

Open CV로 뭘 할 수 있는가?



INDEX

1

Open CV가 뭔가요?

2

그래서 뭘 할 수 있나요?

3

뭘 공부했나요?

1. 주제선정
2. 시스템 설계 및 개발
3. 성과

4

Q&A

1 Open CV가 뭔가요?

OpenCV

OpenCV (Open Source Computer Vision Library)

실시간 컴퓨터 비전을 목적으로 한 프로그래밍 Library 이다.

실시간 이미지 프로세싱에 중점을 둔 라이브러리이다. 인텔 CPU에서 사용되는 경우 속도의 향상을 볼 수 있는 IPP (Intel Performance Primitives) 를 지원한다.

이 라이브러리는 윈도우, 리눅스 등에서 사용 가능한 크로스 플랫폼이며 오픈소스 BSD 허가서 하에서 무료로 사용할 수 있다. OpenCV 는 TensorFlow , Torch / PyTorch 및 Caffe 의 딥러닝 프레임워크를 지원한다

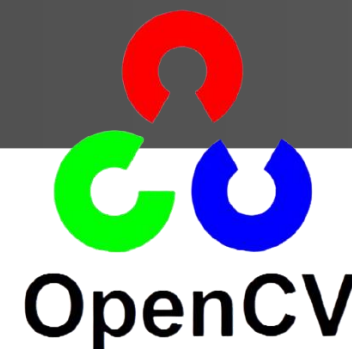
컴퓨터 비전 분야는 사람이 시각 정보를 입력값으로 하여 행동하기 이전에 생각하고 판단하는 부분을 컴퓨터가 대신하도록 하는 인공지능 관련 학문이며 다만, 시각적인 입력 데이터, 즉 영상을 주로 다룬다는 것이 차이점이다.

OpenCV는 Computer Vision 관련 프로그래밍을 쉽게 할 수 있도록 도와주는 Open Library이다. OpenCV는 영상처리, 3D 구성, 추적, 기계학습, 인식 그리고 딥러닝까지 유용한 기능이 아주 많으며 상업적인 용도로도 무료이다

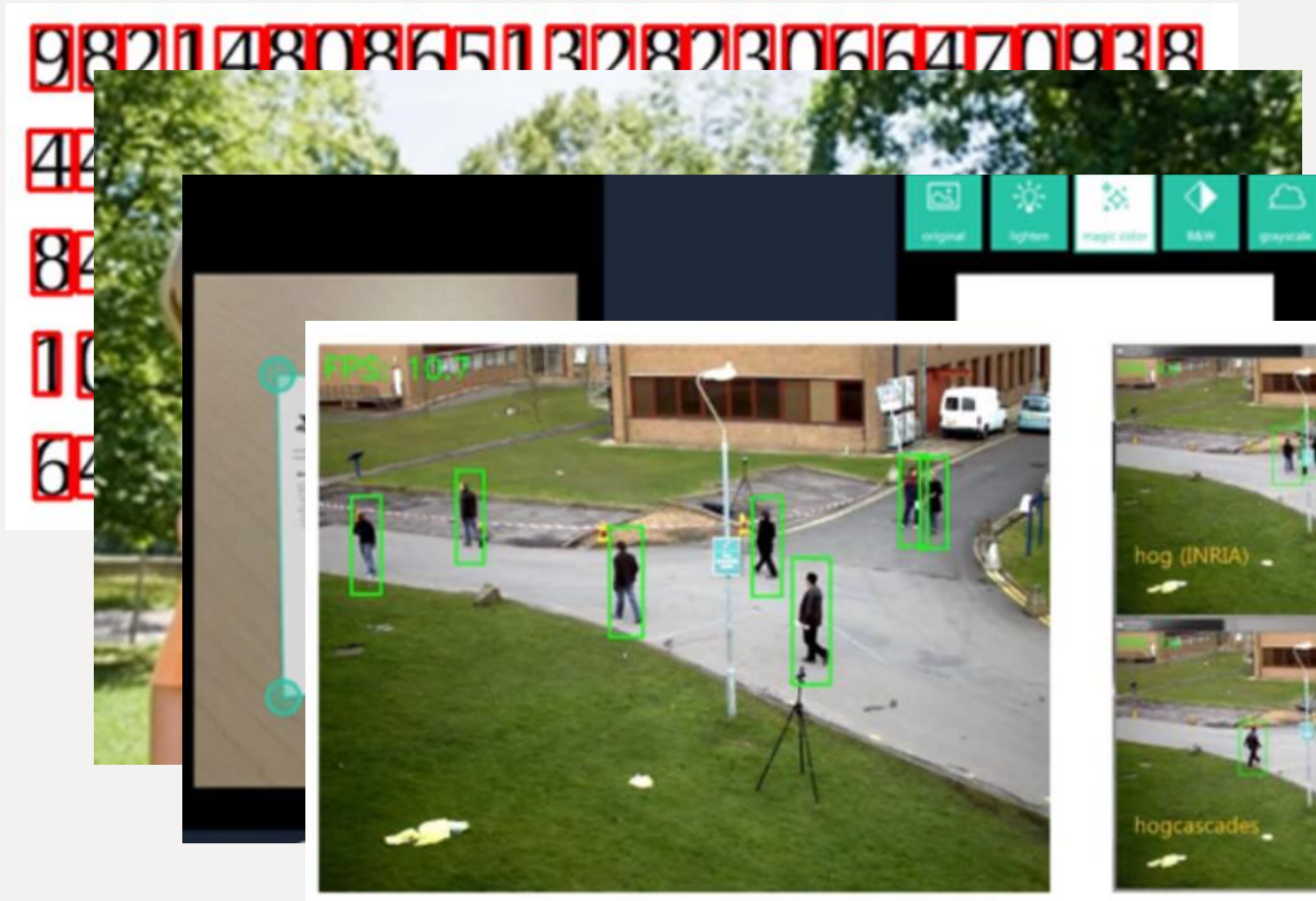
OpenCV는 C / C++ 프로그래밍 언어로 개발 되었으며 Python , Java , matlab / OCTAVE 에 바인딩 되어 프로그래머에게 개발 환경을 지원한다

