
C# 기반 OpenCV4를 사용한 객체 정렬 시스템

개인프로젝트

2022.04.04 - 2022.04.08

CONTENT

객체 정렬 시스템 프로젝트 목차

인플루언서 관리

부제목을 입력하세요.

01

콘텐츠 기획 제작

부제목을 입력하세요.

02

브랜드 뉴스테러 운영

부제목을 입력하세요.

03

SNS운영관리

부제목을 입력하세요.

04

C# 기반 OpenCV4를 사용한

객체 정렬 시스템

#정렬 #aligner

입사검토 메일을 받고 내용을 확인하던 중 자동화기기 OLED Aligner SW라는 문구가 눈에 띄었습니다.

OLED Aligner가 어떤 공정인지 궁금하여 검색을 해보니, 회로 패턴이 담긴 Mask와 회로 패턴을 찍어내기 위해 필요한 Wafer를 정렬하는 공정이었습니다.

만약 학부생 시절에 OLED Aligner 공정을 알고 있었다라면 "공장자동화 시스템"에 접목 시켜볼 수 있지 않았을까?" 하는 **아쉬움**이 들었습니다.

비록 이 공정을 공장자동화 시스템에 접목 시키진 못 했지만 OLED Aligner라는 공정을 접하게 된 이번 기회를 발판으로,

OLED Aligner 공정과 비슷한 작업을 하는 프로그램을 만들어 보는 것도 좋을 것 같다는 생각이 들었습니다.

만들게 된다면 아쉬움도 털고 새로운 분야에 접해보는 좋은 기회이자 경험이 될 것 같았기 때문입니다.

이러한 생각을 바탕으로 Aligner S/W를 개발하게 되었고, 지금의 '객체 정렬 시스템'이 만들어진 이유입니다.

C# 기반 OpenCV4를 사용한

객체 정렬 시스템

#아쉬운 점 #향후 개선점

아쉬운 점

- ☑ 축 정렬 후 mask가 이동을 시작하면 초기 정렬 상태로 바뀌며, wafer에 도착 후 재정렬 하여 align 됨
- ☑ 약간의 오차값이 있음



향후 개선점

- ☑ 축 정렬이 된 상태로 mask를 이동시켜 wafer에 align 하기
- ☑ mask를 이동시킨 후 wafer에서 align 하기
- ☑ 오차값이 없도록 하거나, 오차값 범위를 최소화 시키기
- ☑ 오차 발생 횟수를 최소화 시키기

C# 기반 OpenCV4를 사용한

객체 정렬 시스템

#배운점 #느낀점

배운점

- ☑ 객체 인식을 위한 인공지능과 정렬 알고리즘이 OLED Align S/W에 왜 필요한 기술인지 알게 되었다.
- ☑ 약간의 오차값이 있음

느낀점

- ☑ C#을 사용하여 처음으로 영상처리를 도전해 봤는데 생각보다 재미있었고, 객체를 인식 하는 게 신기했다.
- ☑ OpenCV는 처음 접해봐서 구글링을 해야 했는데 공식 홈페이지에는 C++, Java, Python에 관한 튜토리얼만 있어서 아쉬웠다.

기여도



기여도



THANK YOU°



CONTACT

010-1234-5678
MIRIKIM@MIRICANVAS.COM