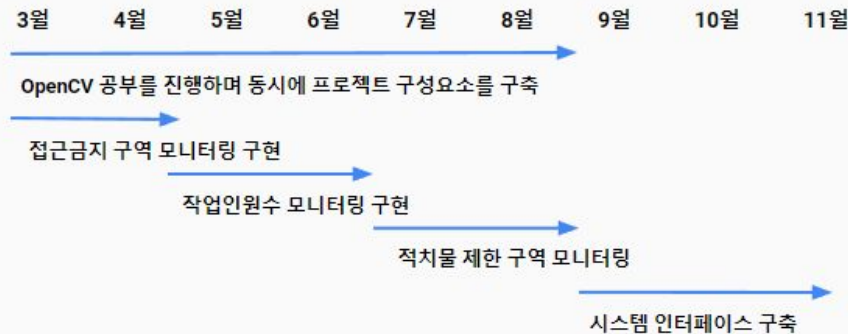


4주차

1. 세부 계획
2. 세부계획에 따른 진행
 - a. 참고도서 정리
 - b. 이미지에 대하여
 - i. 이미지란 무엇인가
 - ii. **GrayScale** 구현
 - iii. 이미지는 어떻게 저장되는가(jpg)
 - c. 필요한 세부 기술 찾기

세부 계획



기본 일정을 참고하여 주마다 세부계획을 세워 따른다.

구체적인 라이브러리의 사용법은 필요가 발생 할 때마다 공부한다.

기본적인 원리와 **OpenCV**의 기본기를 위해서 참고도서를 공부한다.

참고도서공부는 3월~5월에 걸쳐서 공부한다.

4주차 계획 [3/14~3/20]

1. 책을 일독 하여 정리한다.

4주차의 예제 실습은 **GrayScale**을 직접 구현 하는 것으로 같음한다.

2. 이미지에 대하여 공부한다.

[이미지 형태를 파악, **GrayScale**을 직접 구현, 이미지의 저장방법(jpg)]

3. 접근금지 구역 모니터링을 구현하기 위한 세부 기술을 파악한다.

공부 세부 계획-참고도서 OpenCV 학습

+**[알짜배기 예제로 배우는 OpenCV]**의 구성요소

책은 이미지의 동작을 다루는 부분을 제외하고
기능의 원리에 대한 간략한 소개와 이를 구현하는 법에 초점이 맞추어져 있다.

+책을 공부하는 방법은 다음과 같다.

코드의 구체적인 구성요소를 제외하고 책을 전부 읽는다.
책에서 소개하고 있는 **opencv**의 기능을 정리한다.[이번 4주차에 한다.]
필요해보이는 예제를 구현해본다.[3월~5월에 걸쳐 실습한다.]

참고도서 정리1-각 기술의 목록과 요약 1

-이미지와 동영상을 불러오는법

-이미지의 컴퓨터에서의 동작

-이진화=>이미지를 흑, 백과 같이 두가지의 색으로만 표현함=>영역별로 다른 임계값을 적용하여 더 좋은 이진화 된 이미지를 만드는 법=>이진화를 사용하는 이유는 이진화를 통해 이미지 안의 특징과 사물을 쉽게 추출 할 수 있도록 하기 위함이다.

-블렌딩=>두개의 입력 이미지의 투명도를 조정하여 이미지가 겹쳐보이도록함.

참고도서 정리1-각 기술의 목록과 요약 2

- 차영상=>같은 장소를 촬영한 두가지의 이미지를 이용하여 새로 추가된 객체를 검출함
- 다양한 그리기 도구
- 회전행렬에 의한 이미지 회전
- 이미지의 크기 조정
- 이동행렬에 의한 이미지 이동
- 퍼스펙티브변환=>시점을 변환함

참고도서 정리1-각 기술의 목록과 요약 3

-블러링=>이미지에서 노이즈 제거

-에지 검출=>소벨,캐니에지디텍터, 미분을 통해 픽셀 값이 급격히 변하는 부분을 검출하여 이미지의 테두리를 그려낼 수 있다.

