|  |  |
| --- | --- |
| 교육제목 | 데이터 기반 인공지능 시스템 엔지니어 양성 과정 |
| 교육일시 | 211112 |
| 교육장소 | YGL 학과장 및 자택(디스코드 사용 온라인 학습) |
| **교육내용** | |
| 1. Mean filter   import numpy as np  import sys  import cv2  src = cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  kernel\_3 = np.ones((3,3), dtype = np.float32)/9. # 정수로해도되는데 누적되면 곤란. 플롯으로가자  kernel\_5 = np.ones((5,5), dtype = np.float32)/25.  # kernel\_5 = np.ones((11,11), dtype = np.float32)/121. # 이미지 뭉개짐  # mean filter 만들기 -1은 똑같은 형식으로..  dst = cv2.filter2D(src, -1, kernel\_5)  # 한방에 해뿌리기. 커널지정안해도 그냥 크기만 적어도 된다  dst\_blur = cv2.blur(src, (5,5))  cv2.imshow('src', src)  # cv2.imshow('dst', dst)  cv2.imshow('dst\_blur', dst\_blur)  cv2.waitKey()  cv2.destroyAllWindows()   1. 가우시안 필터   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) # 1 주면 x,y 시그마 전부 1로드감  dst\_gaussian2 = cv2.GaussianBlur(src, (0, 0), 2) # 시그마2 좀더 블러됨  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()   1. 샤프닝 필터: 2를 곱해서 가우시안을 뺌   rc = cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  src\_gblur = cv2.GaussianBlur(src, (0,0), 2) # 시그마 = 2  dst\_sharp = cv2.addWeighted(src, 2, src\_gblur, -1, 0)  cv2.imshow('src', src)  cv2.imshow('src\_gblur', src\_gblur) # 블러  cv2.imshow('dst\_sharp', dst\_sharp) # 샤프닝(노이즈심해지고 더 반짝반짝해짐)  cv2.waitKey()  cv2.destroyAllWindows()   1. 양방향 필터(bilateral filter)   src = cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  dst\_gaussian = cv2.GaussianBlur(src, (0,0), 3.0) # 시그마가 5나되네... 블러빡셈  dst\_bilateral = cv2.bilateralFilter(src, -1, 10, 3) # 동일한 시그마인데 차이 확연  cv2.imshow('src', src)  cv2.imshow('dst\_gaussian', dst\_gaussian)  cv2.imshow('dst\_bilateral', dst\_bilateral)  cv2.waitKey()  cv2.destroyAllWindows()   1. 잡음제거필터 : 잘 안쓴다.   src = cv2.imread('fig/blue\_eyes.png', cv2.IMREAD\_GRAYSCALE)  src.shape # 790x1200  salt\_pepper\_1 = np.random.choice((0,255), src.shape, p = (0.95, 0.05)).astype(np.int32)  salt\_pepper\_2 = np.random.choice((0,255), src.shape, p = (0.95, 0.05)).astype(np.int32)  src\_noise = src - salt\_pepper\_1 + salt\_pepper\_2  src\_noise = np.clip(src\_noise, 0, 255).astype(np.uint8)  # cv2.imshow('salt\_pepper\_1', salt\_pepper\_1)  cv2.imshow('src\_noice', src\_noise)  cv2.waitKey()  cv2.destroyAllWindows() | |