

시험대비- 4장 연습문제

201935282 송우석 4장 연습문제

1. 콜백 함수란 무엇인가?

콜백 함수는 매개 변수를 통해 다른 함수를 전달받고, 이벤트가 발생할 때 매개 변수에 전달된 함수를 호출하는 역할을 한다.

즉, 특정한 이벤트가 발생하면 다른 함수를 실행하는 함수이다.

예를 들어 마우스 콜백 함수는 윈도우에 마우스 이벤트가 발생했을 때, 특정한 함수에 이벤트를 전달해 실행한다.

2. cv2.WINODW_NORMAL과 cv2.WINDOW_AUTOSIZE 차이

1. `cv2.WINDOW_NORMAL` :

- 이 상수는 생성된 창의 크기를 사용자가 변경할 수 있도록 합니다. 즉, 사용자가 창의 크기를 마음대로 조정할 수 있습니다.
- 이미지가 원본 크기보다 크더라도 화면에 전체 이미지가 표시되지 않을 수 있습니다. 스크롤바가 나타나며 사용자가 이미지를 스크롤하여 볼 수 있습니다.

2. `cv2.WINDOW_AUTOSIZE` :

- 이 상수는 생성된 **창의 크기**를 이미지의 크기에 자동으로 맞추습니다. 즉, **이미지가 표시되는 창의 크기가 이미지의 크기와 같아집니다.**
- 이 속성을 사용하면 사용자가 **창의 크기를 조정할 수 없으며**, 이미지가 창에 완전히 표시됩니다.

3. 타원을 그리는 `cv2.ellipse()` 함수의 인수를 자세히 설명하시오.

`cv2.ellipse(img, center, axes, angle, starAngle, endAngle, color[, thickness[, lineType[, shift]])]`

img: 타원을 그릴 이미지이다. (도화지에 타원을 그린다고 생각하면 편하다)

center: 타원의 중심 좌표 (x, y) 튜플

axes: (장축, 단축) 튜플

angle: 타원 전체의 각도

startAngle: 그림을 시작할 각도

endAngle: 그림을 종료할 각도

color: 색상 → (R, G, B) 튜플 또는 정수값

thickness: 두께, 기본 값 1 (default)

lineType: 선의 형태 `cv2.LINE_4`, `cv2.LINE_8`, `cv2.LINE_AA` 중 선택

5. 다음 예시 코드의 실행 결과를 설명하시오.

```
# 5 - 1
import numpy as np, cv2
image = np.zeros((300,400), np.uint8)
image[:] = 100
title = 'Window'
cv2.namedWindow(title, cv2.WINDOW_NORMAL)
cv2.moveWindow(title, 100,200)
cv2.imshow(title, image)
cv2.waitKey(0)
cv2.destroyAllWindows()
```

image는 이미지 300,400 사이즈로 들어가는 값은 int형이다. image의 색은 100 (회색)으로 설정한다. 윈도우의 이름은 Window이고 윈도우의 창 관리 옵션은 `cv2.WINDOW_NORMAL`이다. `WINDOW_NORMAL`은 이미지가 기존의 창 사이즈에 맞춰서 들어가게 된다 (만약 기존의 윈도우 창이 (1000,1000)사이즈 라면 이미지도 이 사이즈에 맞춰서 조정된다.) 이와 반대로 `WINDOW_AUTOSIZE`는 윈도우 창이 이미지 사이즈에 맞춰서 조정된다. 전자의 경우는 창 크기를 조정할 수 있고, 후자는 창 크기를 조절할 수 없다. `cv2.waitKey(0)`로 키보드 입력을 무한 대기하고 종료 시 열린 모든 윈도우 창을 닫습니다.

```
# 5 - 2
import numpy as np, cv2
```

```

image = np.zeros((400,600,3), np.uint8)
image[:] = (255,255,255)
pt1, pt2 = (50,100), (200,300)

cv2.line(image, pt1,pt2,(0,255,0),5)
cv2.rectangle(image, pt2, (300,400), (0,0,255), -1 , cv2.LINE_4,1)
#마지막 인수 1 때문에 위치가 달라짐 왜 일까?
#shift 비트 시프트이고 그리기 좌표 값의 축소 비율, default는 0
# 1일때는 1/2배, 2일때는 1/4배 줄어든다.
cv2.imshow("Line & Rectangle", image)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

직선과 직사각형을 그릴 때는 좌측 상단의 좌표와 우측 하단의 좌표가 필요하기에 pt1, pt2 값을 초기화한다.

직선은 (50,100)에서 (200,300)까지 초록색 (0,255,0) 굵기는 5로 image에 그려진다.

직사각형은 (200,300)~(300,400)까지 직사각형을 image에 그린다. 그리기 옵션이 -1이므로 직사각형 전체가 빨간색 (0,0,255)으로 그려진다. Shift 값이 1이므로 기존 좌표, 크기가 1/2배 줄어서 그려진다.

결국 직사각형은 1/2배 줄어든 좌표 (100,150) ~ (150, 200)에서 색이 채워진 빨간색이 그려진다.

6 300행, 400열의 행렬을 회색 바탕색(100)으로 생성해서 500행, 600열의 윈도우에 표시하시오.

