

Open CV를 이용한 번호판 인식기



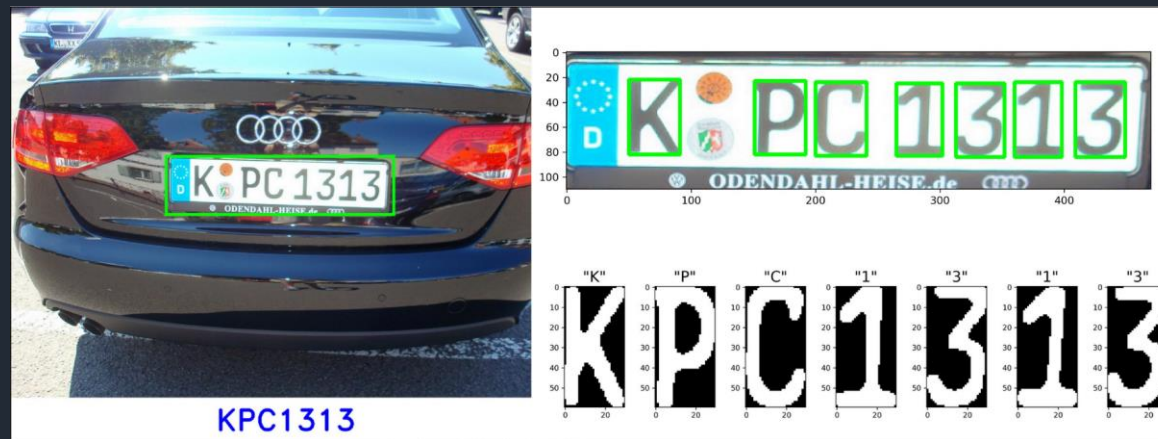
Date : 2021 . 04 . 02

발표자 : 27기 18이기진

CONTENTS

- Project Introduction
 - Motive for Selection
 - Development Goals
- Project Contents
 - System Architecture
 - Project Contents
 - Project Scenarios
- Project Schedule
- Q & A

- 반복적인 작업뿐 아니라 일상에서 일어나는 상황들에 대해 잘 대처하려면, 로봇도 인간과 마찬가지로 눈이 필요하다.
- 로봇의 눈은 주변 물체와의 거리를 계산하거나, 정보를 읽어내는 능력 등이 필요하다.
- 이번 작품에서는 번호판 인식기를 만들어 주변 숫자 및 글자 정보에 대해 파악할 수 있도록 만들어 보고자 한다.



