

과제#2 Term Project

카메라 모듈 - 이물질 검사를 위한 영상처리

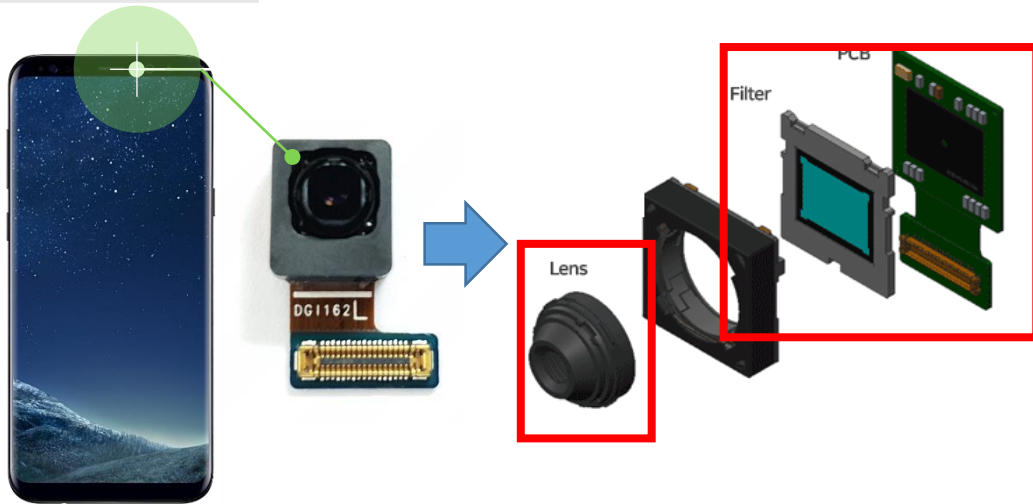
2020254013

김병근

2021.10.20

1. Open CV처리용 영상 수집

자사 생산 제품

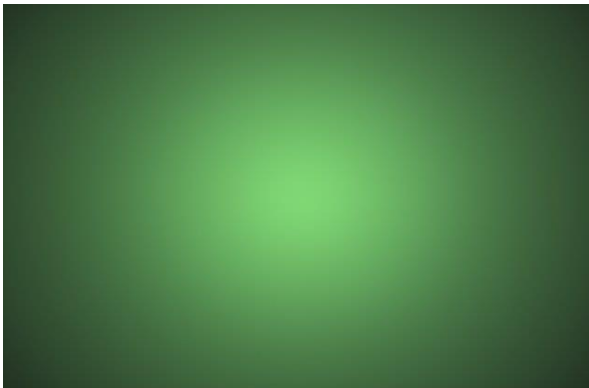


이물질 검증용 영상

영상을 실제 폰에 삽입 후, 육안으로 확인 中

[전수 검사] Lens, Filter, Sensor(PCB)의 이물질 확인

OK



NG(Stain,멍) - 확대

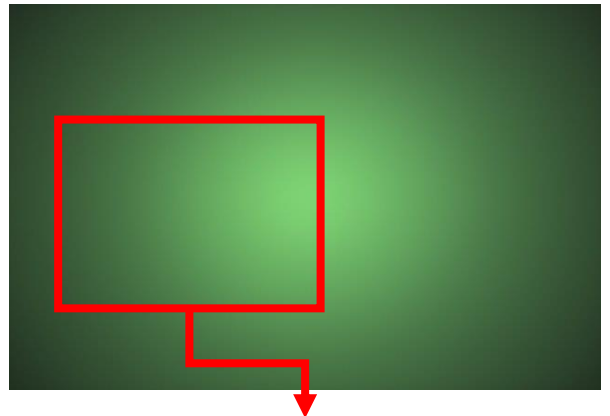


IMAGE 1장
원본(183mb)
→ 전처리 (100kb이하)



