

REPORT



과목명 | 컴퓨터비전

담당교수 | 송현주

학과 | 컴퓨터학부

학년 | 4

학번 | 20182588

이름 | 김민수

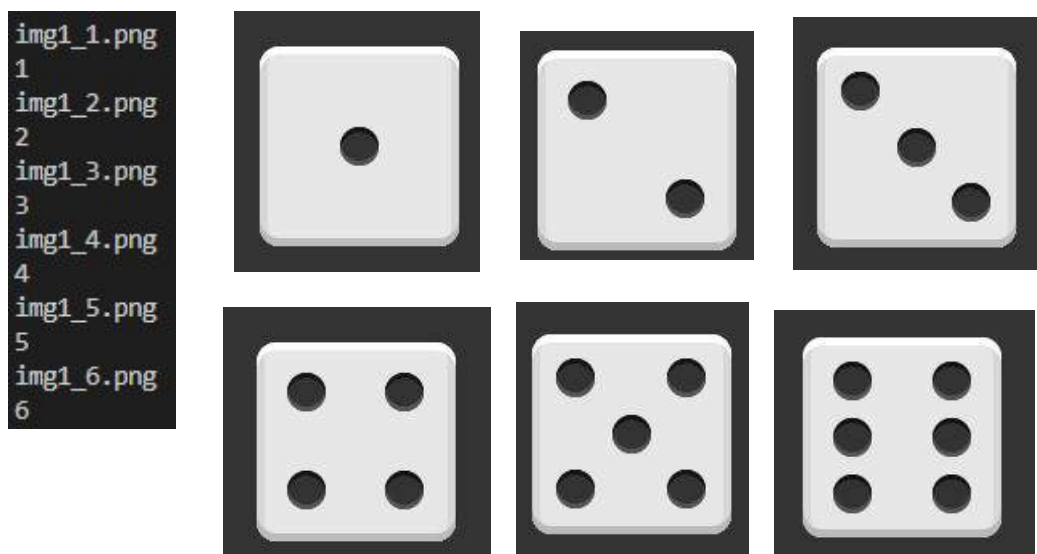
제출일 | 22.12.01

1. 사각형 내의 원의 수 세기

코드 흐름

1. 이미지를 Grayscale로 읽어온다.
2. 객체 검출을 위해 cv.threshold로 이진화를 수행한다.
3. cv.connectedComponentsWithStats로 레이블링을 수행한다.
4. 검출된 객체의 x,y 좌표 height, width를 이용하여 부분 이미지를 추출한다.
5. 부분 이미지에서 cv.HoughCircles를 이용해 원을 검출한다.
6. 검출된 원의 숫자를 console에 출력한다.

출력 결과



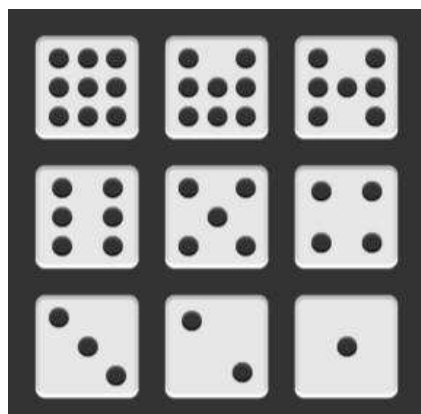
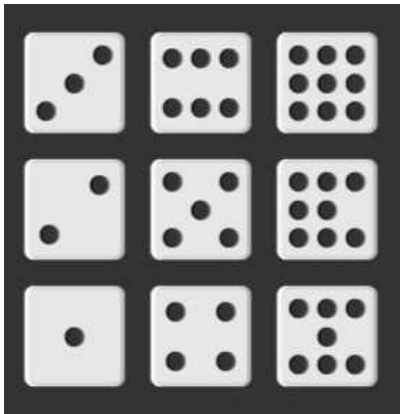
2. 사각형 내의 원의 수 세기

코드 흐름

1. 이미지를 Grayscale로 읽어온다.
2. 객체 검출을 위해 cv.threshold로 이진화를 수행한다.
3. cv.connectedComponentsWithStats로 레이블링을 수행한다.
4. 각각의 검출된 객체의 부분 이미지를 추출한다.
5. 부분 이미지에서 원을 검출하여 원의 숫자를 arr 배열에 추가한다.
6. arr 배열을 오름차순으로 정렬 후 console에 출력한다.

