

# 커버글래스 이미지 전처리

김초명

2022-01-03 연구공유

상세 설명 및 코드

[https://github.com/MonicaKim89/Defect\\_detection](https://github.com/MonicaKim89/Defect_detection)

- 사용 라이브러리  
opencv, matplotlib, numpy, scikit image 외
- 사용데이터  
촬영 된 데이터 중 '\_02.bmp'으로 끝나는 이미지만 사용
- 전처리방향
  - 1) 컨투어 추출
  - 2) 커버글래스, 스크린 영역 컨투어 추출
  - 3) 스크린 영역만 추출(trimming)
  - 4) 커버글래스 영역 외 스크린영역까지 흰색 마스크처리
  - 5) 흰색 마스크 투명화

# 사용 데이터



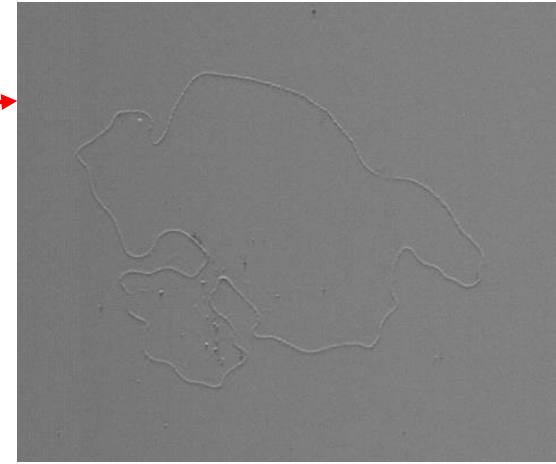
B2\_01.bmp



B2\_02.bmp



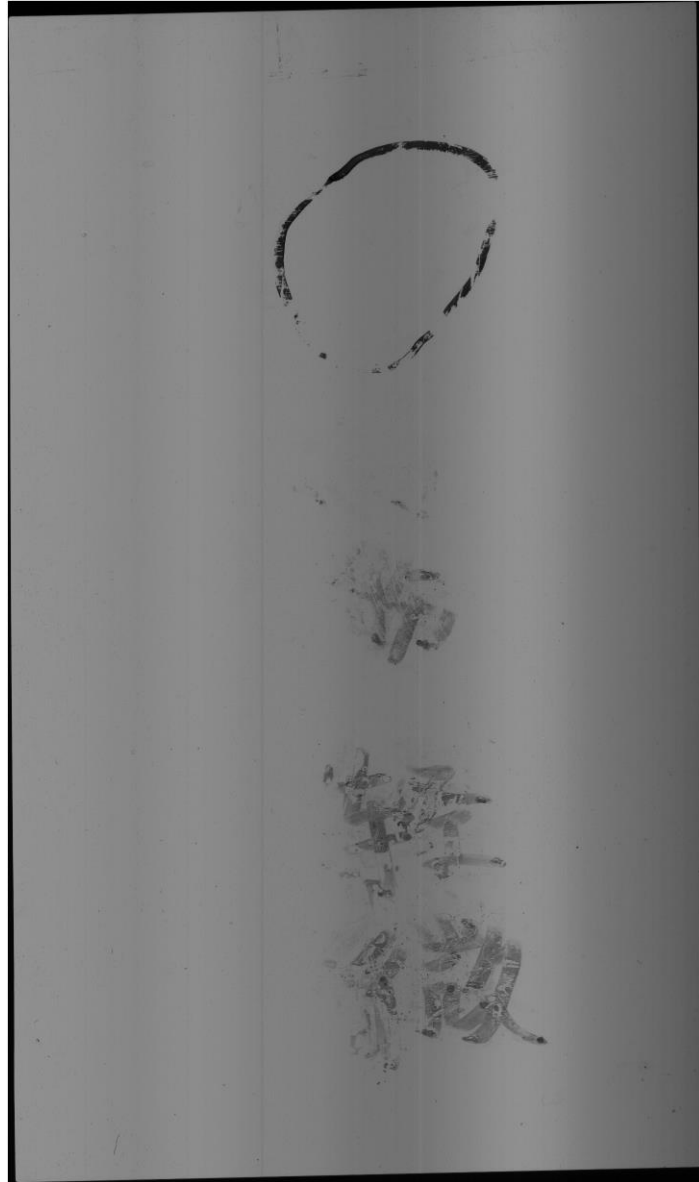
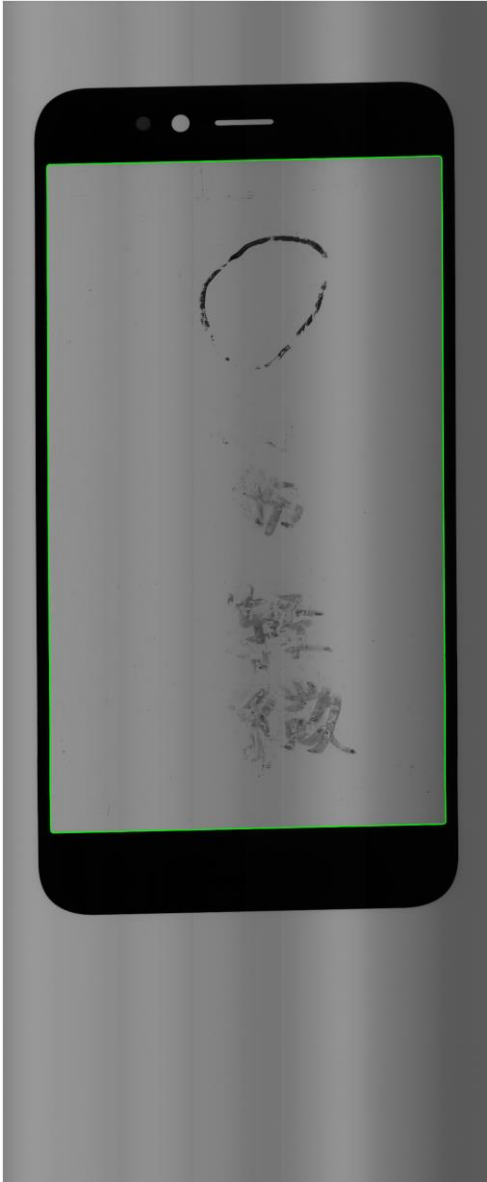
B2\_03.bmp