(ex : OpenCV / C++ 사용하여 영상개발 등)

Low level , High level API 지원 등

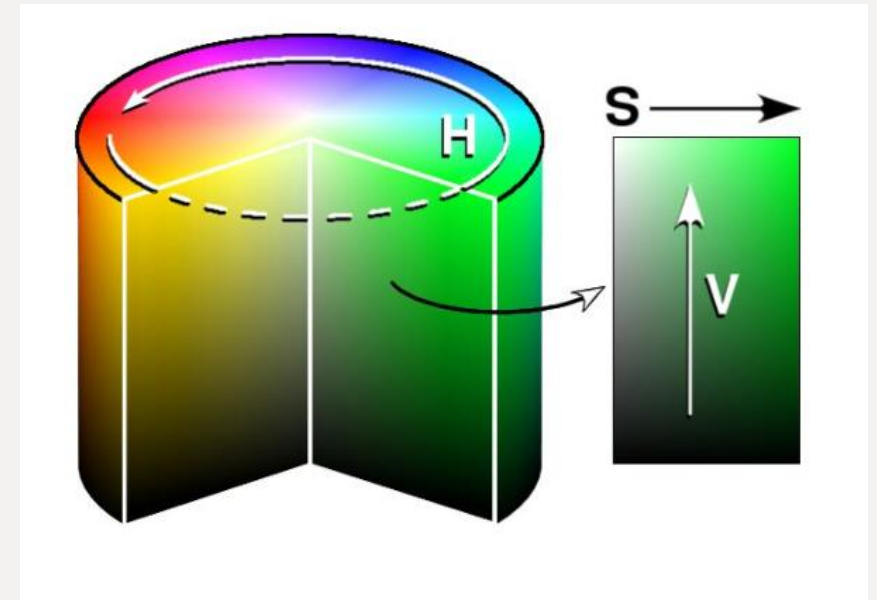
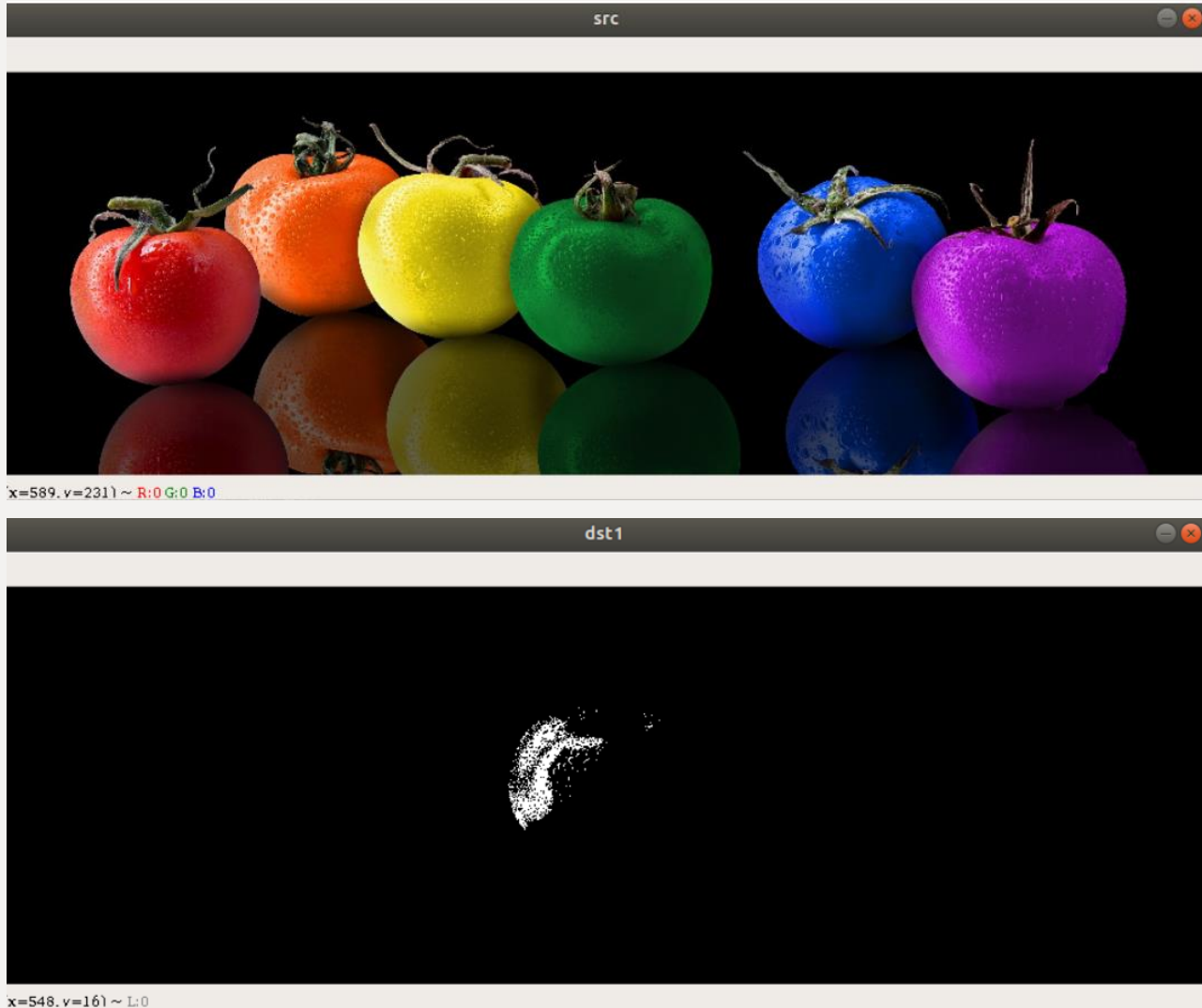


