

임베디드 응용 및 실습 9 주차 과제

2022180034 김한별

1. 얼굴 검출

1. 코드

```
import numpy as np
import cv2

face_cascade = cv2.CascadeClassifier("/home/pi/hello-git/log-
git/week10/haarcascade_frontalface_default.xml")

cap = cv2.VideoCapture(0, cv2.CAP_V4L) # 첫번째 카메라 영상
cap.set(cv2.CAP_PROP_FRAME_WIDTH, 640) # 카메라 영상 넓이
cap.set(cv2.CAP_PROP_FRAME_HEIGHT, 480) # 카메라 영상 높이

while (True):
    ret, img = cap.read()
    img = cv2.flip(img, -1)

    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_cascade.detectMultiScale(gray, 1.2, 5)
    print("Number of faces detected: " + str(len(faces)))

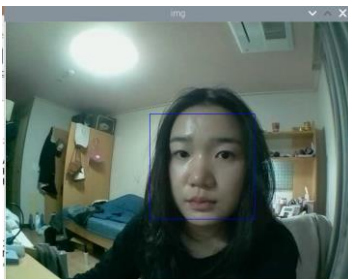
    for (x, y, w, h) in faces:
        img = cv2.rectangle(img, (x, y), (x + w, y + h), (255, 0, 0), 1)
        roi_gray = gray[y:y + h, x:x + w]
        roi_color = img[y:y + h, x:x + w]

    cv2.imshow('img', img)

    k = cv2.waitKey(30) & 0xff
    if k == 27: # Esc 키를 누르면 종료
        break

cap.release()
cv2.destroyAllWindows()
```

2. 결과



2. 흰색과 노란색 선 검출

1. 코드

```
import cv2 as cv
import numpy as np

# 이미지 불러오기
img = cv.imread('/home/pi/hello-git/log-git/week10/imgs/4.jpg')
if img is None:
    print("Error: Unable to load image.")
else:
    # BGR 이미지를 HSV 색상 공간으로 변환
    hsv = cv.cvtColor(img, cv.COLOR_BGR2HSV)

    # 흰색 범위 설정 (HSV 색상 공간에서)
    lower_white = np.array([0, 0, 200])
    upper_white = np.array([500, 30, 255])

    # 노란색 범위 설정
    lower_yellow = np.array([20, 150, 150])
    upper_yellow = np.array([30, 255, 255])

    # 흰색과 노란색 마스크 생성
    mask_white = cv.inRange(hsv, lower_white, upper_white)
    mask_yellow = cv.inRange(hsv, lower_yellow, upper_yellow)

    # 두 마스크를 합쳐서 흰색과 노란색만 남기기
    mask_combined = cv.bitwise_or(mask_white, mask_yellow)

    # 마스크를 사용해 흰색과 노란색 부분만 남기고 나머지는 검게 처리
    result = cv.bitwise_and(img, img, mask=mask_combined)

    # 결과 이미지 출력
    cv.imshow('Filtered Image (White and Yellow)', result)
    cv.waitKey(0)
    cv.destroyAllWindows()
```

2. 결과

<원본 사진>

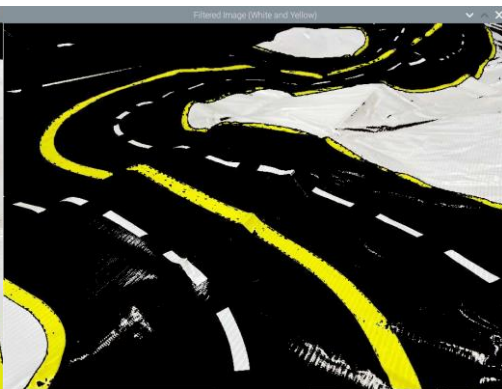


<결과 사진>

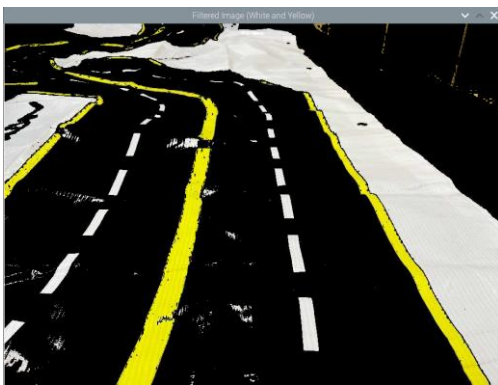
-1 번째 사진



-2 번째 사진



-3 번째 사진



-4 번째 사진

