

OpenCV day1 과제 1 보고서

20기 예비 김도현

목차

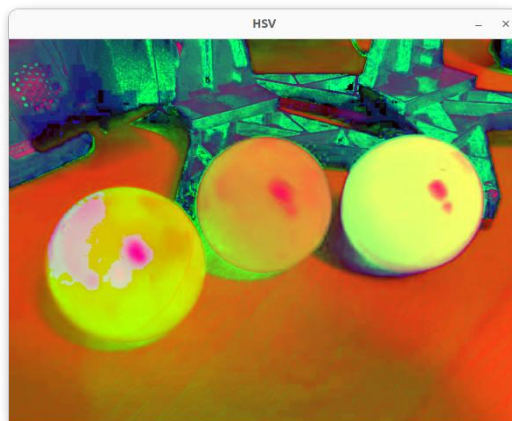
1. 코드
2. 가우시안 필터
 - 2-1. 적용
 - 2-2. 가우시안 필터를 적용한 이미지의 차이점
 - 2-3. 가우시안 필터를 사용하는 이유

코드

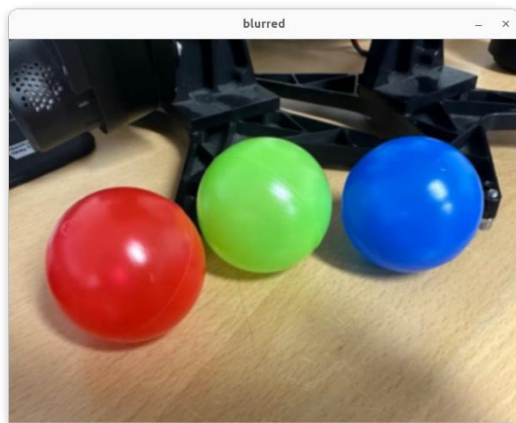
- 이미지 파일 불러오기
`cv::imread("balloon.png")`



- HSV형식으로 변환
`cv::cvtColor(img, hsv, cv::COLOR_BGR2HSV);`



- Gaussian blur 적용
`cv::GaussianBlur(img, blurred, cv::Size(5, 5), 0);`



- 이미지 출력

```
cv::imshow("Original", img);
```

- Inrange

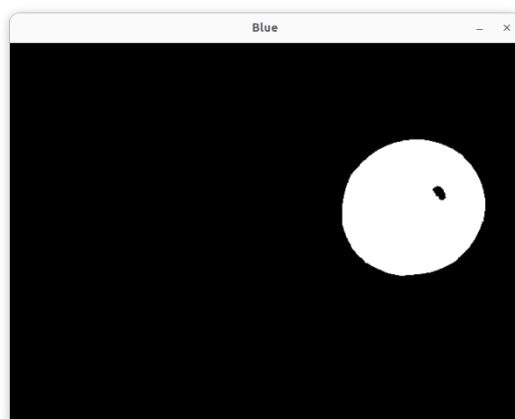
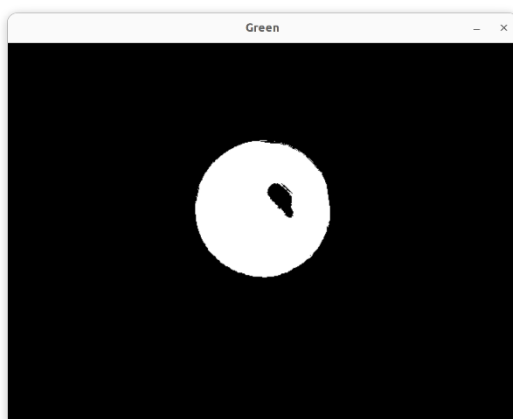
```
cv::Mat green(cv::Mat original)
```

```
cv::Mat blue(cv::Mat original)
```

원본 이미지에 `inRange()`를 적용하여 해당 범위에 해당하는 바이너리 이미지를 반환한다.