2 그래서 뭘 할 수 있나요?



3 뭘 공부했나요?

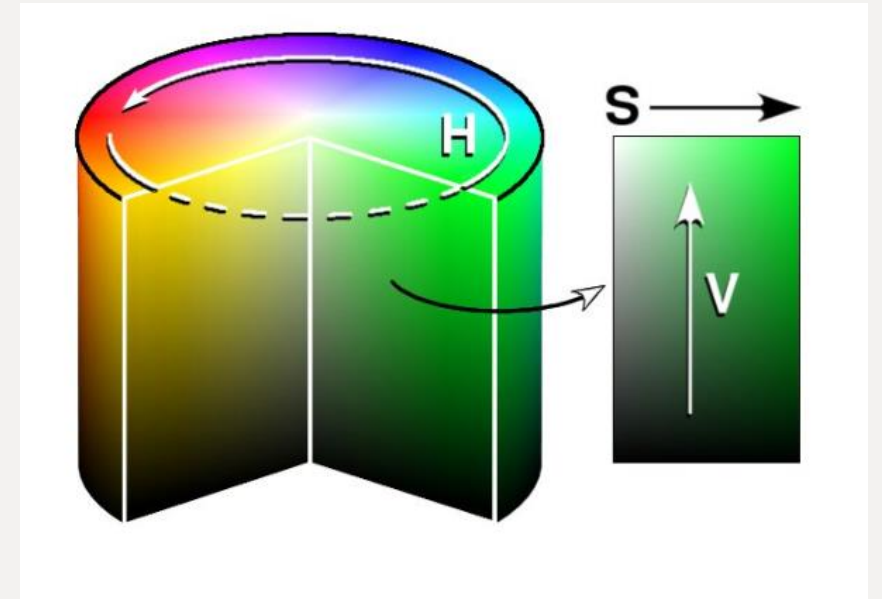
3.1 주제 선정



HSV 컬러 스페이스

3 뭘 공부했나요?