-모폴로지=>이진화된 이미지의 흰색으로 나타내지는 오브젝트 영역의 형태 개선

참고도서 정리1-각 기술의 목록과 요약 4

- 허프라인트랜스폼=>이미지에서 직선을 찾는 알고리즘
- 허프서클트랜스폼=>이미지에서 원을 찾는 알고리즘
- 히스토그램=>이미지를 구성하는 픽셀값 분포=>히스토그램 평활화,CLAHE 를 통해 밝은 부분과 어두운 부분의 밝기차이를 조절하여 이미지를 조절할 수 있다.
- 템플릿 매칭=>템플릿 이미지와 일치하는 영역을 입력 이미지에서 찾음.

참고도서 정리1-각 기술의 목록과 요약 5

-영상분할

=>이미지에서 관심있는 부분을 분리해내는 방법

=>가장 기본적으로 이미지를 그레이스케일로 변환한 후 이진화하여 분리 할 수 있다.

=>색을 특정하여 그 색을 가진 구성요소만을 분리 할 수 있다.

=>라벨링을 통해 각 이진화 이후의 각 흰색 영역마다 번호를 부여 할 수 있다.

=>Background Substraction을 통해 배경을 제거하고 물체를 검출한다(일정 시간 배경이 되는 장소를 촬영후, 새로운 객체가 등장하면 해당 객체만 검출 할 수 있다.)

참고도서 정리1-각 기술의 목록과 요약 6

-컨투어 검출

=>특정 영역의 경계를 따라 같은 픽셀값을 갖는 지점을 연결, 영역에 대한 모양 분석이나 오브젝트 검출을 위한 전처리

=>영역의 크기를 계산

=>영역을 직선으로 근사화

=>영역의 무게중심 계산

=>영역의 경계사각형 그리기

=>Convex Hull, 영역을 구성하는 모든 점으로 포함하는 블록 다각형 그리기

=>Convex Defects, 영역이 오목하게 들어간 부분을 표시

참고도서 정리1-각 기술의 목록과 요약 7

-물체추적

=>특정색 기반으로 추적

=>Meanshift, 물체추적 알고리즘으로, 최초로 선택한 위치에 적당한 크기의 영역을 형성하고 그 안의 무게중심을 계산하여 최초에 선택한 위치를 그 무게중심으로 계속 이동시켜 그러한 이동이 정지될 때 까지 반복하는 방식으로 이루어진다.

=>Camshift, Meanshift에서는 객체가 카메라에 가까이 이동하거나 멀리이동하여도 경계사각형이 일정하다는 문제가 있다. Camshift는 이점을 고려하여 경계사각형의 크기가 조절하며, 회전도 하게 해준다.

이미지란 무엇인가

이미지는 픽셀의 집합이다.

각 픽셀은 숫자 배열의 형태이며, 숫자는 RGB의 정도를 나타낸다.

```
1 import cv2 as cv
2 img_color=cv.imread("opencvStudy/3.jpg",cv.IMREAD_COLOR)
3 print(img_color.shape)
```

```
c:/Users/Hamster/
(225, 225, 3)
```

이미지의 형태

우측의 컵사진은 높이 225픽셀 너비 225픽셀이며
3개의 각 채널이 있다.

즉, rgb 3개의 값이 225*225개가 있다고 이해 할 수 있다.



GrayScale로의 직접 변환

GrayScale은 이미지의 특징을 추출하기 쉬운 이진화의 전처리로 가장 많이 이용된다. 이를 위해서는 **rgb**값을 대응하는 공식에 넣어 픽셀마다 변환하여 구현할 수 있다.



GrayScale로의 직접 변환

```
img_color=cv.imread("opencvStudy/3.jpg",cv.IMREAD_COLOR)

#이미지의 높이와 너비
height,width=img_color.shape[:2]

#gray scale image를 저장할 numpy 배열을 생성
img_gray=np.zeros((height,width), np.uint8)#8bit unsigned integer
```

이미지의 높이와 너비를 구한다.

GrayScale로 변환 한후 그 이미지 픽셀 배열을 저장 할 Numpy배열을 준비한다.

GrayScale로의 직접 변환

```
for y in range(0, height):  
    for x in range(0, width):  
        b=img_color.item(y,x,0)  
        g=img_color.item(y,x,1)  
        r=img_color.item(y,x,2)  
  
        gray=int(r*0.216+g*0.7152+b*0.0722)  
        img_gray.itemset(y,x,gray)
```



오리지널 이미지에서 각 픽셀별 RGB값을 읽어온 후 이를 공식에 맞게 변형하여 GrayScale이미지의 픽셀 배열에 저장한다.