공 가운데 감지되지 않은 공간은 빛 반사로 인해 하얀색으로 감지된 부분이 입력된 색 범위에서 벗어났기 때문이다.

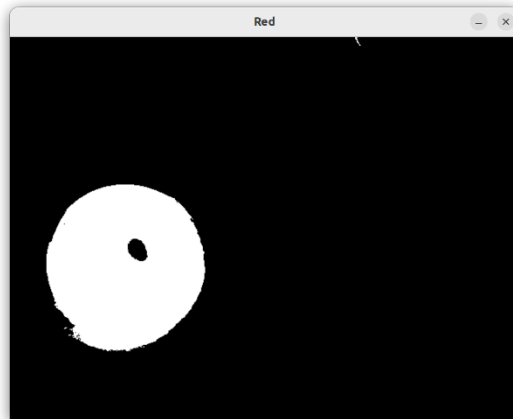
해당 부분을 인식하기 위해 색 범위를 조정한다면 의도하지 않은 부분까지 포함되기 때문에 조정을 중단하였다.



- 빨간공 inrange

```
cv::Mat red(cv::Mat original)
```

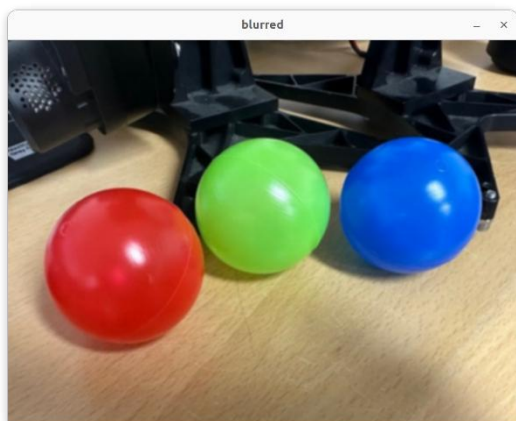
빨간색의 H 값은 0과 179 근처로 분포하기 때문에 0 근처와 179 근처의 범위를 각각 `inRange()`로 마스크 처리한 후 `bitwise_or()`을 통해 하나의 바이너리 이미지로 합성하여 반환한다.



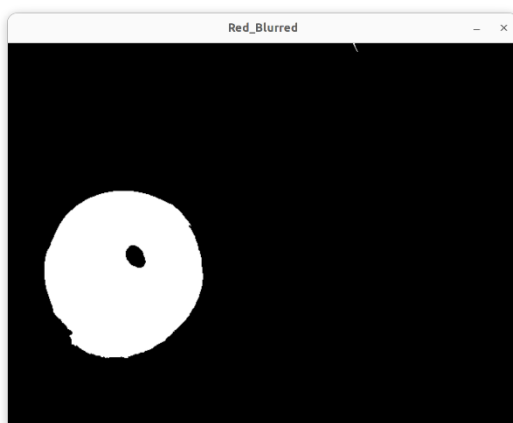
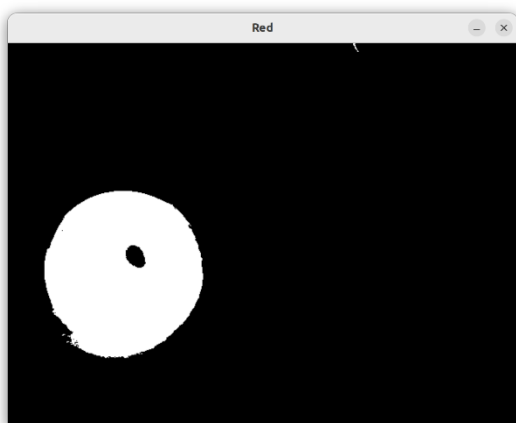
Gaussian filter

