

Final-TERM Project

제품에 대한 영상 처리

2021. 12. 15

2020254002

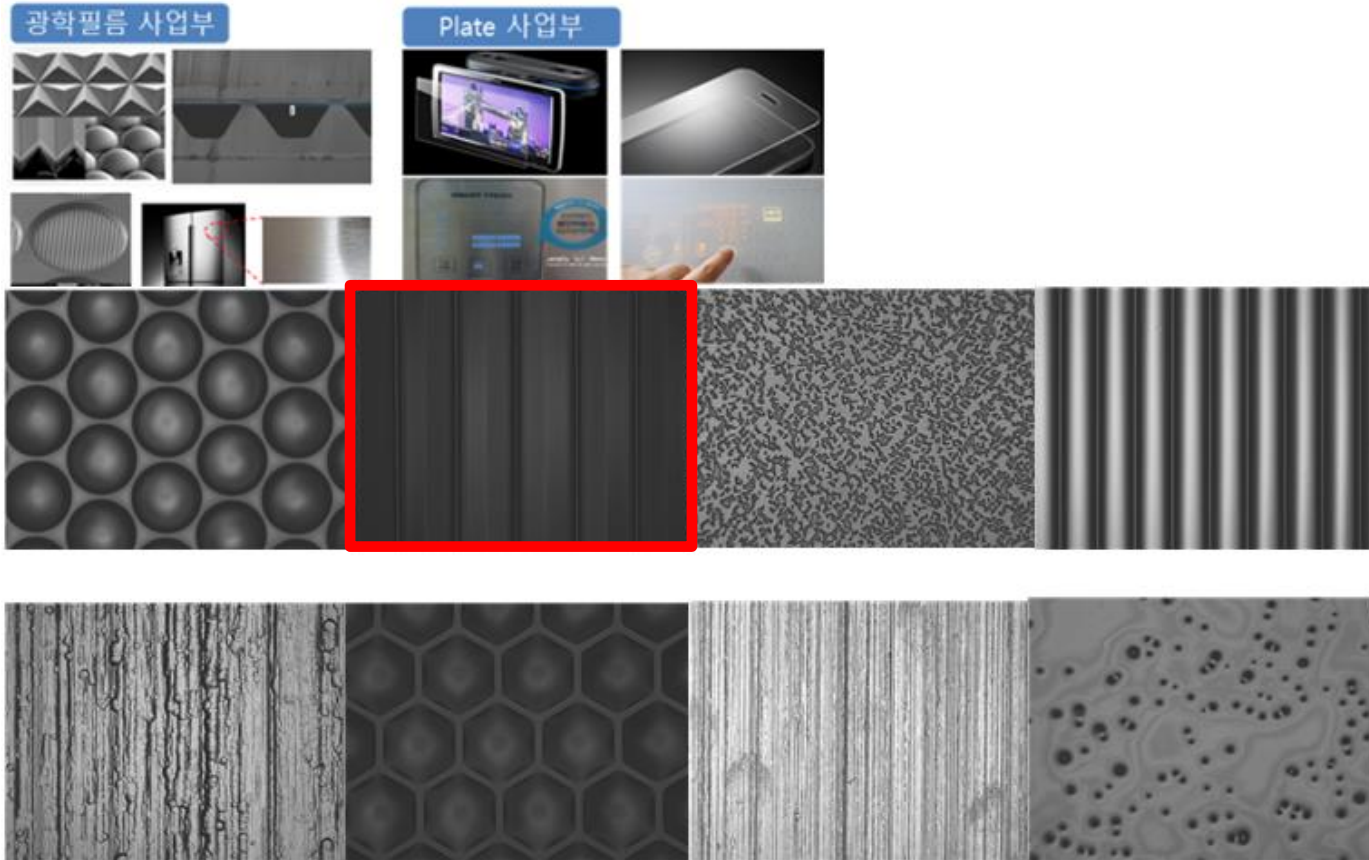
고 정 재

과제 내용

- 중간 프로젝트 때 정한 문제 혹은 새롭게 정한 현업에서의 컴퓨터 비전 문제에 대해서 최종 프로젝트를 진행함
- 14주차 강의까지 배운 내용들을 활용해서 특징 추출, 대응점 탐색, 옵티컬 플로우, 파노라마 스티칭, 카메라 캘리브레이션, 스테레오 매칭 등을 적용
- 12월 15일 강의시간에 각자 5page 내외로 발표
- 발표자료와 소스코드를 e-campus를 통해 제출 (제출 기한 12/15 11:59PM)