출력 결과

```
img2_1.png  
[1, 2, 3, 4, 5, 6, 7, 8, 9]  
img2_2.png  
[1, 2, 3, 4, 5, 6, 7, 8, 9]
```



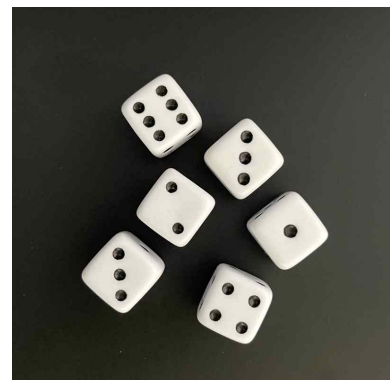
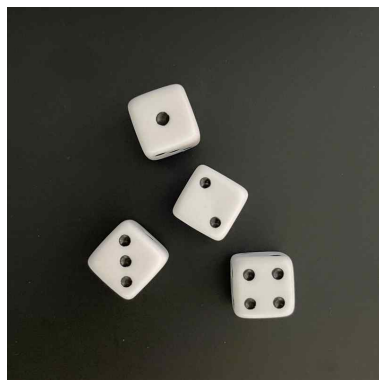
3. 주사위 읽기 (1)

코드 흐름

1. 이미지를 Color로 읽어온다.
2. YCrCb 색 공간으로 변환한다.
3. 채널을 나누어 Y 성분에 대해서 평활화를 수행한다.
4. 객체 검출을 위해 cv.threshold로 이진화를 수행한다.
5. Grayscale로 변환한다.
6. cv.adaptiveThreshold로 적응형 이진화를 수행한다.
7. cv.morphologyEx로 열기 연산을 수행한다.
8. cv.medianBlur로 미디언 필터링을 수행한다.
7. cv.connectedComponentsWithStats로 레이블링을 수행한다.
8. 각각의 검출된 객체의 부분 이미지를 추출한다.
9. 부분 이미지에서 원을 검출하여 원의 숫자를 arr 배열에 추가한다.
10. arr 배열을 오름차순으로 정렬 후 console에 출력한다.

출력 결과

```
img3_1.png  
[2, 3, 4, 6, 6]  
img3_2.png  
[1, 3, 4, 5]  
img3_3.png  
[1, 2, 3, 4]  
img3_4.png  
[1, 2, 3, 3, 4, 6]
```



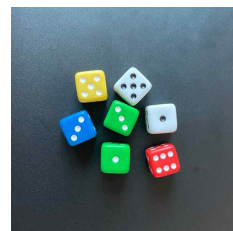
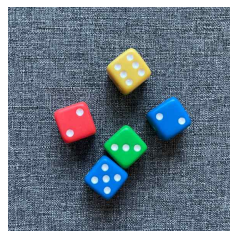
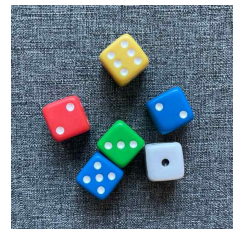
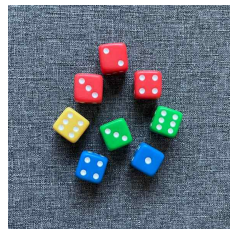
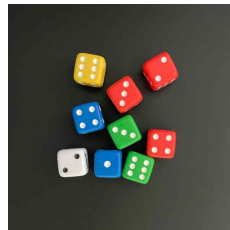
4. 주사위 읽기 (2)