2. Term project 진행 개요

진행 내역

1.촬영영상

- ✓ Galaxy 시리즈 카메라 모듈
- 폰검사용 촬영영상 샘플

2.전처리

- ✓ 183mb/1개(화소마다 다름) → 10 kb로 전처리

3.분석

- ✓ 수업시간에 실습한 내용으로 영상 → 이물질 검출 테스트

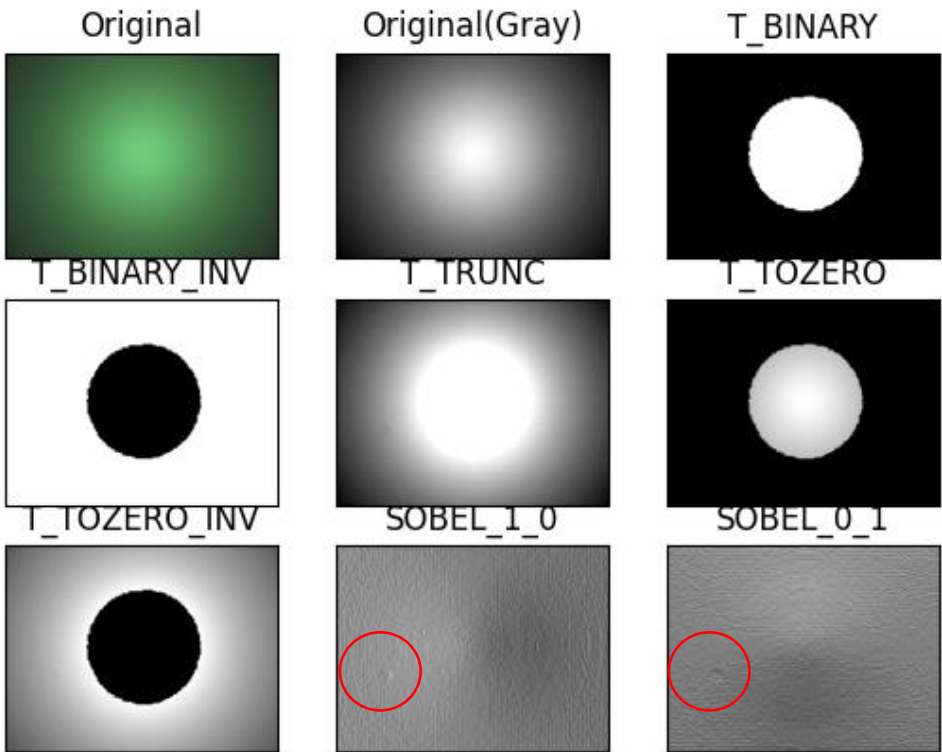


4.방법 선정

- ✓ 6주차 경계선 검출로 선정
- threshold: 다양한 스타일옵션
- sobel filter: 미분차수 변경

- PDF 1주차_산업컴퓨터비전
- PDF 2주차_Image InputOutput & GUI
- PDF 3주차_히스토그램과 필터링 옵션
- PDF 4주차_주파수 기반 필터링
- PDF 6주차_경계선 검출 옵션
- PDF 7주차_Image Segmentation

5.영상처리



3. Term project 개발

Python source

```
1  import cv2
2  import numpy as np
3  from matplotlib import pyplot as plt
4
5  imgOrg = cv2.imread('stain2.jpg')
6  img = cv2.imread('stain2.jpg',0)
7
8  #----- cv2.threshold(src, threshold_value, value, flag)
9  # src: 입력 영상( grayscale)
10 # threshold_value: 픽셀 문턱값
11 # value: 픽셀 threshold값보다 커질때 적용되는 최대값
12 # flag: threshold 적용 스타일
13 ret,thresh1 = cv2.threshold(img,127,255,cv2.THRESH_BINARY)
14 ret,thresh2 = cv2.threshold(img,127,255,cv2.THRESH_BINARY_INV)
15 ret,thresh3 = cv2.threshold(img,127,255,cv2.THRESH_TRUNC)
16 ret,thresh4 = cv2.threshold(img,127,255,cv2.THRESH_TOZERO)
17 ret,thresh5 = cv2.threshold(img,127,255,cv2.THRESH_TOZERO_INV)
18
19 # ---| cv2.Sobel(src, ddepth, dx, dy, dst, ksize, scale, delta, borderType) -----
20 # src: 입력 영상
21 # ddepth: 출력 영상의 dtype (-1: 입력 영상과 동일)
22 # dx, dy: 미분 차수 (0, 1, 2 중 선택, 둘 다 0일 수는 없음)
23 # ksize: 커널의 크기 (1, 3, 5, 7 중 선택)
24 # scale: 미분에 사용할 계수
25 # delta: 연산 결과에 가산할 값
26
27 # dx, dy: 0/1/2 중 실습 시 배운 미분차수가 가장 좋음
28 thresh6 = cv2.Sobel(img, cv2.CV_32F, 1, 0) #dx/dy(미분차수) = 1 / 0
29 thresh7 = cv2.Sobel(img, cv2.CV_32F, 0, 1) #dx/dy(미분차수) = 0 / 1
30
31
32 titles = ['Original', 'Original(Gray)', 'T_BINARY', 'T_BINARY_INV', 'T_TRUNC',
33          'T_TOZERO', 'T_TOZERO_INV', 'SOBEL_1_0', 'SOBEL_0_1']
34 images = [imgOrg, img, thresh1, thresh2, thresh3, thresh4, thresh5, thresh6, thresh7]
35 for i in range(9):
36     plt.subplot(3,3,i+1)
37     plt.imshow(images[i], 'gray')
38     plt.title(titles[i])
39     plt.xticks([])
40     plt.yticks([])
41 plt.show()
```

3. Term project 개발

결과 영상

- ✓ 실습으로 배운 다양한 Filter와 경계선검출의 영상처리를 해본 결과
자사 CM(Camera module)영상은 적합하지 않았다.
좀 더 배우고, OpenCV lib를 통해 다양한 처리기법을 사용하면
될거 같습니다
- ✓ Threshold는 이물질의 경계선이 아닌 밝기에 대한 경계선을 검출 했음
- ✓ Sobel filter는 Original 영상으로 보는 거 보다. 쉽게 식별 가능 함

