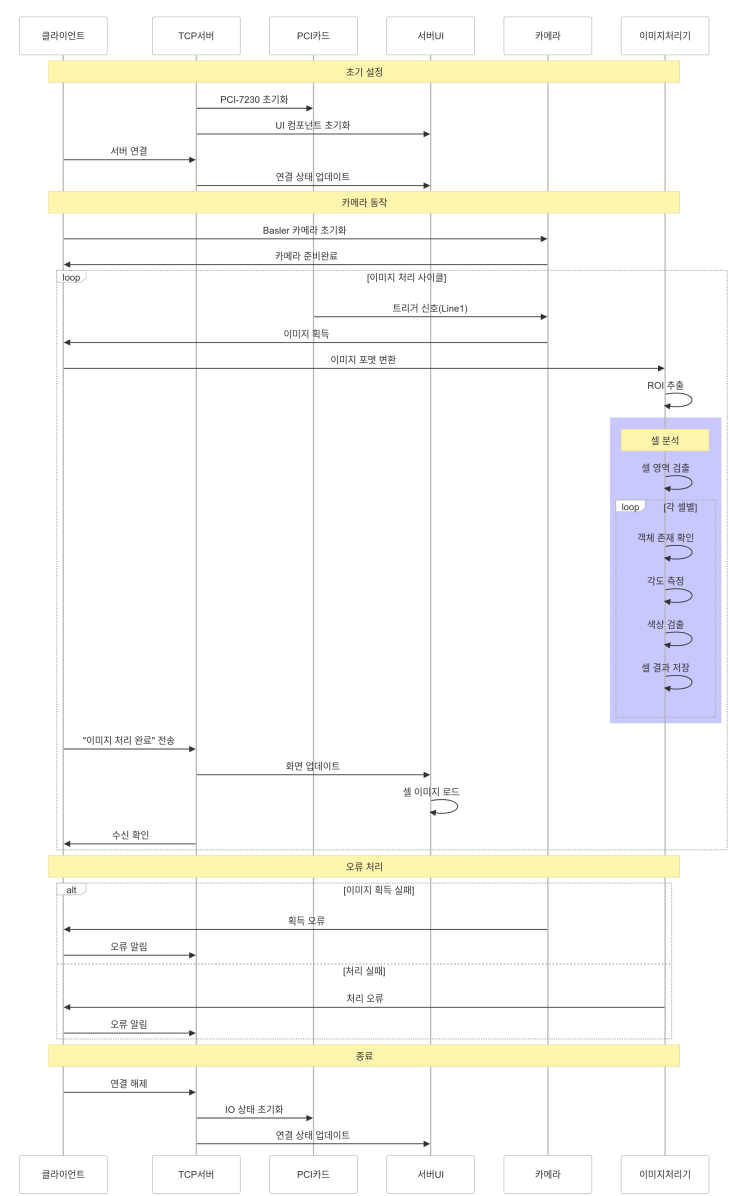


OpenCV 객체 각도 측정 알고리즘 구현

개요



- 서버(C#), 클라이언트(c++) 사이 TCP 통신으로 이미지 그랩 및 처리 진행
- 트리거 신호를 통한 하드웨어 그랩으로 이미지 취득
- 취득한 이미지를 전처리를 통해 ROI 영역 확보
- 확보한 ROI 영역에 대하여 객체 존재 여부 및 존재시 각도 측정