```

import numpy as np, cv2

image = np.full((300,400), 100,np.uint8)

title = "title"

cv2.imshow(title, image)
cv2.resizeWindow(title, 600,500)
cv2.waitKey(0)

```

7. 다음 예시 코드는 컴파일 혹은 런타임 에러가 발생한다. 에러가 발생하는 부분을 수정하고 실행 결과를 적으시오.

```

# 7 - 1
import numpy as np, cv2

image = np.zeros((300,400,3), np.uint8)
image[:] = (255,255,255)

pt1, pt2 = (50,130),(200,300)

cv2.line(image, pt1,(100,200))
cv2.line(image, pt2,(100,100,100), (0,0,255))
title = "Line & Rectangle"
cv2.namedWindow(title)
cv2.imshow(title, image)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

오류 1. Cv2.line(image, pt1, (100,200))에 색상 옵션이 없다.

오류 2. Cv2.line(image, pt2, (100,100,100), (0,0,255)) 에서 두 번째 좌표가 3차원이다.

```

# 7 - 1 에러 수정
import numpy as np, cv2

image = np.zeros((300,400,3), np.uint8)
image[:] = (255,255,255)

pt1, pt2 = (50,130),(200,300)

cv2.line(image, pt1,(100,200), (0,0,0))

```

```

cv2.line(image, pt2,(100,100), (0,0,255))
title = "Line & Rectangle"
cv2.namedWindow(title)
cv2.imshow(title, image)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

```

# 7 - 2
import numpy as np, cv2

def onMouse(event, x,y,flags,param):
    global title
    if event == cv2.EVENT_LBUTTONDOWN:
        cv2.circle(image, pt,5,100,1)

    elif event == cv2.EVENT_RBUTTONDOWN:
        cv2.rectangle(image, pt, pt+(30,30),100,2)
        cv2.imshow(title,image)

image = np.ones((300,400),np.uint8) * 255

title = "Draw Event"
cv2.namedWindow(title)
cv2.imshow(title, image)
cv2.setMouseCallback(title, onMouse)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

오류 1. image를 global로 선언하지 않았다.

오류 2. pt를 x와y를 사용해 초기화 하지 않았다.

오류 3. 좌클릭시 결과를 보여주는 cv2.imshow가 없다.