가. 웹캠 영상 불러오기

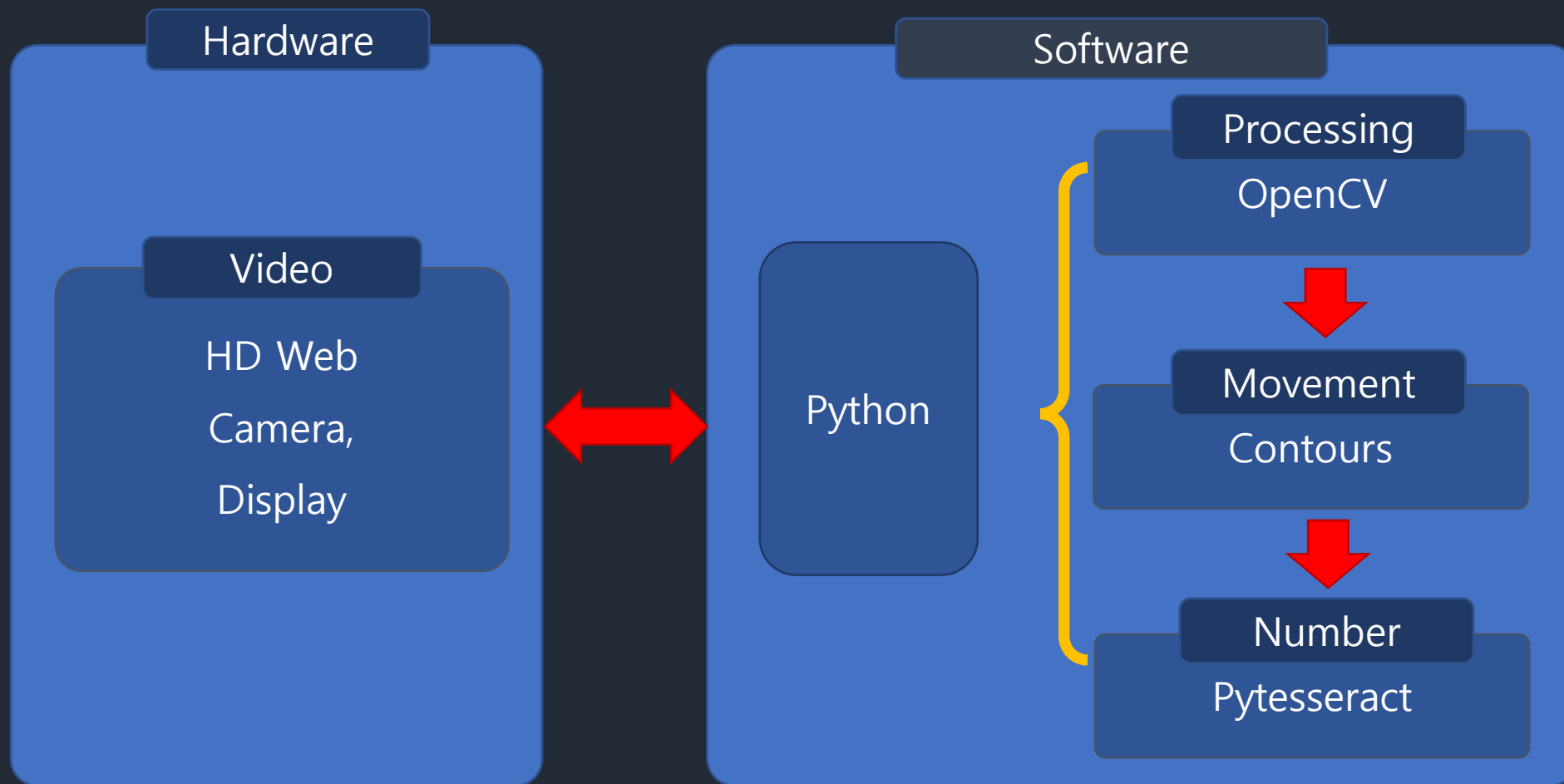
-웹캠 앞에서 번호판을 들고 돌아다님

나. 번호판 부분을 감지

-번호판 숫자부분을 직사각형으로 표시하고, 거리 및 높이 등을 계산하여 번호판이라는 것을 알아냄

다. 번호판의 숫자를 인식

-광학 문자 인식 라이브러리를 통하여 번호판의 숫자 및 문자 인식

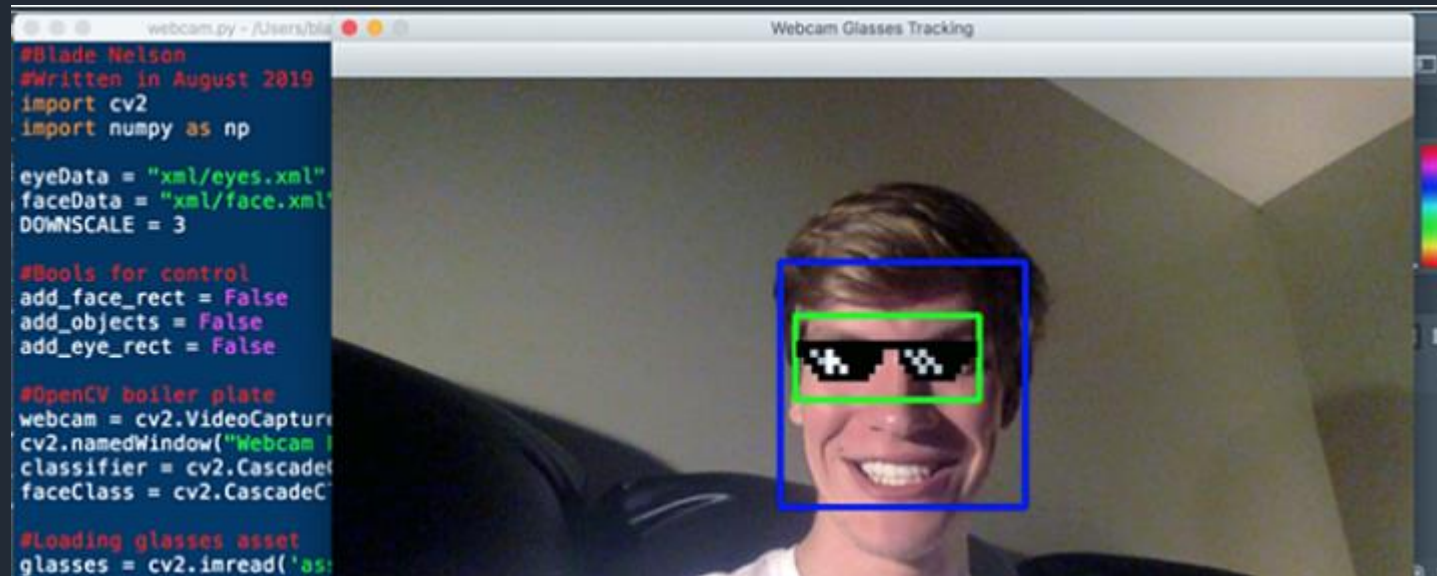


사람이 웹캠 앞에서 숫자[폰트]가 적혀있는 번호판을 들고 있으면 종이의 숫자 영역을 인식 및 출력한다.