적용

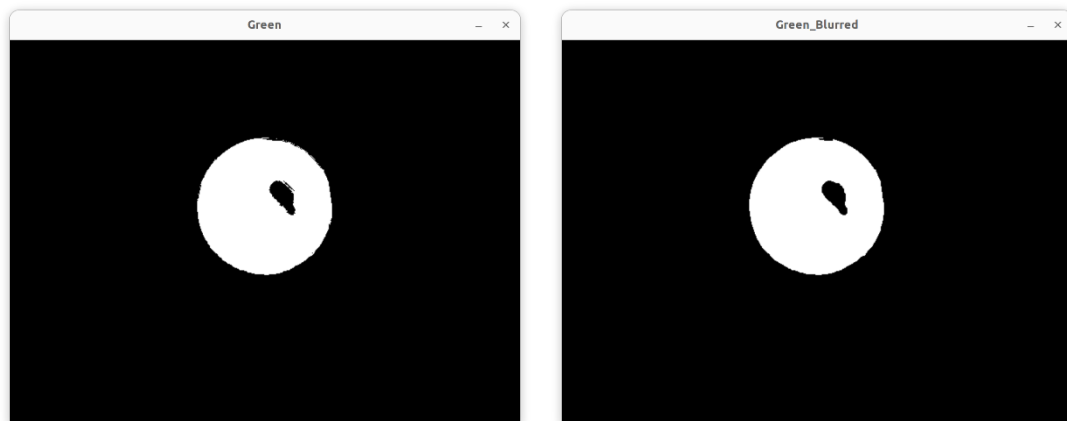
가우시안 분포를 기반으로 주변 픽셀의 평균을 적용하여 이미지의 노이즈를 제거하는 필터링 방식이다.