영상 처리를 위한 데이터베이스 수집

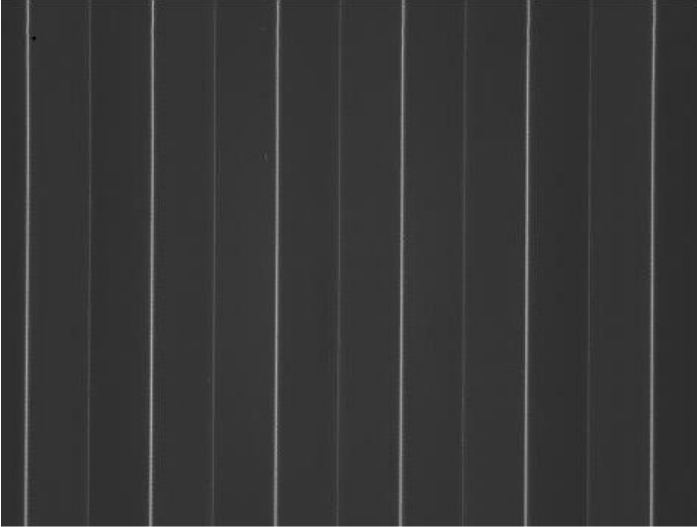
- 당사는 광학필름 및 가전 데코레이션 필름을 개발,제조하는 회사입니다.



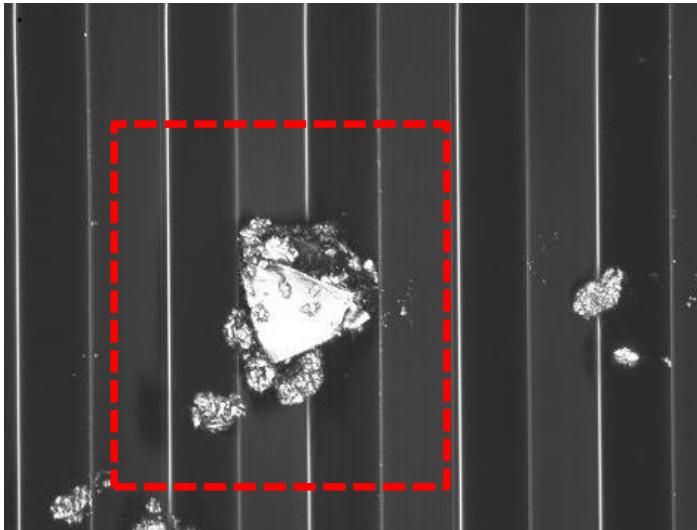
- 위와 같이 다양한 패턴을 가지고 제품을 개발 , 제조하고 있습니다.
- 이 중 광학필름에 한 종류인 Prism패턴을 영상 처리를 했습니다.

추출 영상

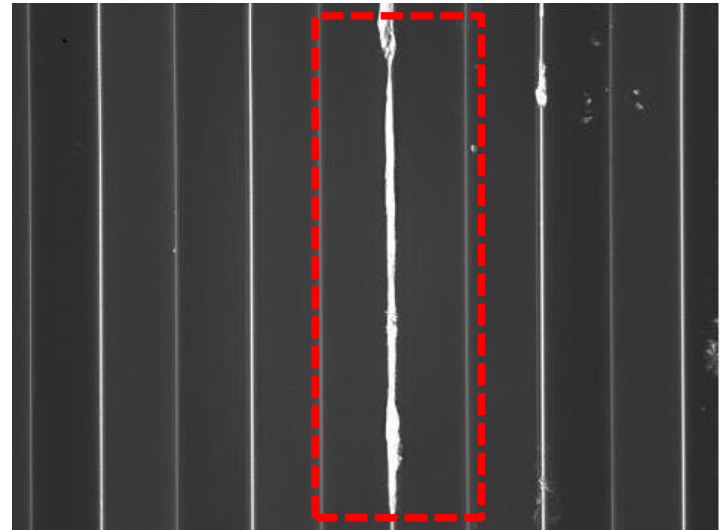
➤ OK 제품



➤ NG 제품



(White spot)