오류 4. pt+(30,30)은 문법 오류이다.

```

# 7 - 2 에러 수정
import numpy as np, cv2
def onMouse(event, x,y,flags,param):
    global title, image
    pt = (x,y)
    if event == cv2.EVENT_LBUTTONDOWN:
        cv2.circle(image, pt,5,100,1)
        cv2.imshow(title,image)

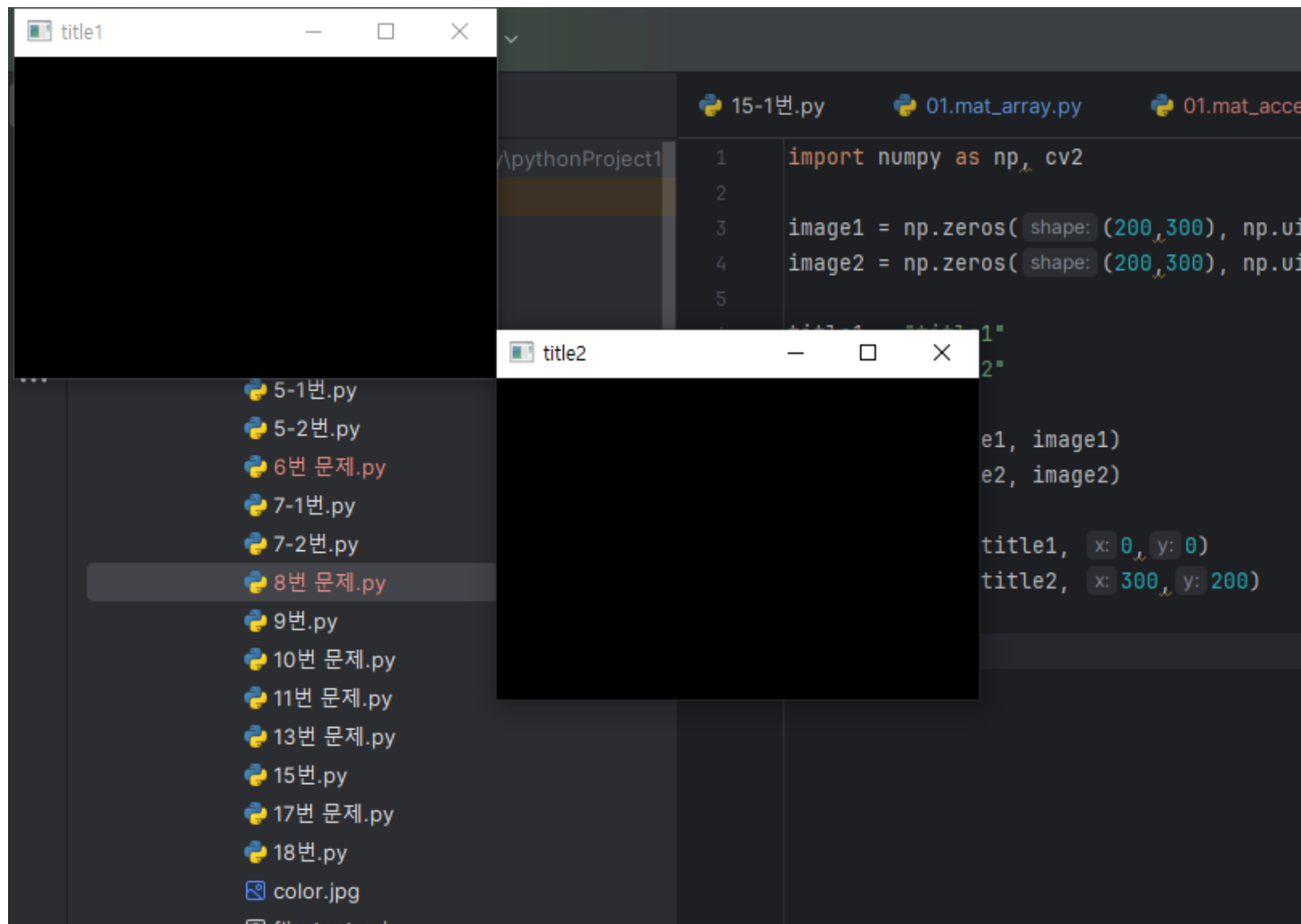
    elif event == cv2.EVENT_RBUTTONDOWN:
        pt2 = (pt[0]+30, pt[1]+30)
        cv2.rectangle(image, pt, pt2,100,2)
        cv2.imshow(title,image)

image = np.ones((300,400),np.uint8) * 255

title = "Draw Event"
cv2.namedWindow(title)
cv2.imshow(title, image)
cv2.setMouseCallback(title, onMouse)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

8번 문제. 200행 300열 행렬 2개를 만들어서 다음과 같이 배치하시오



```
import numpy as np, cv2

image1 = np.zeros((200,300), np.uint8)
image2 = np.zeros((200,300), np.uint8)

title1 = "title1"
title2 = "title2"

cv2.imshow(title1, image1)
cv2.imshow(title2, image2)

cv2.moveWindow(title1, 0,0)
cv2.moveWindow(title2, 300,200)

cv2.waitKey(0)
```

9. 600행, 400열의 윈도우를 만들고, 영상 안의 (100,100) 좌표에 200x300 크기의 빨간 색 사각형을 그리시오.