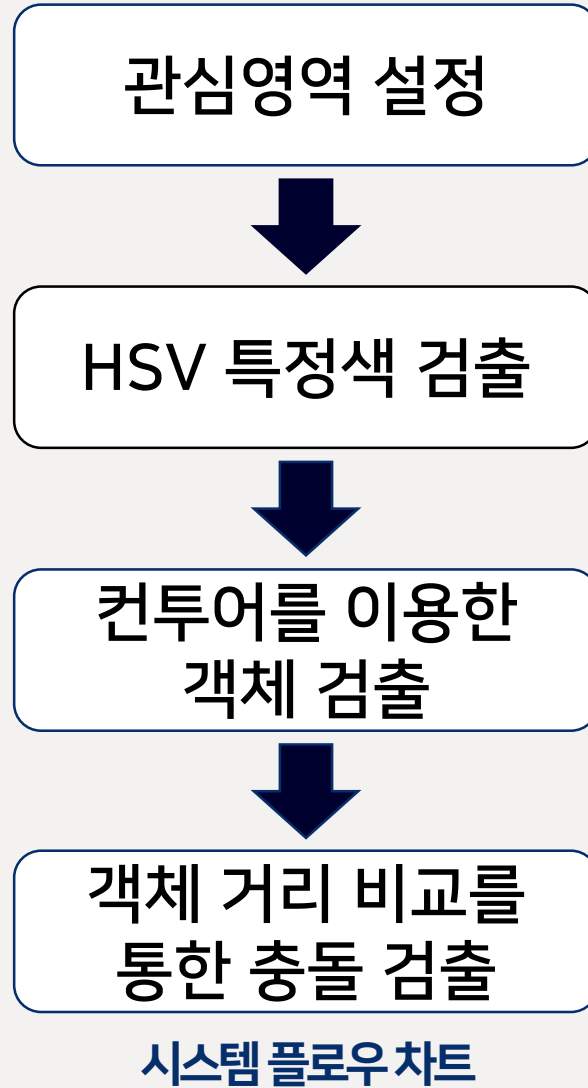
3.1 주제 선정



HSV 컬러 스페이스

3 뭘 공부했나요?

3.2 시스템 설계 및 개발



3 뭘 공부했나요?