(Scratch)

추출 알고리즘 – Harris corner detector/FAST corner detector

1. OK 제품- code

```
import cv2
import numpy as np

img = cv2.imread('c:/img/OK2.PNG', cv2.IMREAD_COLOR)
corners = cv2.cornerHarris(cv2.cvtColor(img, cv2.COLOR_BGR2GRAY), 2, 3, 0.04)

corners = cv2.dilate(corners, None)

show_img = np.copy(img)
show_img[corners>0.1*corners.max()]=[0,0,255]

corners = cv2.normalize(corners, None, 0, 255, cv2.NORM_MINMAX).astype(np.uint8)
show_img = np.hstack((show_img, cv2.cvtColor(corners, cv2.COLOR_GRAY2BGR)))

cv2.imshow('Harris corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()

fast = cv2.FastFeatureDetector_create(30, True, cv2.FAST_FEATURE_DETECTOR_TYPE_9_16)
kp = fast.detect(img)

show_img = np.copy(img)
for p in cv2.KeyPoint_convert(kp):
    cv2.circle(show_img, tuple(p), 2, (0, 255, 0), cv2.FILLED)

cv2.imshow('FAST corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()

fast.setNonmaxSuppression(False)
kp = fast.detect(img)

for p in cv2.Keypoint_convert(kp):
    cv2.circle(show_img, tuple(p), 2, (0, 255, 0), cv2.FILLED)

cv2.imshow('FAST corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()
```

추출 알고리즘 – Harris corner detector/FAST corner detector

2. White spot NG제품 - code

```
import cv2
import numpy as np

img = cv2.imread('c:/img/white spot.PNG', cv2.IMREAD_COLOR)
corners = cv2.cornerHarris(cv2.cvtColor(img, cv2.COLOR_BGR2GRAY), 2, 3, 0.04)

corners = cv2.dilate(corners, None)

show_img = np.copy(img)
show_img[corners>0.1*corners.max()]=[0,0,255]

corners = cv2.normalize(corners, None, 0, 255, cv2.NORM_MINMAX).astype(np.uint8)
show_img = np.hstack((show_img, cv2.cvtColor(corners, cv2.COLOR_GRAY2BGR)))

cv2.imshow('Harris corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()

fast = cv2.FastFeatureDetector_create(30, True, cv2.FAST_FEATURE_DETECTOR_TYPE_9_16)
kp = fast.detect(img)

show_img = np.copy(img)
for p in cv2.KeyPoint_convert(kp):
    cv2.circle(show_img, tuple(p), 2, (0, 255, 0), cv2.FILLED)

cv2.imshow('FAST corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()

fast.setNonmaxSuppression(False)
kp = fast.detect(img)

for p in cv2.Keypoint_convert(kp):
    cv2.circle(show_img, tuple(p), 2, (0, 255, 0), cv2.FILLED)

cv2.imshow('FAST corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()
```