GitHub Repository

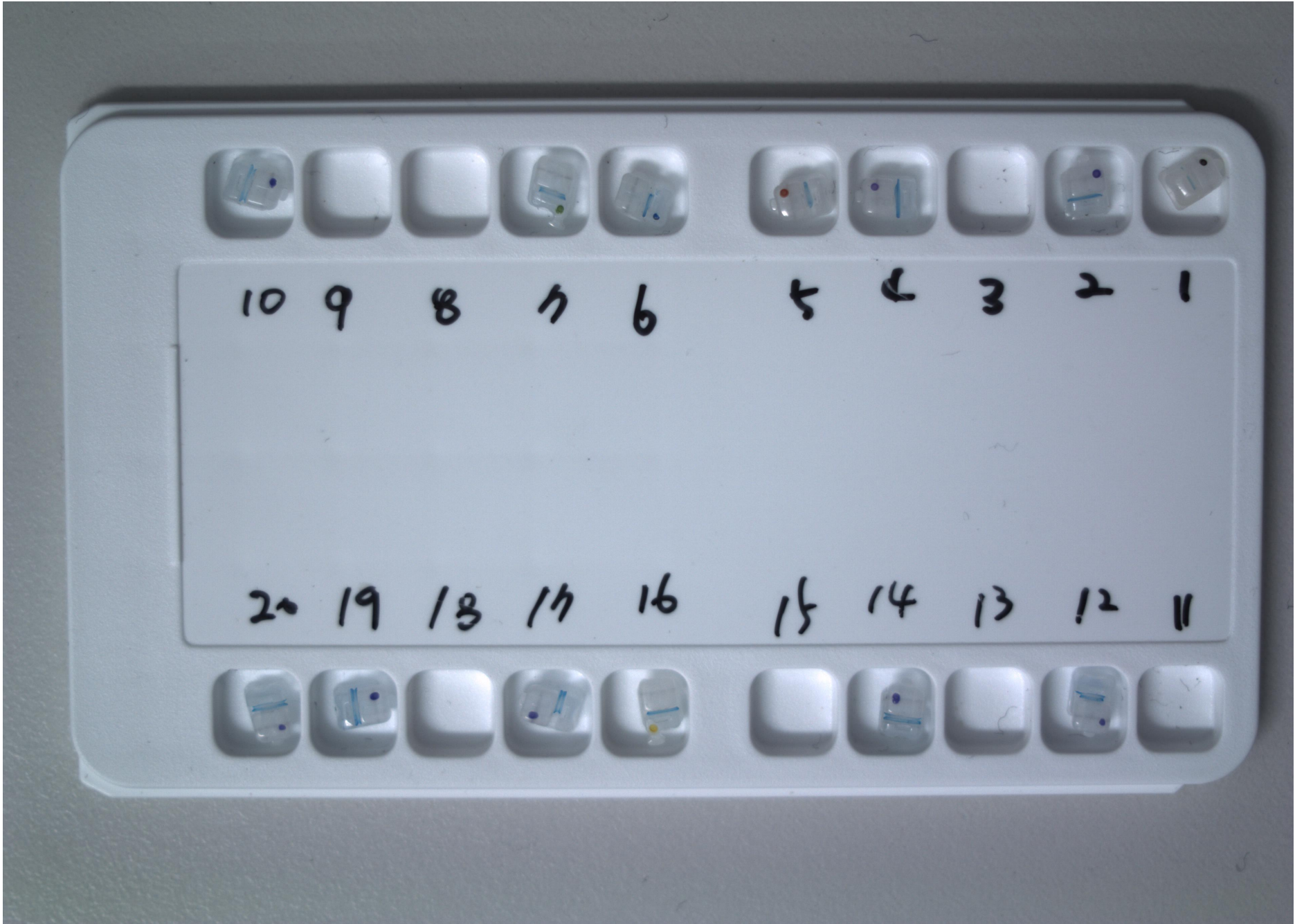
<https://github.com/yoyoyoun18/imageporcessing-for-angle>

각도 검증 시트

<https://docs.google.com/spreadsheets/d/1VM-UePwieXb62UKuNrot9PPdrdKh6N939IfPv2V0Pnc/edit?gid=1367327443#gid=1367327443>

당시 진행했었던 각도측정 검증 알고리즘을 스프레드시트로 구현한것으로 그림판으로 목표 픽셀값을 추출하여 대입 후 프로그램으로 출력된 값과, 스프레드 시트의 값을 비교하여 정확도를 비교하는 시트입니다.