```
import numpy as np, cv2

image1 = np.zeros((600,400,3), np.uint8)+255 # 색상이 있으므로 3차원!

title1 = "title1"
pt1, pt2 =(100,100) , (300,400)
red = (0,0,255)

cv2.rectangle(image1, pt1, pt2, red,cv2.FILLED)

cv2.imshow(title1,image1)
cv2.waitKey(0)
```



11. 10번 연습 문제에서 다음을 추가하여 프로그램을 작성하시오.

기존의 10번 문제 코드

```
import numpy as np, cv2

def onMouse(event, x,y, flags, param):
    global image, title

    if event == cv2.EVENT_RBUTTONDOWN:
        pt = (x,y)
        r= 20
        cv2.circle(image, pt, 20, 0,1)
        cv2.imshow(title,image)
    elif event == cv2.EVENT_LBUTTONDOWN:
        pt1,pt2 = (x,y),(x+30,y+30)
        cv2.rectangle(image, pt1,pt2,0,1)
        cv2.imshow(title,image)

image = np.ones((1000,1000), np.uint8) * 255
title="title"

cv2.imshow(title, image)
cv2.setMouseCallback(title, onMouse)
cv2.waitKey(0)
```

11번 문제 코드

```
import numpy as np, cv2

def onMouse(event, x, y, flags, param):
```

```

global title, image

if event == cv2.EVENT_RBUTTONDOWN: # 우클릭 시
    pt = (x, y)
    # 흑백 이미지 이므로 color는 0
    cv2.circle(image, pt, radius, 0, line_thick)
    cv2.imshow(title, image)
elif event == cv2.EVENT_LBUTTONDOWN:
    pt1, pt2 = (x, y), (x + 30, y + 30)
    cv2.rectangle(image, pt1, pt2, 0, line_thick)
    cv2.imshow(title, image)

def onChange1(value):
# 함수에서 바꾼 radius가 전역에 적용되도록 line_thick 선언
    global image, title, line_thick

    line_thick = value
    cv2.setTrackbarPos(track_bar1, title, line_thick)
    cv2.imshow(title, image)

def onChange2(value):
# 함수에서 바꾼 radius가 전역에 적용되도록 global 선언
    global image, title, radius

    radius = value
    cv2.setTrackbarPos(track_bar2, title, radius)
    cv2.imshow(title, image)

title = "Draw circle and rectangle"
track_bar1 = "line" # 직선 트랙바
track_bar2 = "radius" # 원 반지름 트랙바

image = np.zeros((300, 300), np.uint8)
image[:] = 255
line_thick = 1
radius = 20
black = (0, 0, 0)

cv2.imshow(title, image)
cv2.setMouseCallback(title, onMouse)
cv2.createTrackbar(track_bar1, title, line_thick, 10, onChange1)
cv2.createTrackbar(track_bar2, title, radius, 50, onChange2)
cv2.waitKey(0)
cv2.destroyAllWindows()

```

12번 문제



화살표 키로 트랙바를 이동하여 밝기를 조절하는 코드를 추가하시오.

```

import numpy as np
import cv2

def onChange(value): # 트랙바 콜백 함수
    global image, title # 전역 변수 참조

    #add_value = value - int(image[0][0]) # 트랙바 값과 영상 화소값 차분
    #print("추가 화소값:", add_value)
    #image = image + add_value # 행렬과 스칼라 덧셈 수행
    image[:] = cv2.getTrackbarPos('Brightness', title) # 바뀐 값으로 바로 화면 색 변경되도록 수정...
    cv2.imshow(title, image)

```