추출 알고리즘 – Harris corner detector/FAST corner detector

3. Scratch NG 제품 - code

```
import cv2
import numpy as np

img = cv2.imread('c:/img/Scratch.PNG', cv2.IMREAD_COLOR)
corners = cv2.cornerHarris(cv2.cvtColor(img, cv2.COLOR_BGR2GRAY), 2, 3, 0.04)

corners = cv2.dilate(corners, None)

show_img = np.copy(img)
show_img[corners>0.1*corners.max()]=[0,0,255]

corners = cv2.normalize(corners, None, 0, 255, cv2.NORM_MINMAX).astype(np.uint8)
show_img = np.hstack((show_img, cv2.cvtColor(corners, cv2.COLOR_GRAY2BGR)))

cv2.imshow('Harris corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()

fast = cv2.FastFeatureDetector_create(30, True, cv2.FAST_FEATURE_DETECTOR_TYPE_9_16)
kp = fast.detect(img)

show_img = np.copy(img)
for p in cv2.KeyPoint_convert(kp):
    cv2.circle(show_img, tuple(p), 2, (0, 255, 0), cv2.FILLED)

cv2.imshow('FAST corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()

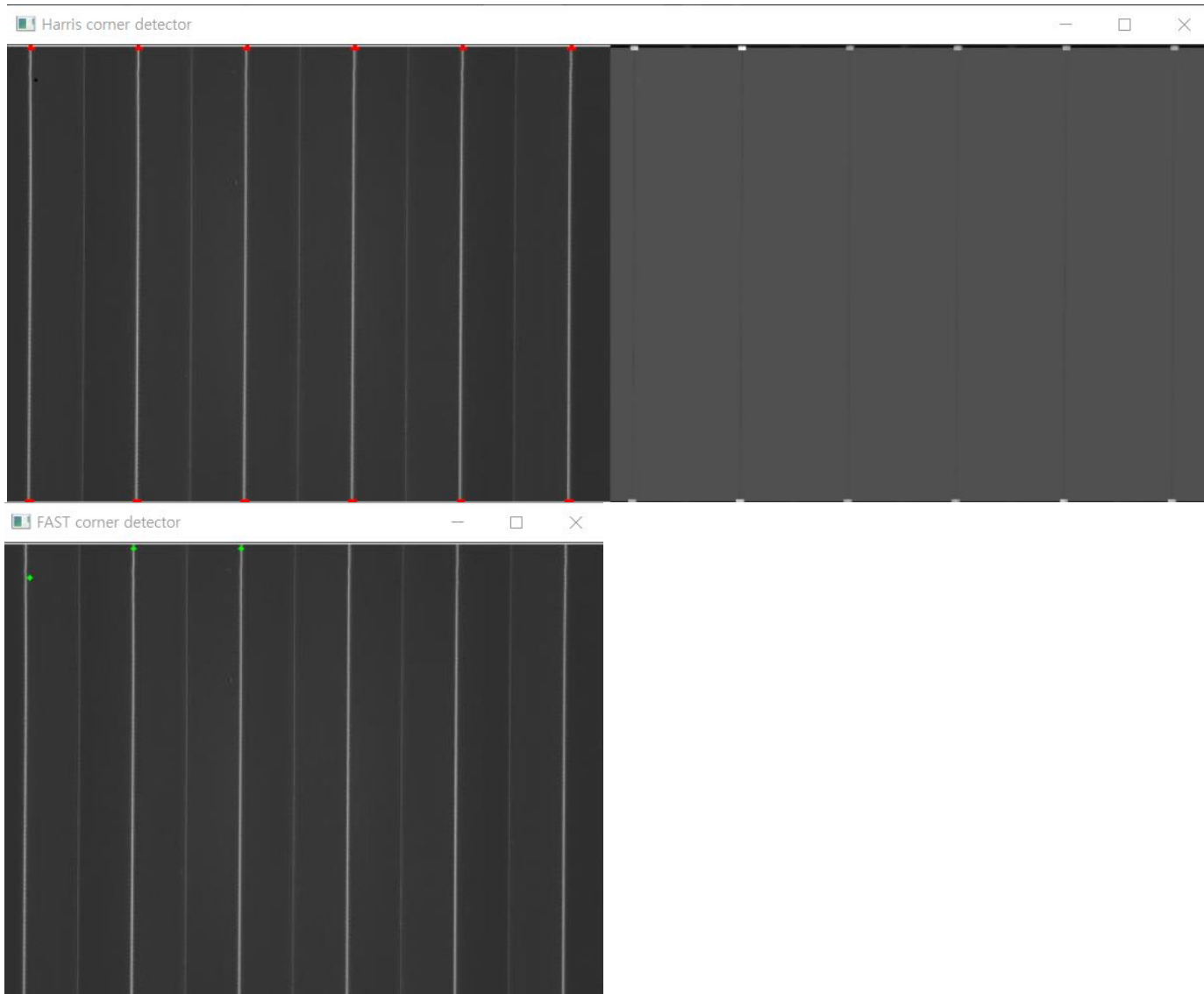
fast.setNonmaxSuppression(False)
kp = fast.detect(img)

for p in cv2.Keypoint_convert(kp):
    cv2.circle(show_img, tuple(p), 2, (0, 255, 0), cv2.FILLED)

cv2.imshow('FAST corner detector', show_img)
if cv2.waitKey(0) == 27:
    cv2.destroyAllWindows()
```