3.2 시스템 설계 및 개발

관심영역 설정



HSV 특정색 검출



컨투어를 이용한
객체 검출



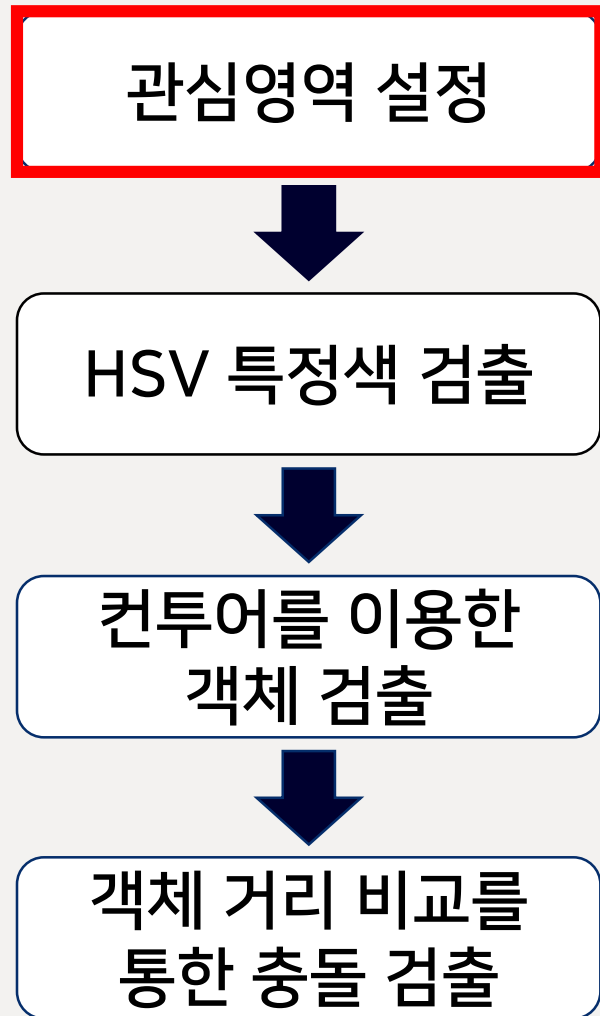
객체 거리 비교를
통한 충돌 검출

시스템 플로우차트

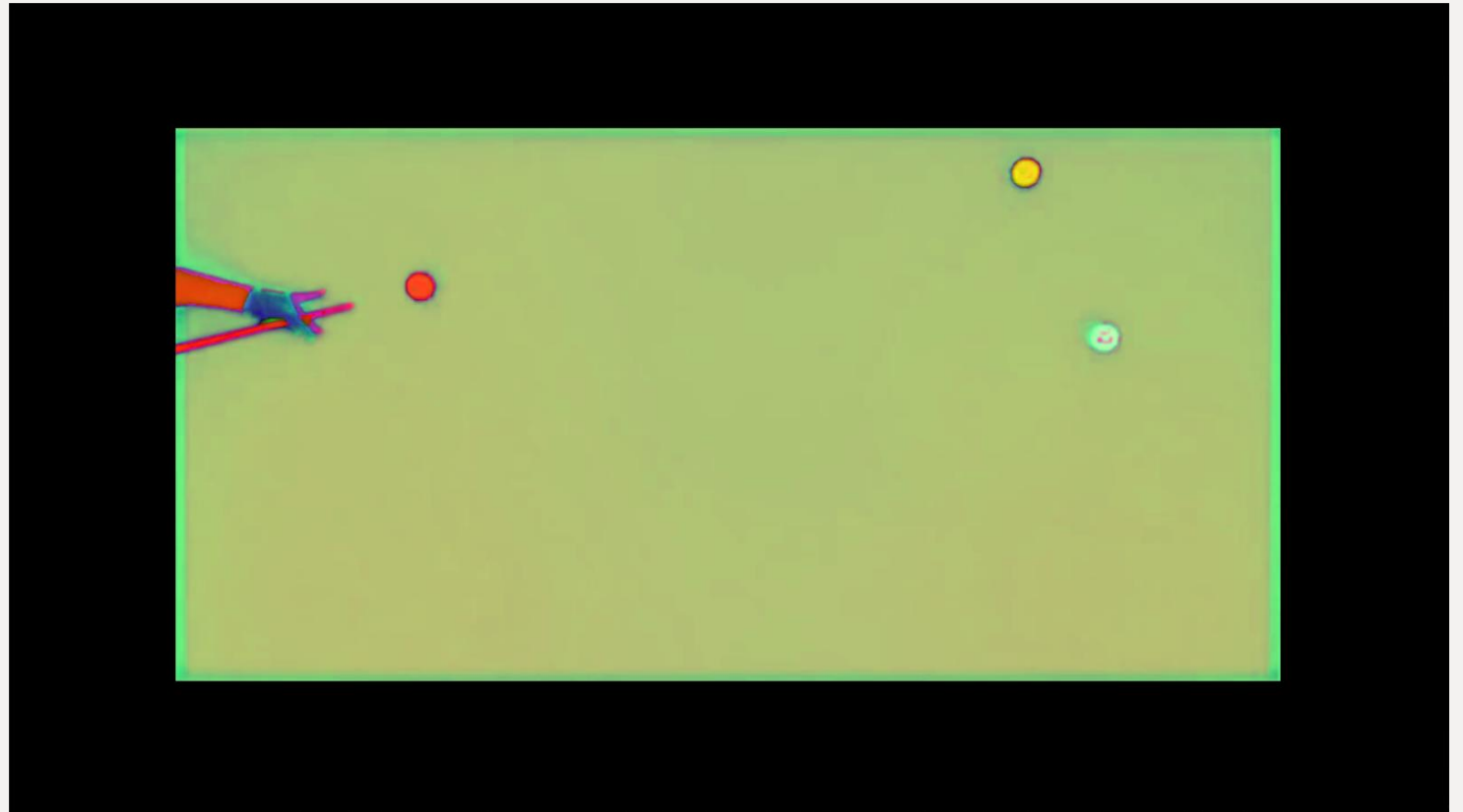


3 뭘 공부했나요?

3.2 시스템 설계 및 개발

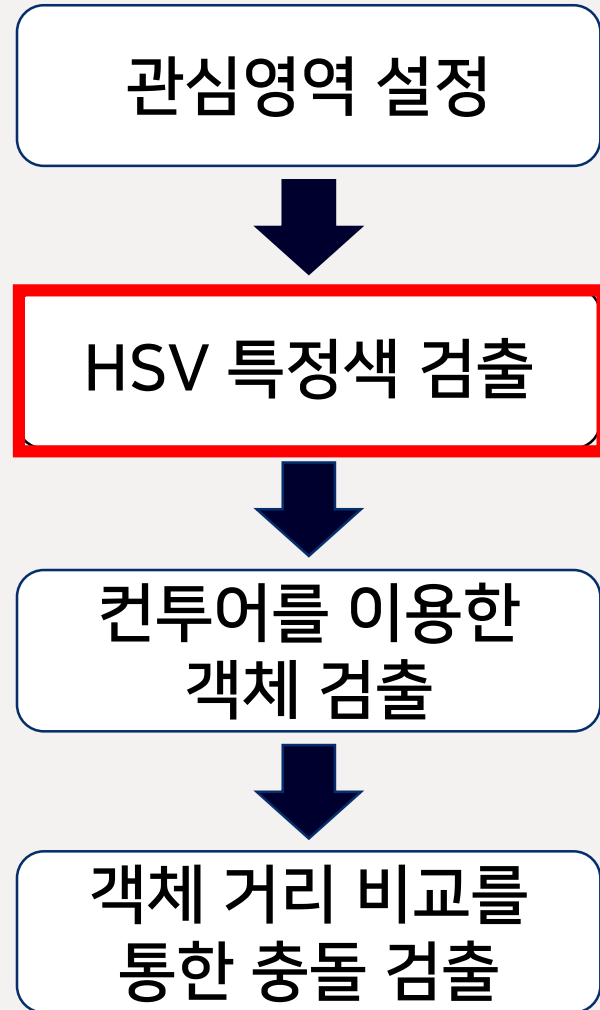


시스템 플로우차트



3 뭘 공부했나요?