검사 대상 객체



20개의 칸이 있고 각 칸마다 물체가 담겨 있음

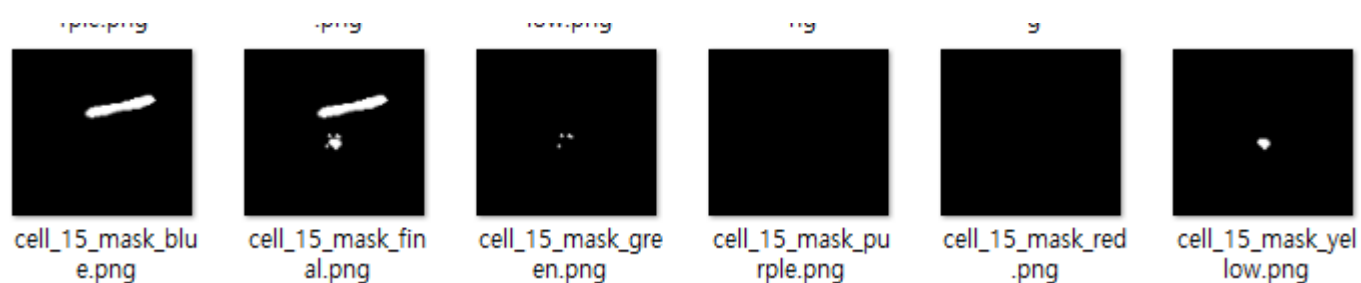
의료 용품 케이스로 케이스는 20개의 칸을 가지고 있고 해당 칸에 물체가 담겨있는 형태.

검사 목표

1. 각 칸의 물체 존재 여부 측정
2. 물체가 있다면 해당 물체의 마커(점) 색상 측정
3. 해당 물체의 기울어진 정도 측정

마킹 컬러 확인

- HSV 색상 공간을 사용하여 다음 색상들을 검출
 - 파란색 (H: 80-130)
 - 검은색 (V: 0-30)
 - 노란색 (H: 20-30)
 - 빨간색 (H: 0-10 및 170-180)
 - 보라색 (H: 130-150)
 - 초록색 (H: 35-85)
- 각 색상에 대해 마스크를 생성 후 해당 마스크들을 취합해 최종 마스크 생성



blue 검출에서 선이 발견되었고 yellow 검출에서 점이 발견되어 최종 병합본(final) 마스크 이미지를 생성하는 방식

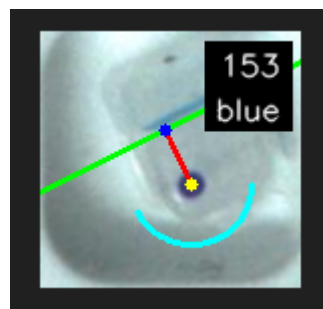
- 셀 영역의 10% 미만 크기를 가진 컨투어 중 가장 큰 영역을 점의 색상으로 판단

객체 유무 확인

- 그레이스케일 변환 후 이진화를 수행
- 외곽선(컨투어) 검출을 수행합니다
- 검출된 컨투어 중 셀 면적의 1% 이상인 영역이 있으면 객체가 있다고 판단



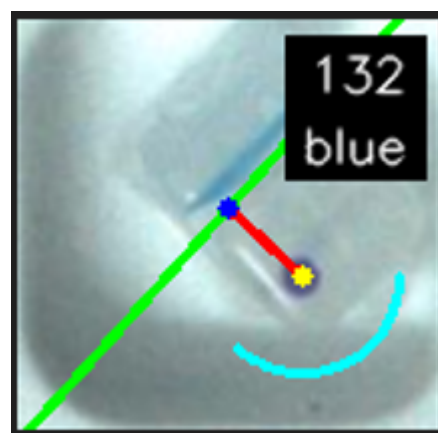
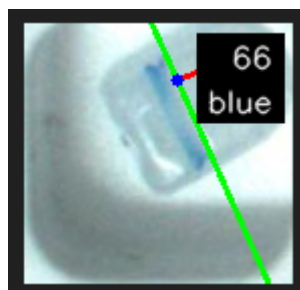
객체가 없을 경우



객체가 있는 경우 각도와 마커색이 표기됨

객체 각도 측정

- 가장 큰 컨투어(선)와 두 번째로 큰 컨투어(점)를 찾습니다
- 점 부분을 상단으로 정의
- 점에서 선까지의 수선의 발을 계산합니다
 - 점 부분을 객체의 상단으로 정의했기 때문에 점에서 선으로의 최단거리 선을 그어 해당 선의 기울기로 각도 측정(12시 방향을 0도로 계산)
- 수선의 발과 점을 이은 선의 각도를 계산합니다