이미지 확대 후 defect 육안확인가능  
\_03.bmp가 더 정확해보이지만 **전처리 자동화 문제발생**(임계치 처리 어려움)

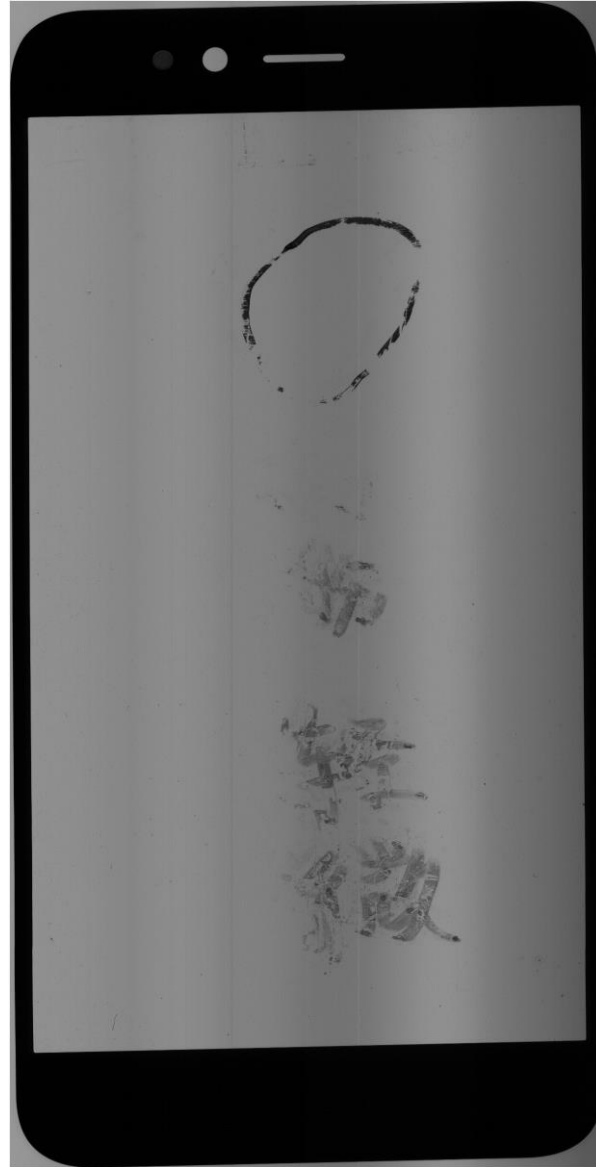
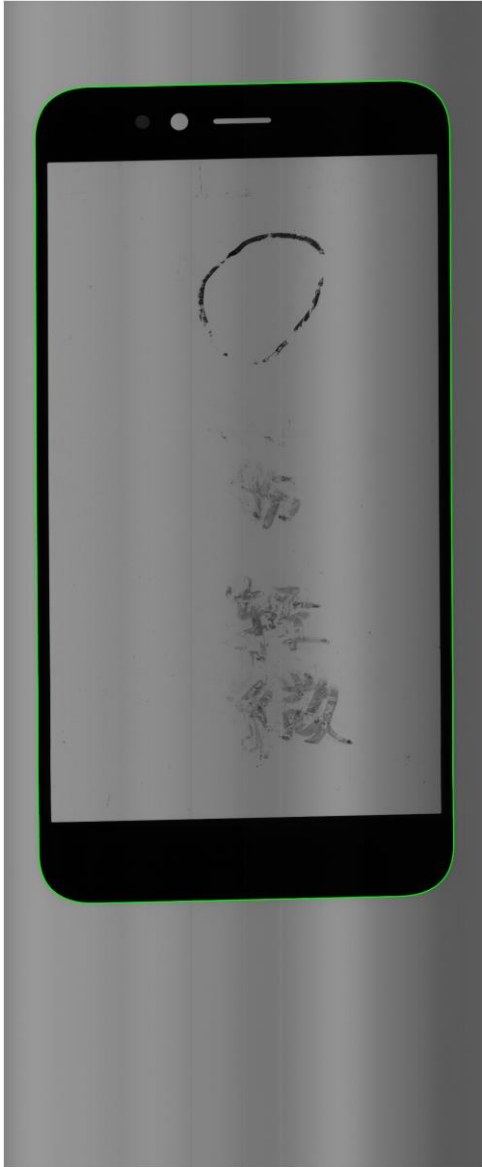
- ➔ 02.bmp으로 defect 위치 확인 후 03.bmp에서 crop할 수도 있음
- ➔ 조건 02.bmp과 03.bmp이 **정확히 같은 위치와 같은 사이즈**로 촬영 되어야함