3.2 시스템 설계 및 개발



시스템 플로우차트



3 뭘 공부했나요?

3.2 시스템



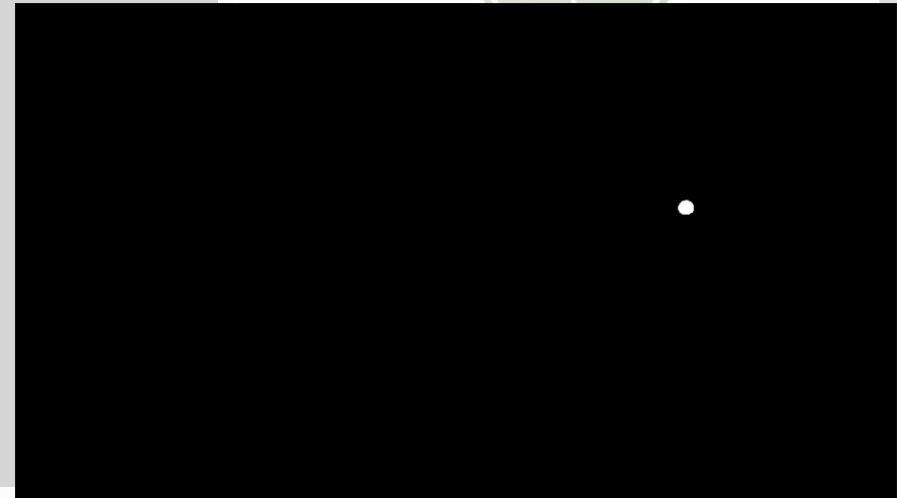
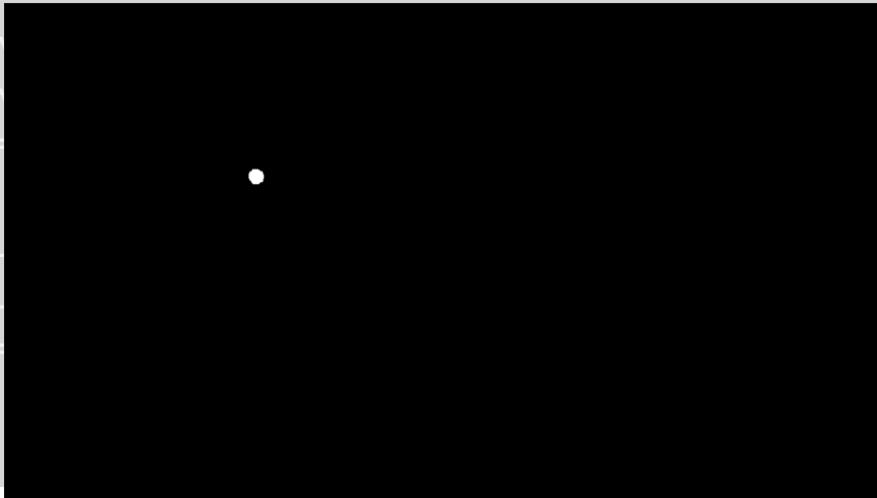
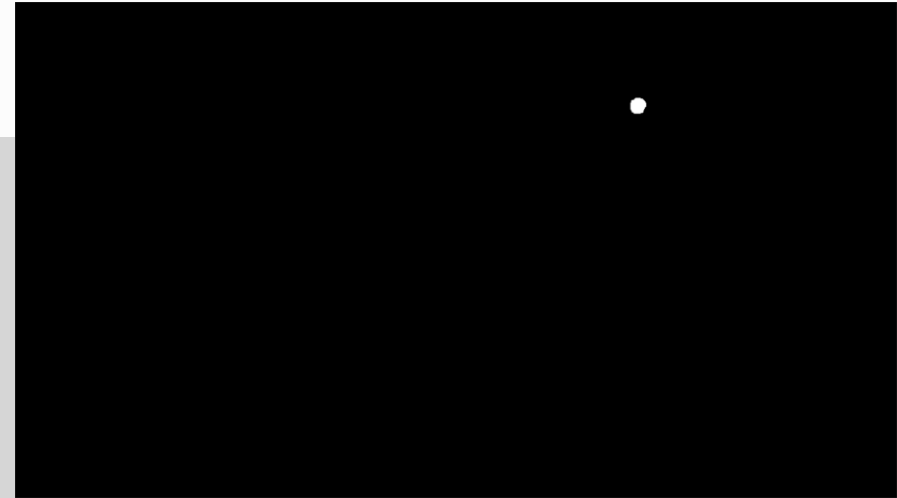
blueLo
blueUp

redLow
redUp

pts = deque(maxlen =64)

yellow
yellow
pts2 =

whiteL
whiteL
pts3 =



300 360
0.8333 1

3 뭘 공부했나요?

