**임베디드 응용 및 실습 9주차 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()

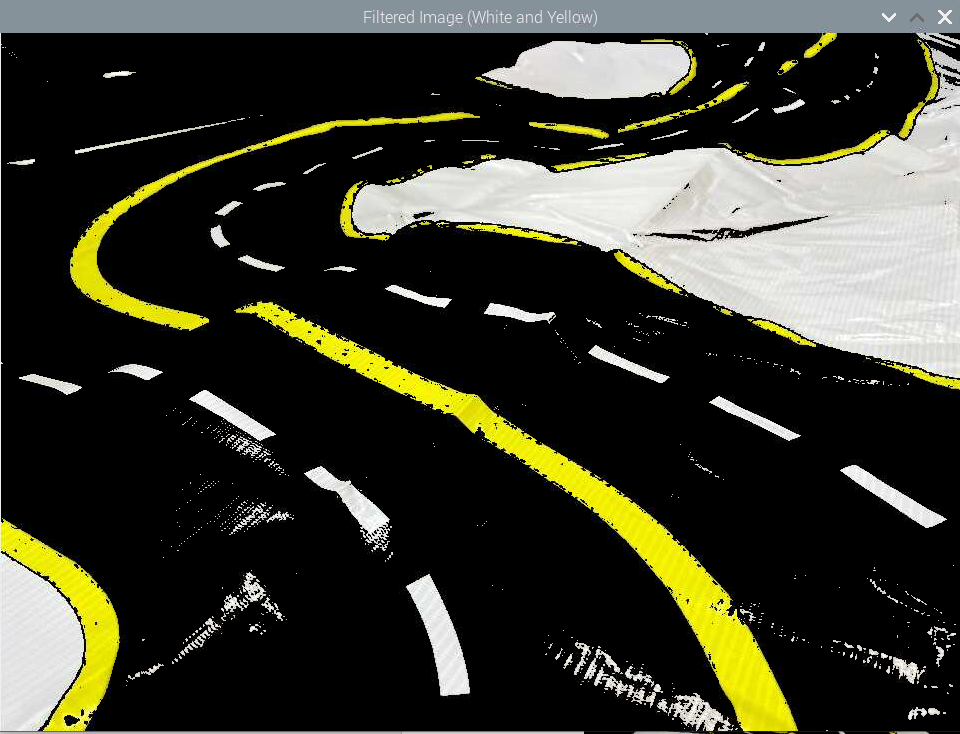
1. 결과

<원본 사진>



<결과 사진>

-1번째 사진 -2번째 사진



-3번째 사진 -4번째 사진