# 전처리 - 스크린 추출



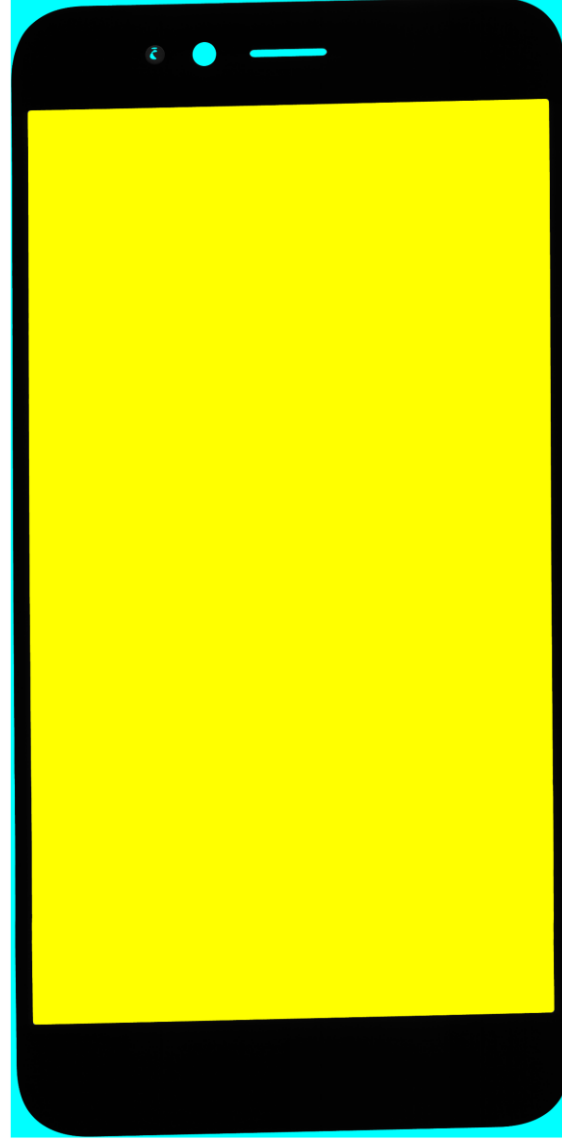
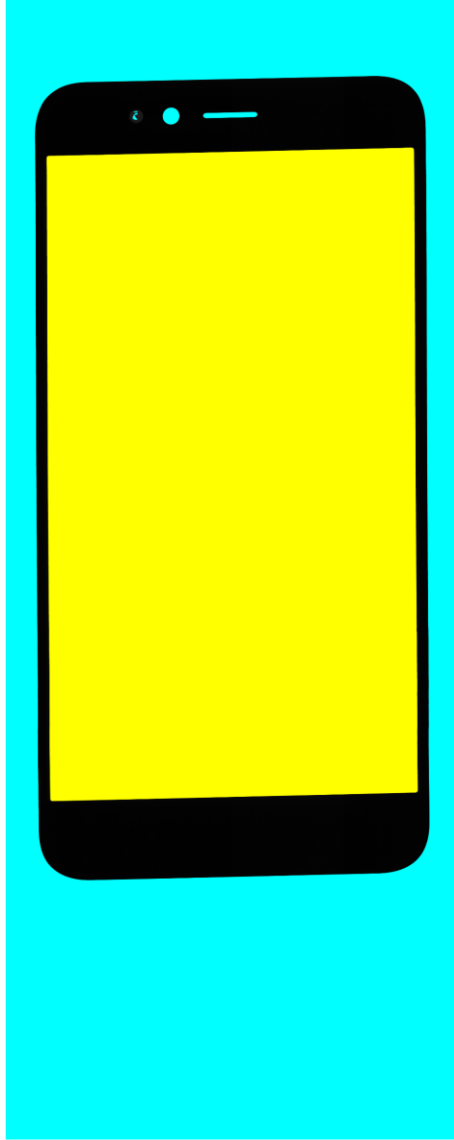
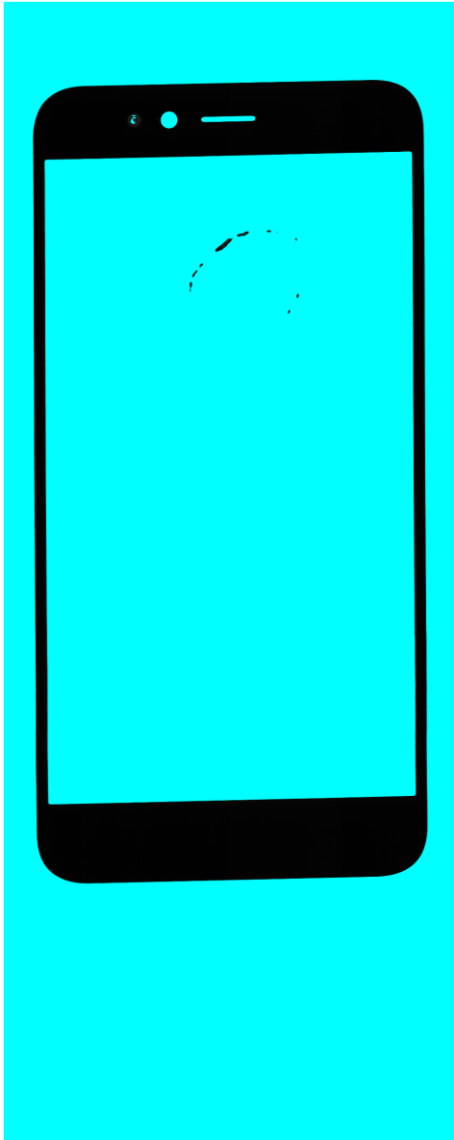
스크린의 컨투어를 확인한 후,  
배경과 커버글래스 엣지 컨투어를 기준으로 제  
거함  
원본이미지에서 제거한 것으로 이미지정보 손실  
없을 것으로 예상