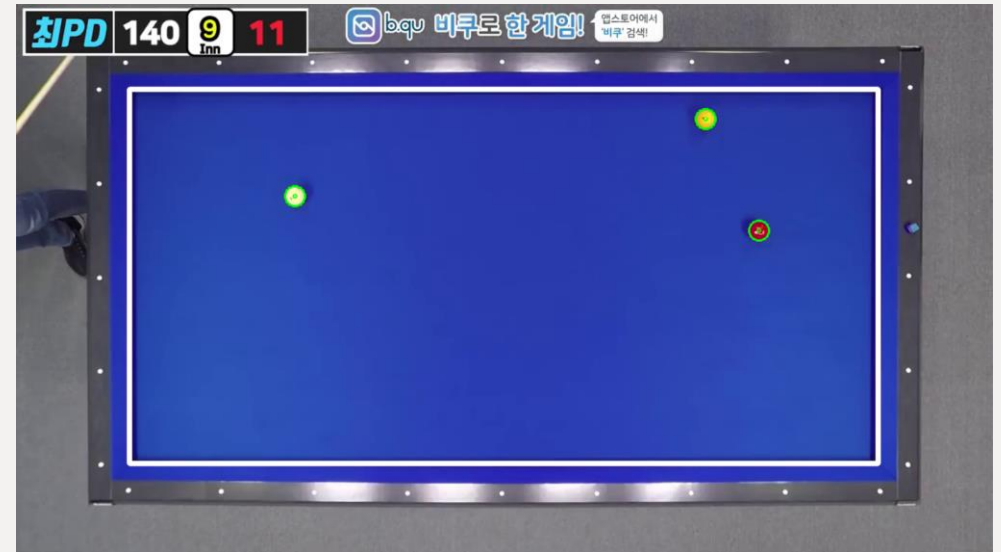
3.2 시스템 설계 및 개발

```
#가장 바깥쪽 컨투어에  
#대해 꼭지점 좌표만 반환  
contours, hierarchy = cv2.findContours(mask.copy(),  
cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)  
    contour = contours[0]  
#모멘트 계산  
mmt = cv2.moments(contour)  
# m10/m00 m01/m00 중심점 계산  
x = int(mmt['m10'] / mmt['m00'])  
y = int(mmt['m01'] / mmt['m00'])
```

```
#빨간공  
if idx == '0':  
    rPoint = (x, y)  
    print("red[x: %d, y: %d]" % (x, y))  
    cv2.circle(img, (x, y), 3, (0, 255, 0), -1)  
    cv2.circle(img, (x, y), 13, (0, 255, 0), 2)
```

동안 중놀 검출

시스템 플로우차트



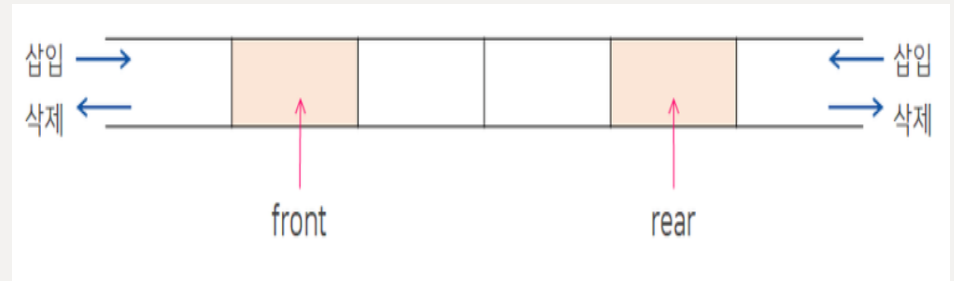
3 뭘 공부했나요?

3.2 시스템 설계 및 개발

#빨간공

```
if idx == '0':  
    rPoint = (x, y)  
    print("red[x: %d, y: %d]" % (x, y))  
    cv2.circle(img, (x, y), 3, (0, 255, 0), -1)  
    cv2.circle(img, (x, y), 13, (0, 255, 0), 2)  
  
pts.appendleft(rPoint)  
for i in range(1, len(pts)):  
    if pts[i-1] is None or pts[i] is None:  
        break  
    cv2.line(img, pts[i-1], pts[i], (0,0,255), 2)
```

Dequeu 자료구조를 이용한
객체 동선 파악



3 뭘 공부했나요?