코드 흐름

1. 이미지를 Color로 읽어온다.
2. Grayscale로 변환한다
3. cv.adaptiveThreshold로 적응형 이진화를 수행한다.
4. cv.morphologyEx로 열기 연산을 수행한다.
5. cv.connectedComponentsWithStats로 레이블링을 수행한다.
6. 각각 검출된 객체의 위치 정보를 토대로 원본 이미지로부터 부분 이미지를 추출한다.
7. 부분 이미지에서 원을 검출하기 위해 명암비를 조절한다.
8. 원 검출을 수행한 후 원의 숫자를 arr 배열에 추가한다.
9. arr 배열을 오름차순으로 정렬 후 console에 출력한다.

출력 결과

```
img4_1.png  
[1, 2, 2, 3, 3, 4, 4, 6, 6]  
img4_2.png  
[1, 1, 2, 3, 3, 4, 4, 6, 6]  
img4_3.png  
[1, 2, 3, 3, 4, 4, 6, 6]  
img4_4.png  
[1, 2, 2, 3, 5, 6]  
img4_5.png  
[2, 2, 3, 5, 6]  
img4_6.png  
[1, 1, 3, 3, 5, 5, 6]
```



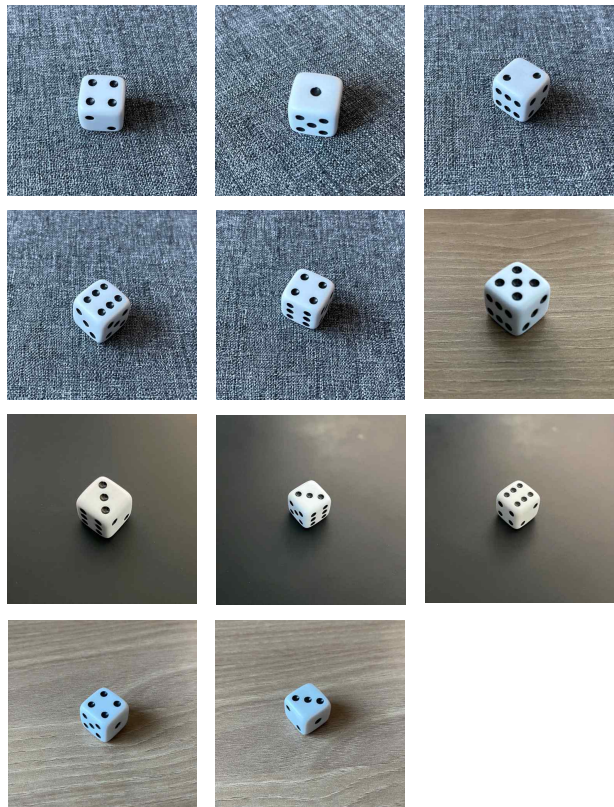
5. 주사위 읽기 (3)

코드 흐름

1. 이미지를 Color로 읽어온다.
2. Grayscale로 변환한다
3. cv.adaptiveThreshold로 적응형 이진화를 수행한다.
4. cv.morphologyEx로 열기 연산을 수행한다.
5. cv.connectedComponentsWithStats로 레이블링을 수행한다.
10. 각각 검출된 객체의 위치 정보를 토대로 원본 이미지로부터 부분 이미지를 추출한다.
6. 부분 이미지에서 원을 검출한다. (가장 넓은 면적의 주사위 눈금만 읽어오도록 cv.houghCircles의 하이퍼 파라미터를 수정한다.)
7. 검출된 원의 숫자를 console에 출력한다.

출력 결과

```
img5_1.png
4
img5_2.png
1
img5_3.png
2
img5_4.png
6
img5_5.png
4
img5_6.png
5
img5_7.png
3
img5_8.png
3
img5_9.png
6
img5_10.png
4
img5_11.png
3
```



참고 문헌

<https://numpy.org/doc/stable/index.html>

<https://docs.opencv.org/4.x/index.html>