, kernel\_5)  cv2.imshow('src', src)  cv2.imshow('src\_mean\_filter3', src\_mean\_filter3)  cv2.imshow('src\_mean\_filter5', src\_mean\_filter5)  cv2.waitKey()  cv2.destroyAllWindows() |
| 오후 | import numpy as np  import sys  import cv2  src=cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  # GaussianBlur(src, ksize, sigmaX[, dst[, sigmaY[, borderType]]]) -> dst  # src: 입력영상  # ksize: mean filter kernel size, (0, 0) 자동으로 결정  # sigmaX: gaussian x 방향의 sigma  # sigmaY: gaussian y 방향의 sigma  dst\_gaussian1=cv2.GaussianBlur(src, (0,0), 1)  dst\_gaussian2=cv2.GaussianBlur(src, (0,0), 2)  dst\_mean=cv2.blur(src, (7,7))  cv2.imshow('src', src)  cv2.imshow('dst\_mean', dst\_mean)  cv2.imshow('dst\_gaussian1', dst\_gaussian1)  cv2.imshow('dst\_gaussian2', dst\_gaussian2)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  src\_gblur=cv2.GaussianBlur(src, (0, 0), 1)  dst\_sharp=cv2.addWeighted(src, 2, src\_gblur, -1, 0)  cv2.imshow('src', src)  cv2.imshow('src\_gblur', src\_gblur)  cv2.imshow('dst\_sharp', dst\_sharp)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  print(src.shape)  salt\_pepper\_1=np.random.choice((0, 255), src.shape, p=(0.95, 0.05)).astype(np.uint32)  salt\_pepper\_2=np.random.choice((0, 255), src.shape, p=(0.95, 0.05)).astype(np.uint32)  src\_noise=src+salt\_pepper\_1 #-salt\_pepper\_2  src\_noise=np.clip(src\_noise, 0, 255).astype(np.uint8)  cv2.imshow('src', src)  # cv2.imshow('salt\_pepper\_1', salt\_pepper\_1)  cv2.imshow('src\_noise', src\_noise)  cv2.waitKey()  cv2.destroyAllWindows()  src=cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  dst\_gaussian=cv2.GaussianBlur(src, (0, 0), 1)  dst\_bilateral=cv2.bilateralFilter(src, -1, 10, 3)  cv2.imshow('src', src)  cv2.imshow('dst\_gaussian', dst\_gaussian)  cv2.imshow('dst\_bilateral', dst\_bilateral)  cv2.waitKey()  cv2.destroyAllWindows() |