- 그래픽으로 보이는선은 약간 부정확해도 객체 각도 자체는 맞게 측정되는 것으로 확인

택타임 최적화

기존 택타임

Image grab time: 815 ms ⇒ 바슬러 카메라 이미지 그랩 시간
Image conversion time: 771 ms ⇒ 그랩된 이미지를 mat 객체로 변환하는 시간
ROI extraction time: 2799 ms ⇒ roi 영역 추출(흰 판 추적 ⇒ 20개의 셀 추적)

바슬러 카메라 이미지 그랩 시간

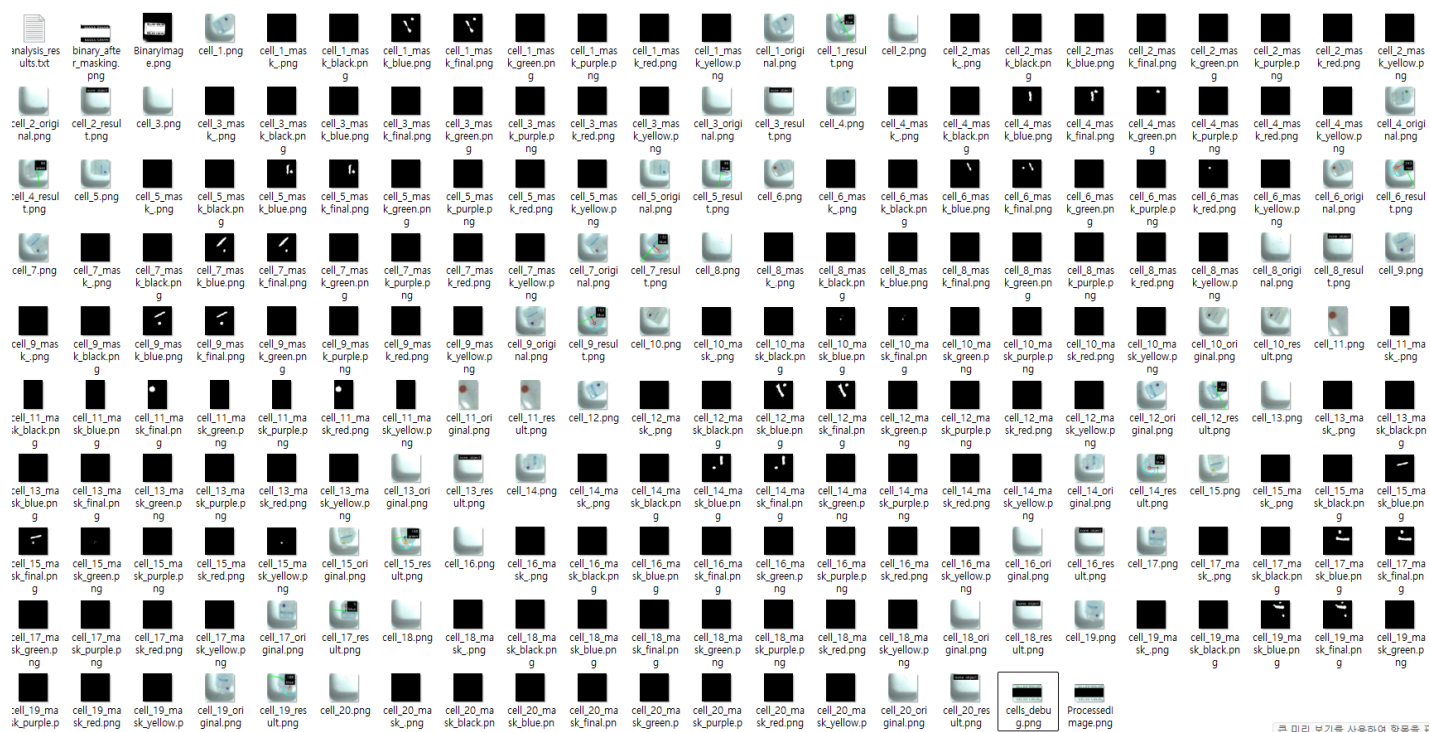
- exposure 타임이 클수록 그랩 시간이 오래걸린다 하여 48000 ⇒ 4800 으로 줄여봤으나 여전히 800ms 내외로 측정됨
- 소프트웨어 그랩 ⇒ 24v 트리거를 주는 하드웨어 그랩으로 변환하였더니 10ms 로 줄어듦