# 전처리 - 커버글래스 edge추출



커버글래스의 컨투어를 확인한 후,  
배경을 컨투어를 기준으로 제거함  
원본이미지에서 제거한 것으로 이미지정보 손실  
없을 것으로 예상

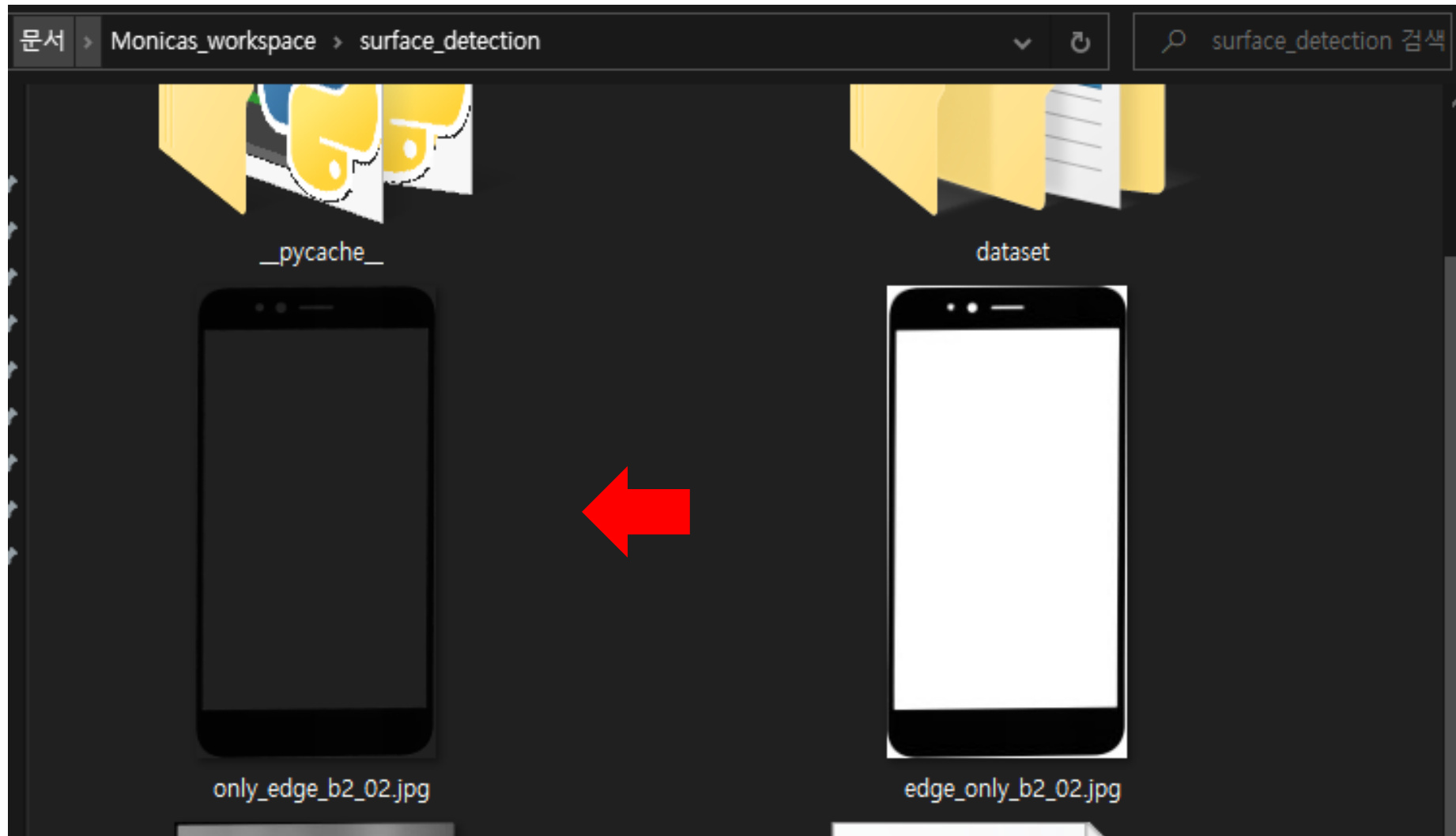
# 전처리 - 커버글래스 마스킹



커버글래스의 컨투어를 확인한 후,  
배경을 컨투어를 기준으로 마스킹처리  
스크린 컨투어를 기준으로 마스킹처리  
이 후 커버글래스 컨투어 기준으로 배경  
제거

((해당이미지의 마스킹은 이해를 위한 색  
상으로, 실제로는 흰색 마스킹 처리))