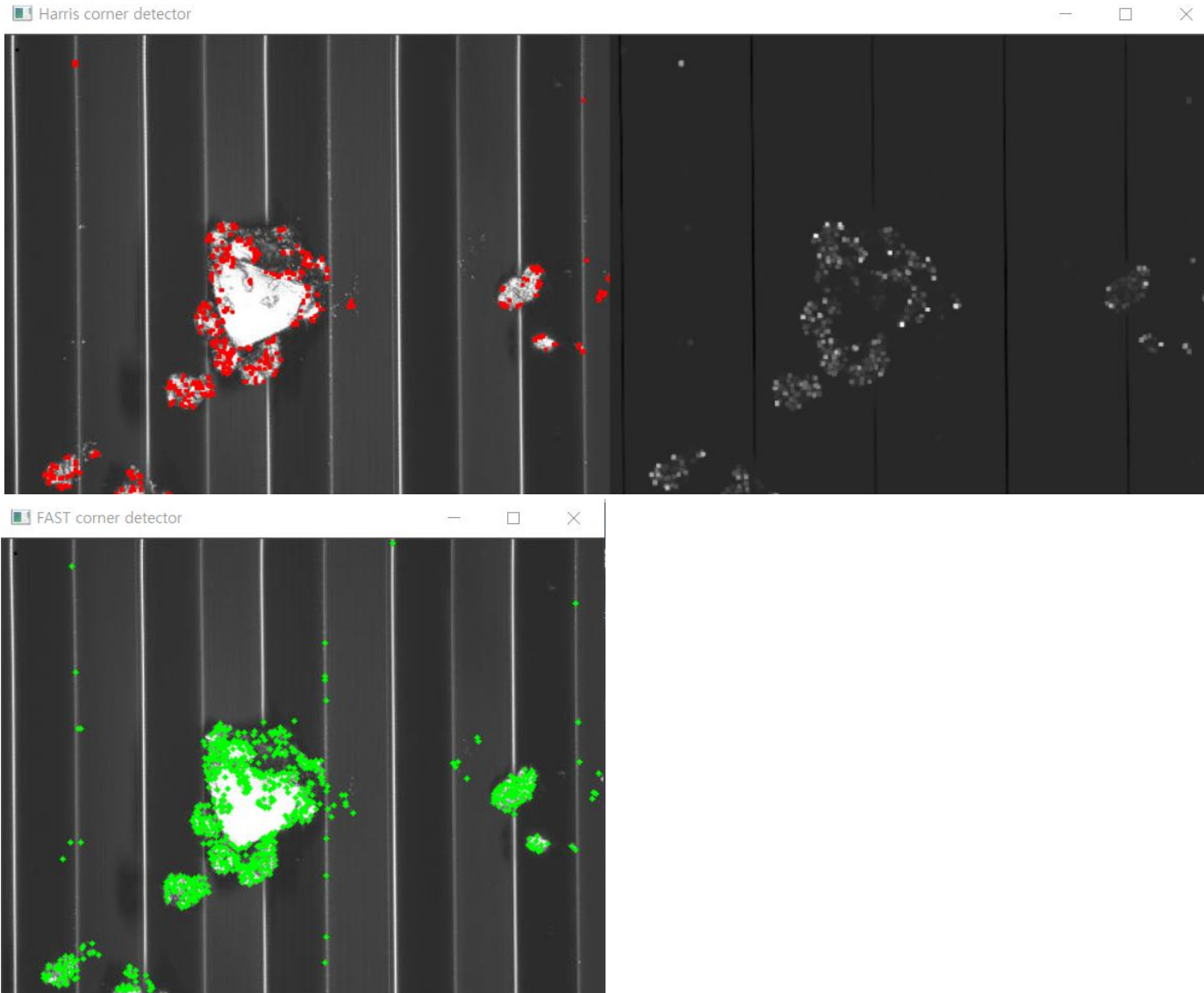
추출 알고리즘 – Harris corner detector/FAST corner detector