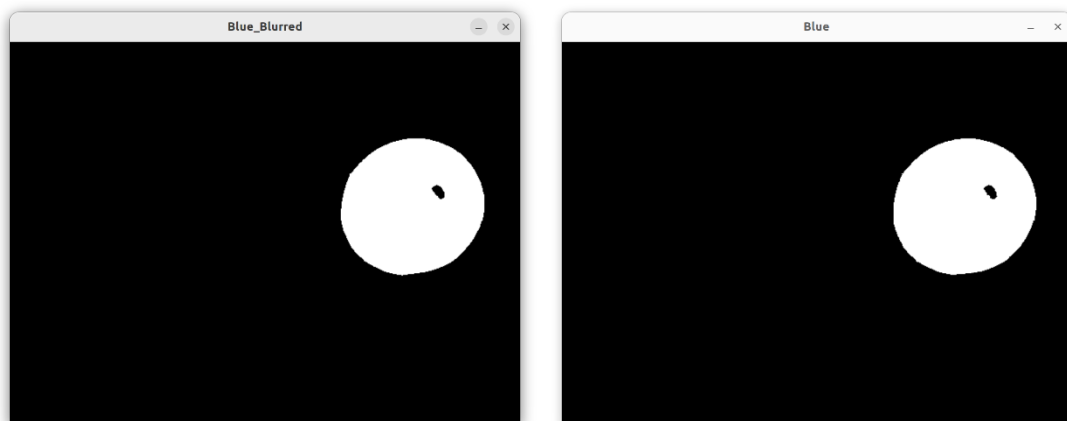
빨간 공 가우시안 필터 적용 전후



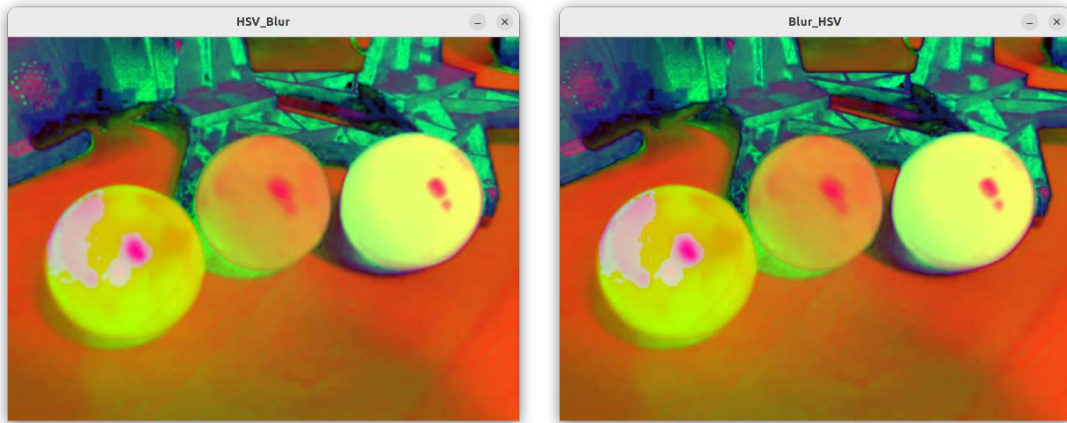
초록 공 가우시안 필터 적용 전후



파란 공 가우시안 필터 적용 전후



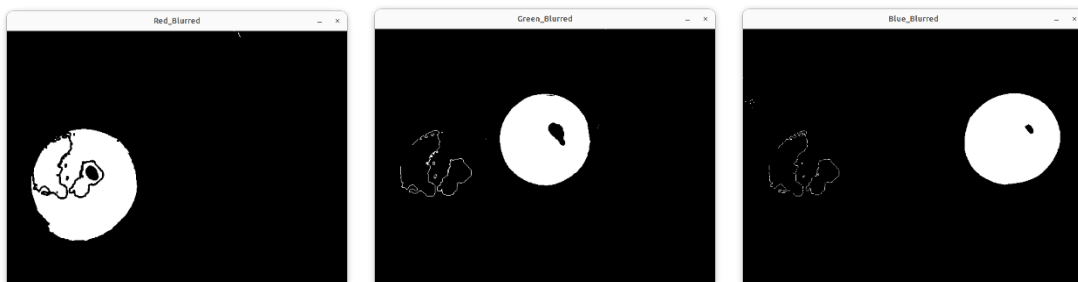
`inRange()` 함수를 적용하기 위해서는 RGB로 표현된 이미지를 HSV 색공간으로 변환해야 한다. 이때, 가우시안 필터를 적용하는 방법에는 두 가지가 있다. 하나는 RGB 이미지에 먼저 가우시안 필터를 적용한 후 HSV로 변환하는 방식이고, 다른 하나는 HSV로 변환한 이미지에 가우시안 필터를 적용하는 방식이다.



HSV로 변환된 이미지에 가우시안 필터를 적용한 이미지(좌)

가우시안 필터를 적용한 이미지를 HSV로 변환한 이미지(우)

초기에는 HSV로 변환된 이미지에 가우시안 필터를 적용하는 방식을 사용하였으나 필터링 이후 오히려 검출 정확도가 감소하는 문제가 발생하였다.



원인은 필터링 과정에서 색상 경계가 변화하면서 `inRange()` 함수로 검출되던 경계 픽셀들이 범위 밖으로 벗어난 것으로 추정되며 원본 이미지에 가우시안 필터를 먼저 적용하고 이후 HSV 색공간으로 변환하자 문제가 해결되었다.

가우시안 필터를 적용한 이미지의 차이점

가우시안 필터를 적용한 이미지에서는 노이즈가 감소하여 경계면을 보다 뚜렷하게 검출하였다.

가우시안 필터를 사용하는 이유

가우시안 필터는 이미지의 고주파 신호를 평탄화하는 LPF(Low-Pass Filter)로 기능한다.

이를 통해 색상의 경계가 완화되면서 이미지의 잡음과 급격한 밝기 변화를 완화하며

inRange()를 통한 마스킹 과정에서 필터링 범위가 더 효과적으로 적용되도록 한다.

이로 인해 노이즈가 완화되면서 객체 검출의 정확도를 향상시키고 검출 오류의 가능성을 감소시킬 수 있다.