이미지는 어떻게 저장되는가

+래스터 방식

색상정보가 담긴 픽셀로 표현되는 방식
픽셀의 수가 많을 수록 이미지의 질이 좋아진다.
JPEG, GIF, PNG, TIFF, BMP, Raw 형식이 존재한다.

+벡터 방식

수학적인 수식을 통해 표현되는 방식
확대 및 축소에 영향 받지 않는 선명함을 가진다.
AI, SVG, VML, CGM, 거버 포맷

이미지는 어떻게 저장되는가 - JPEG

1) 이미지 색 공간 변환: RGB 색공간에서 YCbCr 색공간으로 전환한다.

Y는 픽셀의 밝기를 나머지는 색차를 의미한다.

2) 다운샘플링: 색과 관련된 채널들은 밝기 채널에 비해 덜 중요하므로 다운샘플링한다.

4:2:0 방식을 사용한다. 2X4사이즈의 픽셀 블록에서 첫 번째 행에서 2개 두 번째 행에서 0개를 추출하고 나머지를 버린다.

3) 이산코사인변환(DCT): 각 채널을 8x8 블록으로 분할한 다음에 각 블록마다 이산코사인변환을 시행한다.

Y채널은 8X8 나머지는 16X16 블록으로 분할한다. 픽셀 값의 범위를 -128~127로 변환한후 DCT를 각 블록마다 시행한다. 주파수 관점으로 이미지를 보게 만든다. 이 때 왼쪽 위에 있는 가장 큰 값을 가지는 계수를 DC 계수, 나머지를 AC 계수라고 한다.

이미지는 어떻게 저장되는가 - JPEG

4) 양자화: DCT 계수들을 양자화해서 높은 고주파 신호와 관련된 계수들은 대부분 0이 되게 한다.

이것은 고주파 신호를 제거하기 위함이다. 정해진 양자화 행렬로 나눈후 반올림한다.

5) zigzag 스캐닝: 2차원 배열에 담겨져 있는 양자화된 계수들을 지그재그 순서로 읽어 1차원 벡터가 되게 한다.

6) 양자화된 DC계수 부호화: DC계수들을 DPCM과 허프만 인코딩을 이용해 부호화한다.

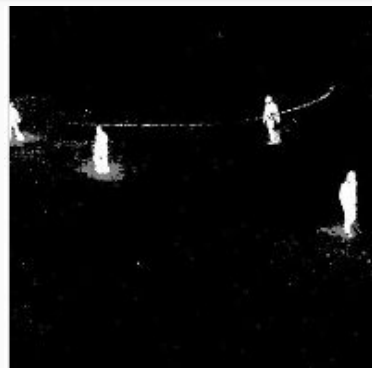
DC계수는 해당 블록의 명도와 관련되어 있다. 인접한 블록의 DC계수는 비슷한 값이 많으므로 그 차이를 구하여 저장한다.

7) 양자화된 AC계수 부호화: AC계수들을 런 력스 부호화와 허프만 인코딩을 이용해 부호화한다.

0과 1로 부호화 한다.

접근금지구역 모니터링에 필요한 것

1. 사람이 접근금지 구역에 있으면 경고한다.
2. 접근금지구역을 카메라로 통해 감시한다.



+영상분할에서 **Background Subtraction**을 사용하는 것이 기초가 된다.

접근금지구역 모니터링에 필요한 것

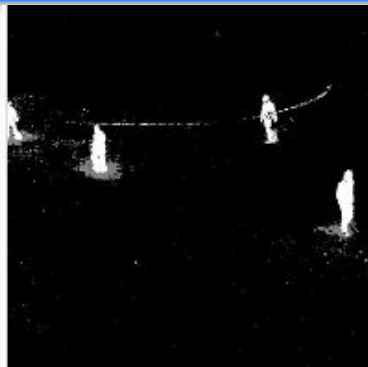
Background Substraction을 사용하기 위해서
선두로 공부해야 하는 것이 있다.

1. 블러링
2. 모폴로지

노이즈를 제거하여 선명한 이진 영상을 추출하기 위해서 필요하다.

3. 라벨링

하얀색으로 표시되는 영역을 구분하고 이를 개별적으로 다루기 위해서
필요하다.



접근금지구역 모니터링에 필요한 것

5주차에는 이상의 기술 사용법을 익히고

이후 필요한것은 다시 생각해보는 시간을 갖는다.