# 전처리 - 커버글래스 투명화

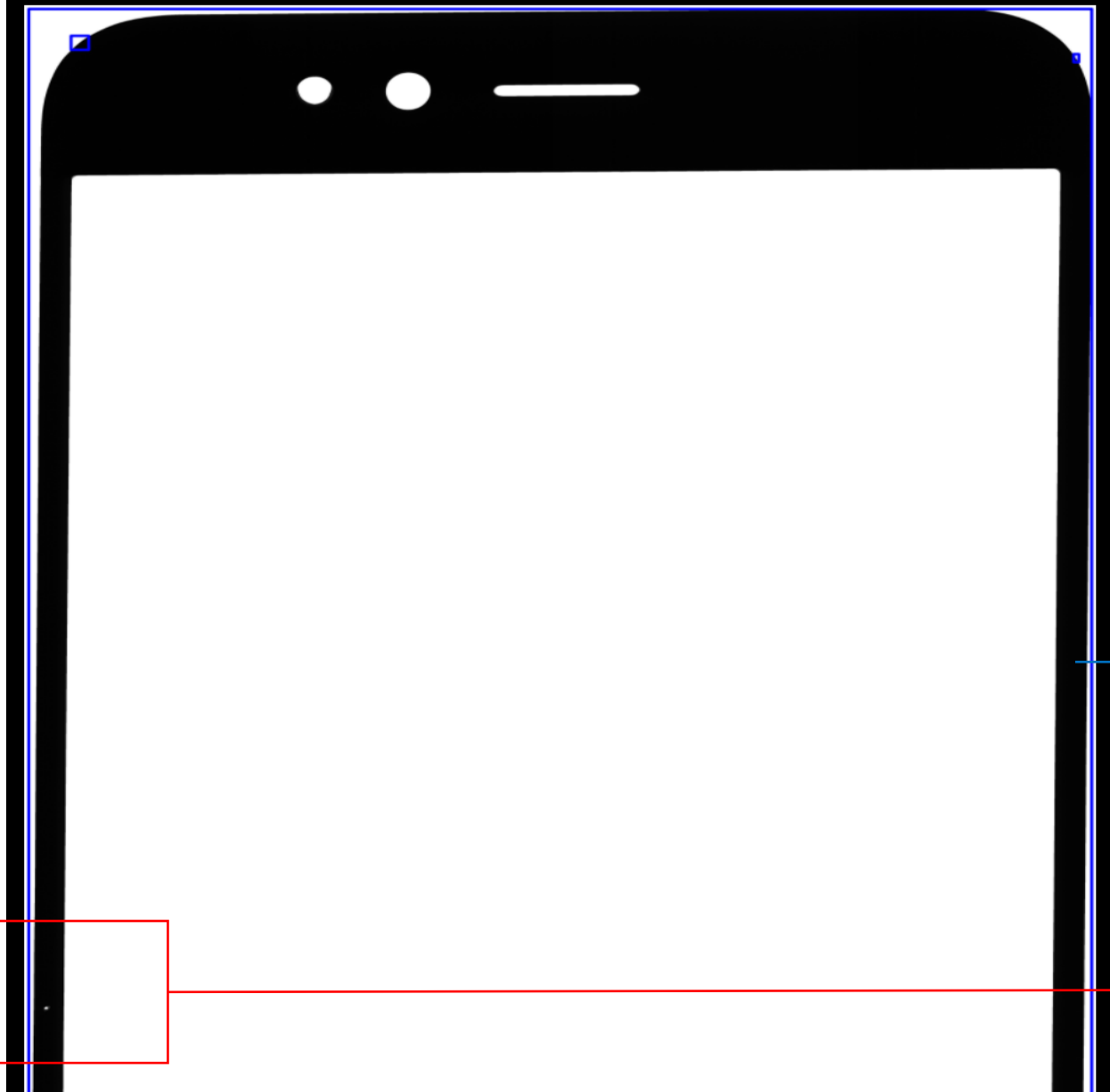


파일탐색기에서 확인한 화면, 배경이 투명해 파일탐색기 배경색상투영

커버글래스 defect detection  
v1



# Normal image- defect image 비교



Threshold 값 조정을 통해 denoising 후 임의로  
정상이미지 만듦

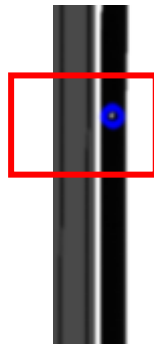
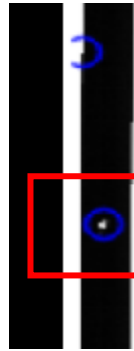
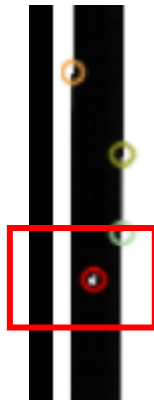
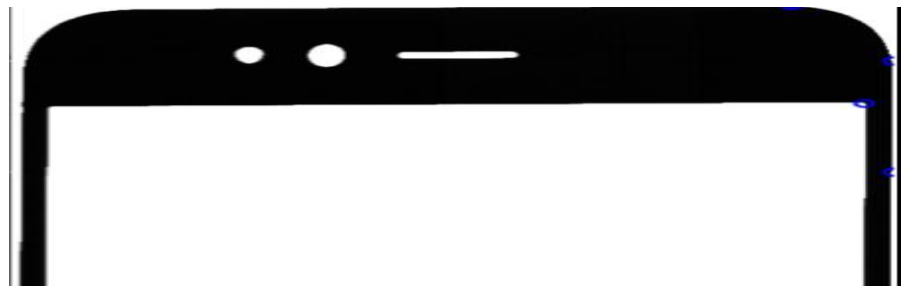
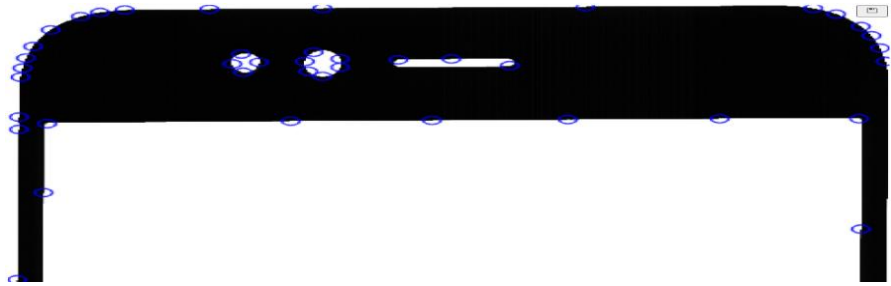
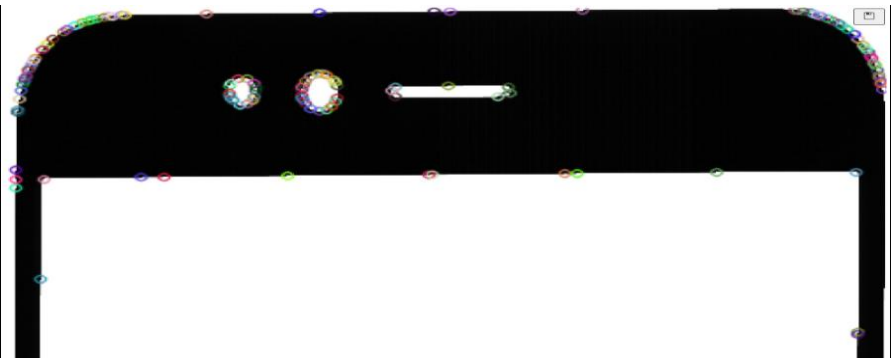
이후 defect 이미지와 비교했지만 **defect 추출  
불가**

→ Threshold를 조정하면서 곡선의 값이 변경된  
것으로 예상됨

변화된 것으로 확인되는 픽셀의  
bounding box 이미지 컨투어의 픽셀이  
변경된 것을 예상할 수 있음

탐지 했어야 할 노이즈 객체

# 특징점 추출



1) GFTT 검출

2) 시와 토마시 검출

3) 해리스 검출

→ Threshold를 조정하면서 곡선의 값이 변경된 것으로 예상됨

탐지했어야 할 노이즈 객체 탐지했으나 모든 곡선과 주변부 픽셀과 비교했을 때 결합 및 변화가 있는 경우 모두를 특징점으로 잡기 때문에 노이즈 detector로는 **활용 불가능**

커버글래스 defect detection  
v2

- 방법론

- sliding window( $n \times n$ )으로 커버글래스 edge 및 스크린 이미지 crop
- opencv를 통한 이미지 개선(edge, screen 따로 진행되어야)
- denoising을 통한 미세노이즈 개선
- 개선된 이미지 segmentation 작업 (시간 및 전문인력소요예상)
- segmentation을 통한 labeling
- classification training 수행