```

image = np.zeros((300, 500), np.uint8) # 영상 생성

title = 'Trackbar Event'
cv2.imshow(title, image)
cv2.createTrackbar('Brightness', title, image[0][0], 255, onChange) # 트랙바 콜백 함수 등록

while True:
    key = cv2.waitKeyEx(100) # 100ms동안 키 이벤트 대기
    if key == 27: break # esc 키 누르면 종료

    # 왼쪽< 화살표 키 입력
    if key == 0x250000:
        value = cv2.getTrackbarPos('Brightness', title)
        cv2.setTrackbarPos('Brightness', title, value - 1)
    # 오른쪽> 화살표키 입력
    elif key == 0x270000:
        value = cv2.getTrackbarPos('Brightness', title)
        cv2.setTrackbarPos('Brightness', title, value + 1)

    image[:] = cv2.getTrackbarPos('Brightness', title) # 바뀐 값으로 바로 화면 색 변경되도록 수정...
    cv2.imshow(title, image)

cv2.destroyAllWindows()

```

```

import numpy as np, cv2

def onChange(value):
    global image
    #
    # add_value = value - int(image[0][0])
    # image = image + add_value
    image[:] = value
    cv2.imshow(title, image)

image = np.zeros((300,500), np.uint8)

title, bar_name = "ex12", "br"
cv2.imshow(title, image)
cv2.createTrackbar(bar_name, title, image[0][0], 255, onChange)

while(True):
    key = cv2.waitKeyEx(10)
    if key == 0x250000:
        value = cv2.getTrackbarPos(bar_name, title)
        cv2.setTrackbarPos(bar_name,title, value-10)
    elif key == 0x270000:
        value = cv2.getTrackbarPos(bar_name, title)
        cv2.setTrackbarPos(bar_name, title,value+10)
    elif key == 27 : break

```

13. 컬러 영상 파일을 적재하여 “test.jpg”와 “tes.png” 파일로 각각 저장하시오 이때, 영상 파일을 가장 좋은 화질로 압축하시오.

13번 문제 코드

```

import cv2

image = cv2.imread('color.jpg') # color.jpg 읽어오기
if image is None : raise Exception("이미지 읽기 에러") # 예외처리
# JPB 압축 설정 가장 높은값 100 (default - 95)
params_jpg = [cv2.IMWRITE_JPEG_QUALITY, 100]

```

```
# PNG 압축 설정 가장 높은 값 - 9 (default - 3)
params_png = [cv2.IMWRITE_PNG_COMPRESSION, 9]

cv2.imwrite("test.jpg",image, params_jpg)
cv2.imwrite("test.png",image, params_png)
```

14. 마우스 중간 버튼을 클릭하여 타원을 그리세요

```
elif event == cv2.EVENT_MBUTTONDOWN:
    if pt[0] < 0:
        pt = (x,y)
    else:
        pt2 = (x,y)
        center = np.abs(np.add(pt,pt2)) //2
        size = np.abs(np.subtract(pt,pt2)) //2
        cv2.ellipse(image, tuple(center), tuple(size),0,0,360,(0,0,255),2)
        cv2.imshow(title, image)
        pt = (-1, -1)
```

15. 심화 예제_4.5.2인 18.camera_attr.py를 수정해서 트랙바로 카메라 영상의 밝기와 대비 변경할 수 있도록 수정하시오

```
import cv2

def zoom_bar(value):
    global capture
    capture.set(cv2.CAP_PROP_ZOOM, value)

def focus_bar(value):
    global capture
    capture.set(cv2.CAP_PROP_FOCUS,value)

def brightness_bar(value): # 밝기 수정 함수
    global capture
    capture.set(cv2.CAP_PROP_BRIGHTNESS, value)

def contrast_bar(value): # 대비값 수정 함수
    global capture
    capture.set(cv2.CAP_PROP_CONTRAST, value)

capture = cv2.VideoCapture(0)
if not capture.isOpened() : raise Exception("카메라 연결 오류")

capture.set(cv2.CAP_PROP_FRAME_WIDTH,400) # 너비 400
capture.set(cv2.CAP_PROP_FRAME_HEIGHT,300) # 높이 300
capture.set(cv2.CAP_PROP_AUTOFOCUS, 0) # 자동 초점 중지
capture.set(cv2.CAP_PROP_BRIGHTNESS,100) #프레임 밝기 100으로 초기화