그랩된 이미지를 mat 객체로 변환하는 시간

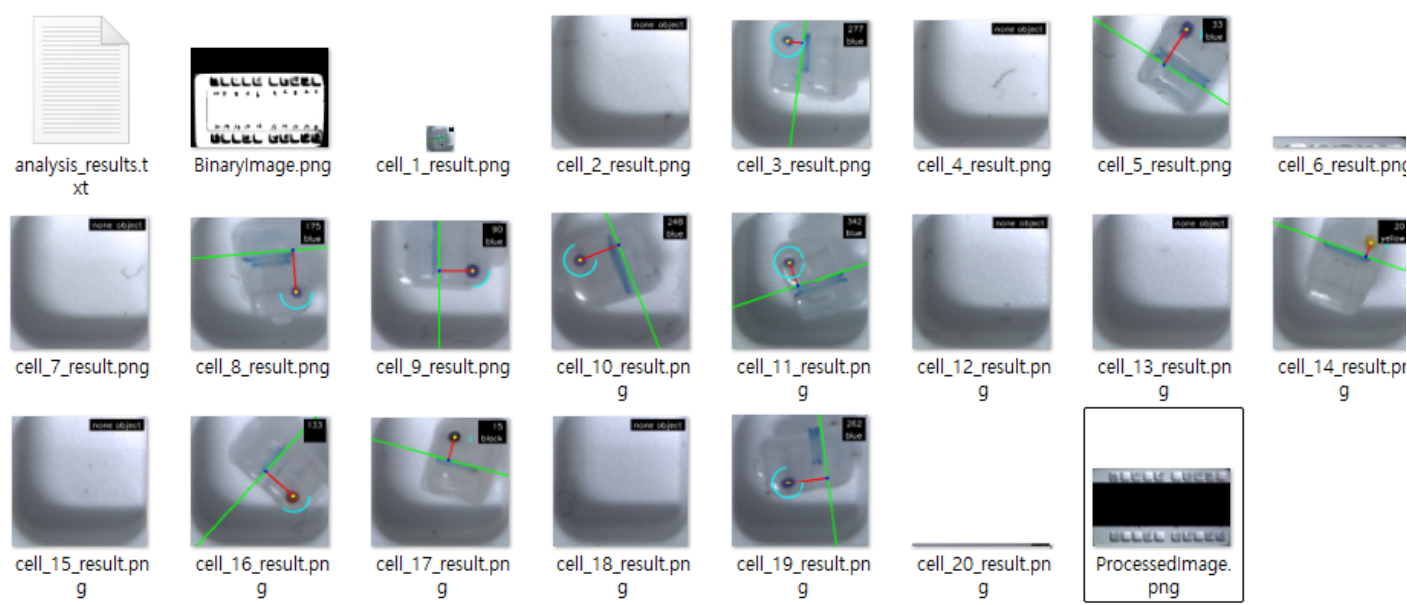
- 원래 디버깅용으로 최초 바슬러 카메라가 그랩한 이미지 버퍼를 png로 저장했다가 그 png를 다시 mat 객체로 변환함
- png로 저장하는 과정을 제외하고 카메라 이미지 버퍼를 바로 mat 객체로 변환하였더니 10ms로 줄어듦

결과: 771ms ⇒ 10ms로 감소

roi 영역 추출(흰 판 추적, 20개의 셀 추적)



기존 1번의 공정당 저장 이미지



필수적인 것만 남긴 상태

결과: 2799ms ⇒ 2049ms

총 택타임 감소

Total processing time: 4459631 microseconds (4459.631 ms)

⇒ Total processing time: 2223956 microseconds (2223.956 ms)

텍타임 약 50퍼센트 감소