

OpenCV의 주요 함수 기능별 정리

기능 분류	함수 이름 (인자 포함)	기능 설명
이미지 입출력	cv2.imread(filename, flags=1)	이미지를 파일에서 읽는다 (flags=0: 흑백, 1: 컬러)
	cv2.imwrite(filename, img)	이미지를 파일로 저장한다
	cv2.imshow(winname, mat)	이미지를 윈도우에 띄운다
	cv2.waitKey(delay=0)	키 입력을 대기한다 (delay=0이면 무한 대기)
	cv2.destroyAllWindows()	모든 OpenCV 윈도우를 닫는다
크기 및 변환	cv2.resize(src, dsize, fx=0, fy=0, interpolation=...)	이미지의 크기를 조절한다
	cv2.getRotationMatrix2D(center, angle, scale)	회전을 위한 변환 행렬을 생성한다
	cv2.warpAffine(src, M, dsize)	Affine 변환을 수행한다
	cv2.cvtColor(src, code)	색상 공간을 변환한다 (예: BGR ↔ GRAY, HSV 등)
필터링	cv2.GaussianBlur(src, ksize, sigmaX)	가우시안 블러를 적용한다
	cv2.medianBlur(src, ksize)	미디안 블러를 적용한다
	cv2.bilateralFilter(src, d, sigmaColor, sigmaSpace)	양방향 필터를 적용한

기능 분류	함수 이름 (인자 포함)	기능 설명
		다
에지 검출	cv2.Canny(image, threshold1, threshold2)	Canny 알고리즘으로 에지를 검출한다
	cv2.Sobel(src, ddepth, dx, dy, ksize)	Sobel 필터로 경계를 검출한다
	cv2.Laplacian(src, ddepth)	Laplacian 필터로 경계를 검출한다
이진화	cv2.threshold(src, thresh, maxval, type)	이미지를 이진화한다
	cv2.adaptiveThreshold(src, maxValue, adaptiveMethod, thresholdType, blockSize, C)	적응형 임계값 이진화를 수행한다
그리기 함수	cv2.line(img, pt1, pt2, color, thickness)	직선을 그린다
	cv2.rectangle(img, pt1, pt2, color, thickness)	사각형을 그린다
	cv2.circle(img, center, radius, color, thickness)	원을 그린다
	cv2.putText(img, text, org, fontFace, fontScale, color, thickness)	이미지 위에 텍스트를 출력한다
윤곽선 처리	cv2.findContours(image, mode, method)	윤곽선을 검출한다
	cv2.drawContours(image, contours, contourIdx, color, thickness)	윤곽선을 그린다
객체 검출	cv2.CascadeClassifier('path').detectMultiScale(img, ...)	Haar cascade로 얼굴 등 객체를 검출한다
비트 연산	cv2.bitwise_and(src1, src2, mask=None)	비트 AND 연산
	cv2.bitwise_or(src1, src2, mask=None)	비트 OR 연산
	cv2.bitwise_not(src)	비트 NOT 연산

기능 분류	함수 이름 (인자 포함)	기능 설명
카메라 입력	cv2.VideoCapture(index)	카메라 장치를 열거나 비디오 파일을 연다 (index=0은 기본 카메라)
	cap.read()	프레임을 읽는다 (ret, frame = cap.read())
	cap.release()	비디오 캡처 객체를 해제한다
비디오 출력	cv2.VideoWriter(filename, fourcc, fps, frameSize)	비디오 파일로 저장하기 위한 객체 생성
	out.write(frame)	프레임을 비디오로 저장한다
	cv2.VideoWriter_fourcc(*'XVID')	비디오 코덱 설정 (예: 'XVID', 'MJPG')