title ="Change Camera Properites"
cv2.namedWindow(title)
cv2.createTrackbar('zoom',title, 0,10,zoom_bar)
cv2.createTrackbar('focus',title, 0,40,focus_bar)
# 밝기 트랙바 설정
cv2.createTrackbar('brightness',title, 0,40,brightness_bar)
# 대비 트랙바 설정
cv2.createTrackbar('contrast',title, 0,40,contrast_bar)

while True:
    ret, frame = capture.read() # 한프레임씩 읽기
    if not ret : break # 빈 프레임이면 중지
    # 딜레이는 30ms, 값 입력시 중지
```



```

        if cv2.waitKey(30) >= 0: break

        cv2.imshow(title, frame) # frame 이미지 윈도우(title)에 출력

# cv2.CAP_PROP_CONTRAST 대비
# cv2.CAP_PROP_BRIGHTNESE 밝기

```

16.

PC 카메라를 통해서 받아온 프레임에 다음의 영상처리를 수행하고, 결과 영상을 윈도우에 표시하는 프로그램 작성



1. (200,100) 좌표에서 100x200 크기의 관심 영역 지정
2. 관심 영역에서 녹색 성분을 50만큼 증가
3. 관심 영역의 테두리를 두께 3의 빨간색으로 표시한다.

```

import cv2

capture = cv2.VideoCapture(0)
if not capture.isOpened(): raise Exception("카메라 연결 안됨")

capture.set(cv2.CAP_PROP_FRAME_WIDTH, 400)
capture.set(cv2.CAP_PROP_FRAME_HEIGHT, 300)
capture.set(cv2.CAP_PROP_AUTOFOCUS, 0)
capture.set(cv2.CAP_PROP_BRIGHTNESS, 0)

title = "title"
cv2.namedWindow(title)

while True:
    ret, frame = capture.read()

    if ret is None: break
    if cv2.waitKey(100) == 27: break # esc누르면 종료

    blue, green, red = cv2.split(frame)
    # cv2.add(green[100:300, 200:300], 50) #첫번째 영상 , 두번째 영상 or 스칼라, 결과를 담을 영상
    green[100:300,200:300] = green[100:300, 200:300] + 50
    frame = cv2.merge([blue, green, red]) # 단일 채널 영상 합성

    cv2.rectangle(frame, (200, 100), (300, 300), (0, 0, 255), 3)

    cv2.imshow(title, frame)

capture.release()

```

17. PC 카메라를 통해서 받아온 프레임을 좌우로 뒤집어서 “flip_test.avi” 이름의 동영상 파일로 저장하는 프로그램을 작성하시오.

17번 코드

```

import cv2

capture =cv2.VideoCapture(0) # 0번 카메라에 연결
if capture.isOpened() == False : raise Exception("카메라 연결 오류")

fps = 15.0 # 15 프레임
delay = round(1000/15)
size = (640, 360) # 비디오 크기
fourcc = cv2.VideoWriter_fourcc(*'DIVX') #코덱 DIVX사용

writer = cv2.VideoWriter('flip_test.avi', fourcc, fps, size)

```

```

if writer.isOpened() == False : raise Exception("영상 개방 오류") # 예외처리

#카메라 세팅
capture.set(cv2.CAP_PROP_FRAME_WIDTH, size[0]) # 카메라 프레임 너비
capture.set(cv2.CAP_PROP_FRAME_HEIGHT, size[1]) # 카메라 프레임 높이
capture.set(cv2.CAP_PROP_AUTOFOCUS, 0) # 오토포커싱 중지
capture.set(cv2.CAP_PROP_BRIGHTNESS, 100) # 프레임 밝기 초기화

while True:
    ret, frame = capture.read()
    if not ret : break
    if cv2.waitKey(delay) >= 0 : break

    frame = cv2.flip(frame,1) # 0 - 상하, 1 - 좌우, -1 상하좌우 반전
    writer.write(frame) # 좌우 반전한 프레임 저장
    cv2.imshow("View Frame from Camera", frame)

writer.release()
capture.release()

```