3.2 시스템 설계 및 개발

```
if dis(rPoint, wPoint) < 30:
    cv2.putText(img, "aa", rPoint,
cv2.FONT_HERSHEY_PLAIN, 2, (0,0,0))

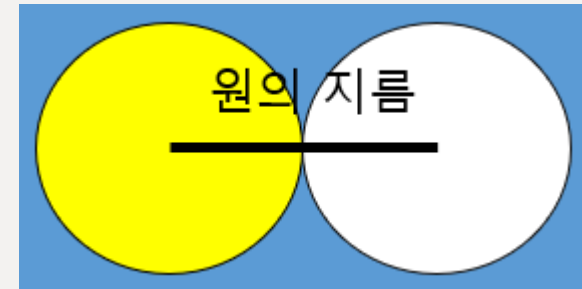
    if dis(rPoint, yPoint) < 30:
        cv2.putText(img, "aa", rPoint,
cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 0))

    if dis(wPoint, yPoint) < 30:
        cv2.putText(img, "aa", rPoint,
cv2.FONT_HERSHEY_PLAIN, 2, (0, 0, 0))

    cv2.imshow('mask'+idx, mask)

def dis(p1, p2):
    return math.sqrt((p2[0]-p1[0])**2 + (p2[1]-
p1[1])**2)
```

좌표값을 이용한 충돌 감지



```
yellow[x: 903, y: 150]
white[x: 365, y: 251]
red[x: 973, y: 296]
yellow[x: 903, y: 150]
white[x: 365, y: 251]
red[x: 973, y: 296]
```

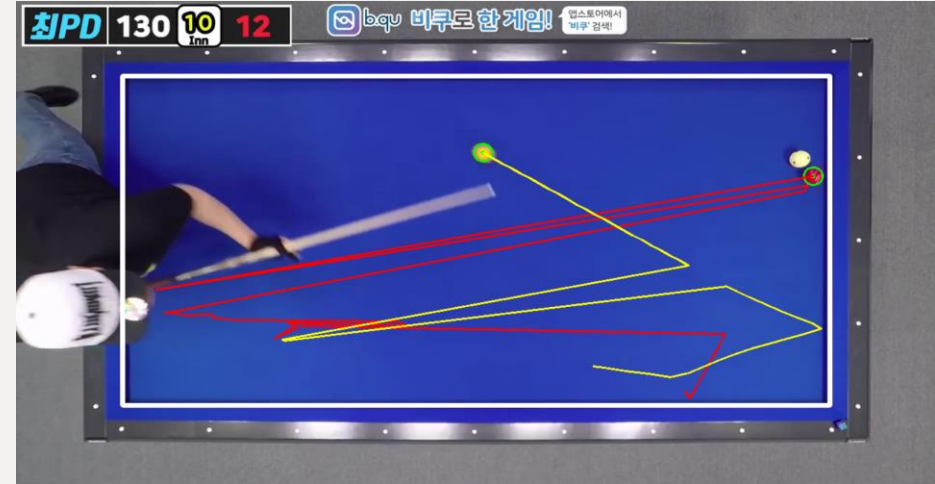
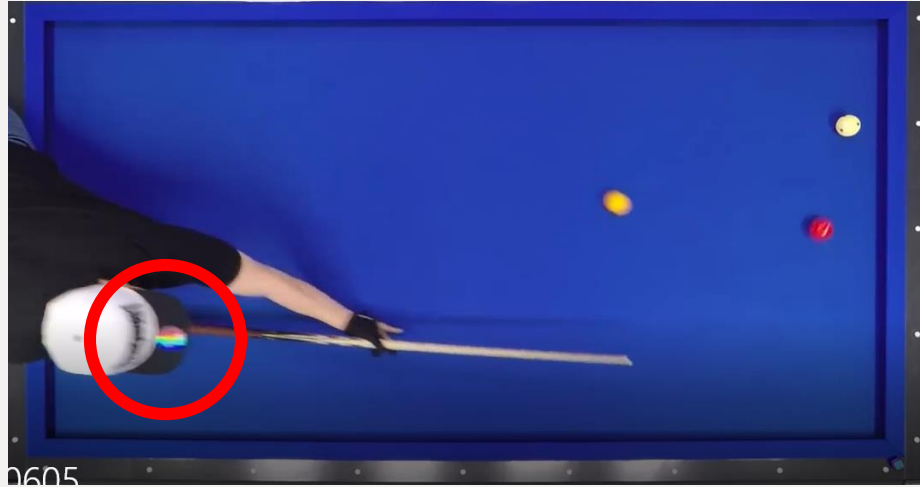

3 뭘 공부했나요?

3.2 시연영상



3 뭘 공부했나요?

3.2 문제점



1. 공과 동일한 색이 카메라에 잡힐 경우 공 탐지가 어려움
2. 연산량으로 인한 동영상 속도 저하

Slug's Power-point template

THANK YOU

DON'T HESITATE TO DOWNLOAD WHENEVER YOU NEED PPTs