1. OK 제품- 실행 결과



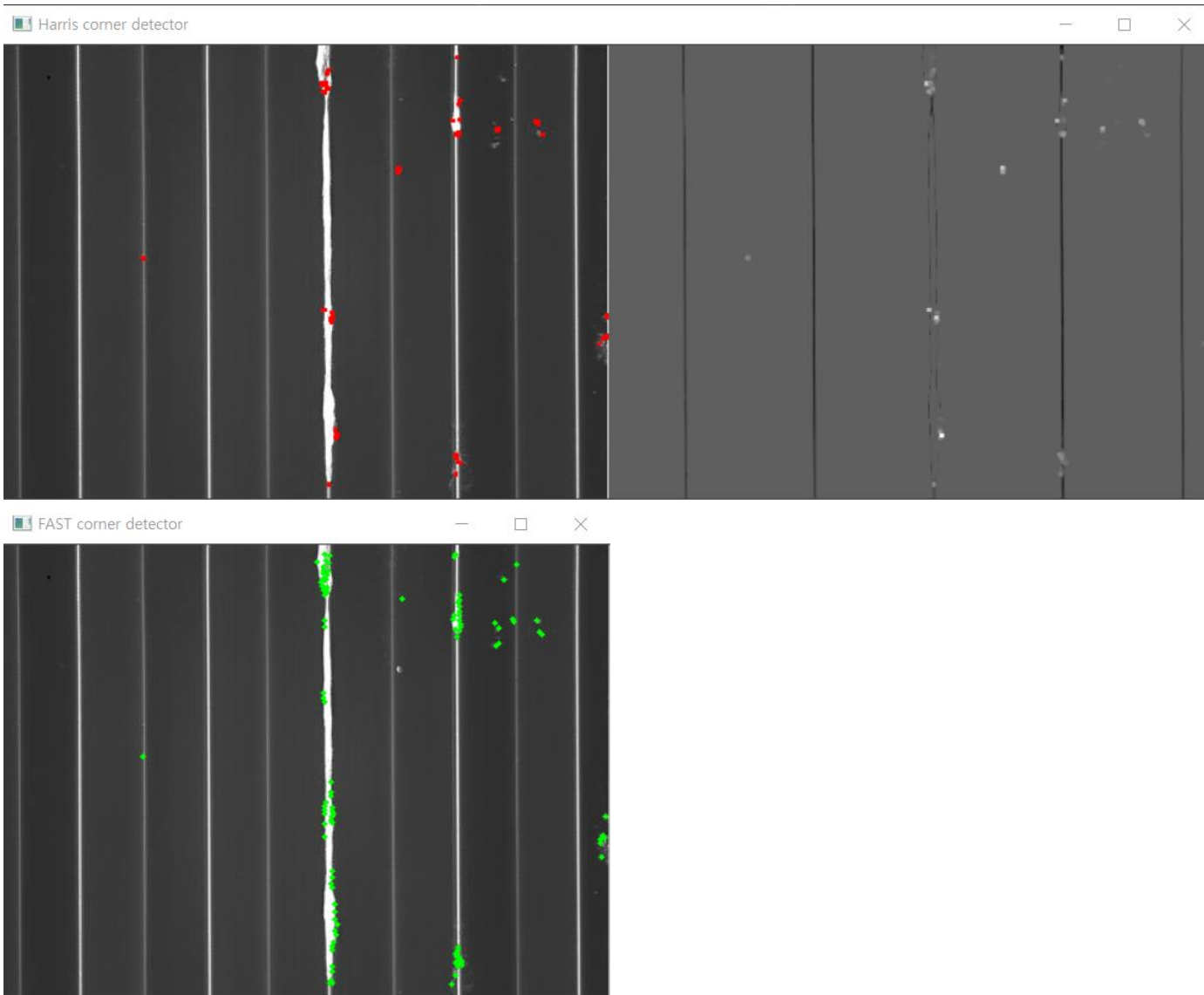
추출 알고리즘 – Harris corner detector/FAST corner detector

2. white spot - 실행결과



추출 알고리즘 – Harris corner detector/FAST corner detector

3. Scratch - 실행결과



실행 결과

- 당사 광학필름의 Prism 형상을 Harris corner detector 및 FAST corner detector 영상처리를 해 본 결과 OK품과 NG품을 확실히 구별 할 수 있었습니다.

감사합니다