|  |  |
| --- | --- |
| 교육 제목 | Image processing |
| 교육 일시 | 2021. 11. 9 |
| 교육 장소 | 영우글로벌러닝 |
| **교육 내용** | |
| 오전 | import numpy as np  import sys  import cv2  img=np.full((600, 1200, 3), 255, np.uint8)  # img=cv2.imread('fig/puppy.bmp', cv2.IMREAD\_COLOR)  cv2.line(img, (50, 50), (300, 50), (0, 0, 255), 5)  cv2.rectangle(img, (50, 70), (300, 100), (255, 0, 0), -1)  cv2.circle(img, (500, 250), 200, (0, 255, 0), 8, cv2.LINE\_AA)  text='Merry Christmas'  cv2.putText(img, text, (600, 400), cv2.FONT\_HERSHEY\_COMPLEX, 2, (0, 0, 255), 2, cv2.LINE\_AA)  cv2.namedWindow('img')  cv2.imshow('img', img)  cv2.waitKey()  cv2.destroyAllWindows()  cap=cv2.VideoCapture(0)  if not cap.isOpened():  print('Video open failed')  cap.release()  sys.exit()    width=int(cap.get(cv2.CAP\_PROP\_FRAME\_WIDTH))  height=int(cap.get(cv2.CAP\_PROP\_FRAME\_HEIGHT))  fps=int(cap.get(cv2.CAP\_PROP\_FPS))  fourcc=cv2.VideoWriter\_fourcc(\*'DIVX')    out=cv2.VideoWriter('edge.avi', fourcc, fps, (width, height))  cv2.namedWindow('webcam', cv2.WINDOW\_AUTOSIZE)  while True:  ret, frame=cap.read()    if not ret:  print('Video read failed')  break    edge=cv2.Canny(frame, 50, 150)  edge=cv2.cvtColor(edge, cv2.COLOR\_GRAY2BGR)  frame=cv2.cvtColor(frame, cv2.COLOR\_BGR2GRAY)    out.write(edge)  cv2.imshow('webcam', frame)  cv2.imshow('edge', edge)    if cv2.waitKey(10)==27:  break    cap.release()  out.release()  cv2.destroyAllWindows()  img=cv2.imread('fig/puppy.bmp', cv2.IMREAD\_GRAYSCALE)  if img is None:  print('Image load failed!')  sys.exit()    # cv2.namedWindow('puppy')  cv2.imshow('img', img)    while True:  key=cv2.waitKey()  if key==27 or key==ord('q'):  break  # elif cv2.waitKey()==ord('q'):  # break    elif key==ord('i'):  img=cv2.bitwise\_not(img)  cv2.imshow('img', img)    cv2.destroyAllWindows()  # call\_mouse(event, x, y, flags, param) -> None  # event: 마우스 이벤트 종류 e.g., cv2.EVENT\_LBUTTONDOWN  # x, y : 창을 기준으로 이벤트 발생좌표  # flags: 이벤트시 발생 상태 e.g., "ctrl"  # param: cv2.setMouseCallback()함수에서 설정한 데이터  def call\_mouse(event, x, y, flags, param):  global oldx, oldy  if event==cv2.EVENT\_LBUTTONDOWN:  # print('EVENT\_LBUTTONDOWN : ', x, y)  oldx, oldy=x,y  # elif event==cv2.EVENT\_LBUTTONUP:  # print('EVENT\_LBUTTONUP : ', x, y)  elif event==cv2.EVENT\_MOUSEMOVE:  if flags==cv2.EVENT\_FLAG\_LBUTTON:  cv2.line(img, (oldx, oldy), (x, y), (0, 0, 255), 6, cv2.LINE\_AA)  cv2.imshow('img', img)  oldx, oldy=x, y      img=np.ones((400, 600, 3), dtype=np.uint8)\*255  cv2.namedWindow('img')  # cv2.setMouseCallback(windowName, onMouse, param = None) -> None  # windowName: 마우스이벤트를 수행할 창 이름  # onMouse: 마우스 이벤트 콜벡함수  # param: 콜백함수에 전달할 데이터  cv2.setMouseCallback('img', call\_mouse, img)  cv2.imshow('img', img)  cv2.waitKey()  cv2.destroyAllWindows() |
| 오후 | def call\_track(pos):  img[:]=pos  cv2.imshow('img', img)    img=np.zeros((400, 600, 3), dtype=np.uint8)  cv2.namedWindow('img')  # createTrackbar(trackbarName, windowName, value, count, onChange) -> None  # trackbarName: 트랙바 이름  # windowName : 트랙바를 생성할 창 이름  # value : 트랙바 위치 초기값  # count : 트랙바 최댓값, 최솟값은 0  # onChange :callback 함수 e.g., onChange(pos) 위치를 정수형태로 전달  cv2.createTrackbar('level', 'img', 0, 255, call\_track)  cv2.imshow('img', img)  cv2.waitKey()  cv2.destroyAllWindows()  def call\_track(pos):  img\_glass=img\*pos  # img\_glass[:]=255-pos  cv2.imshow('img', img\_glass)    img\_alpha=cv2.imread('fig/imgbin\_sunglasses\_1.png', cv2.IMREAD\_UNCHANGED)  img=img\_alpha[:, :, -1]  img[img>0]=1  cv2.namedWindow('img')  cv2.createTrackbar('level', 'img', 0, 255, call\_track)  cv2.imshow('img', img)  cv2.waitKey()  cv2.destroyAllWindows()  -----------------------------------------------------------------------------------------  import numpy as np  import sys  import cv2  src=cv2.imread('fig/lenna.bmp', cv2.IMREAD\_COLOR)  if img is None:  print('Image load failed!')  sys.exit()  # dst=src+100  dst=cv2.add(src, (0, 0, 100, 0))  # dst=np.clip(src+100., 0, 255).astype(np.uint8)  cv2.imshow('img', src)  cv2.imshow('dst', dst)    cv2.waitKey()  cv2.destroyAllWindows() |