Project Contents

[1] 웹캠 영상의 실시간 출력

VideoCapture 함수를 이용하여 웹캠의 영상을 실시간으로 불러온다.



cv2.VideoCapture(0)

괄호 안의 0번은 노트북의 기본 카메라, 1번부터는 외장으로 연결된 웹캠이 할당된다.

Project Contents

[2] 영상 처리

OpenCV에서 번호판 영역 및 숫자의 높은 인식률을 위하여, 각 영역이 잘 구분되도록 해주는 몇가지 영상 처리가 필요하다.

A. GrayScale

BGR포맷 이미지는 0~255의 총 256가지 색상값으로 색을 표현한다. 이 이미지를 회색으로 만들기 위해 Grayscale을 사용하며, 이 때 색상정보는 없이 밝기정보 0~255로 이미지를 표현한다. 검은색은 0, 흰색은 255이다.

```
cv2.cvtColor(video, cv2.COLOR_BGR2GRAY)
```



Project Contents

[2] 영상 처리

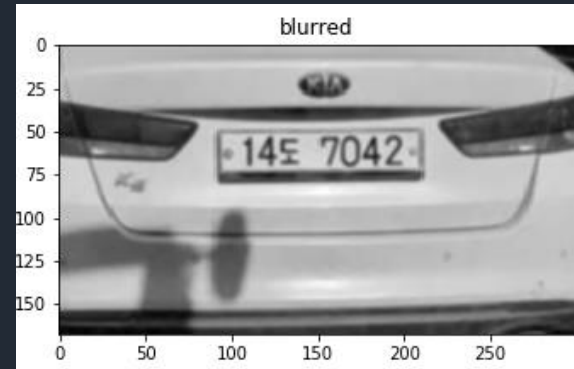
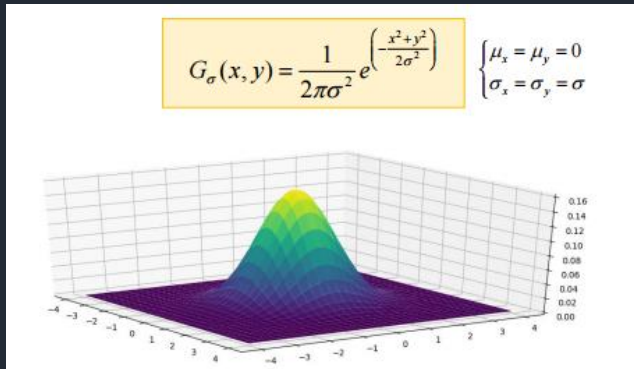
B. Gaussian Blur

이미지를 블러 처리하여 이미지의 노이즈 및 손상을 줄인다. 번호판을 포함한 이미지 전체를 조금씩 뿌옇게 만든다.

```
dst =cv2.GaussianBlur(src, ksize, sigmaX[, dst[, sigmaY[, borderType=BORDER_DEFAULT]]])
```

src: 입력 이미지, ksize: 커널의 크기, sigmaX, Y: 각 축에 따른 표준 편차

이미지가 너무 흐릿해 지지 않도록 적당한 커널의 크기와 시그마값을 찾아야 한다.



Project Contents

[2] 영상 처리

C. Binarization

영상 이진화는 RGB로 다양하게 분포되어 있는 색상값들을 0과 1만으로 표현하는 것이다. 즉, 검은색, 흰색으로만 나타내겠다는 것이다. Gray로 바뀐 RGB영상은 하나의 픽셀당 0~255의 값중 하나를 갖게 된다. 그리고 사용자가 임계값(Threshold)을 설정해주면 그 값을 초과하는 픽셀은 1로, 그 값 이하의 픽셀은 0으로 변환시킨다.

```
cv2.threshold(gray, 100, 255, cv2.THRESH_BINARY)
```



Project Contents

[3] 번호판 부위 인식

웹캠 앞에서 사람이 번호판 모형을 들고 돌아다니면, 표지판이 있는 부분을 인식하여 잘라낸다.

A. Contours

Countour는 윤곽선이다. 모양 분석이나 객체 검출에 사용된다. 이진화까지 마친 이미지 중에서 번호판의 각 숫자 부분을 Countour를 통하여 검출후 직사각형 표시한다.

```
cv2.findContours(thr, cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
```

thr: 소스 이미지, cv2.RETR_TREE: contour 추출모드,
cv2.CHAIN_APPROX_SOIMPLE: contour 근사 방법이다.

Project Contents

[3] 번호판 부위 인식

```
cv2.drawContours(img, contours, -1, (0, 255, 0), 3)
```

findContours를 통해 찾아낸 윤곽선을 실제로 그리는 함수이다.

Img: 대상 이미지, contours, -1: contour의 인덱스 파라미터(-1일땐 윤곽선을 다 그려준다),
(0, 255, 0): 윤곽선의 BGR값, 3: 선의 두께

```
x, y, w, h = cv2.boundingRect(contour),  
(xcenter, ycenter) = (x + (w/2), y + (h/2))
```

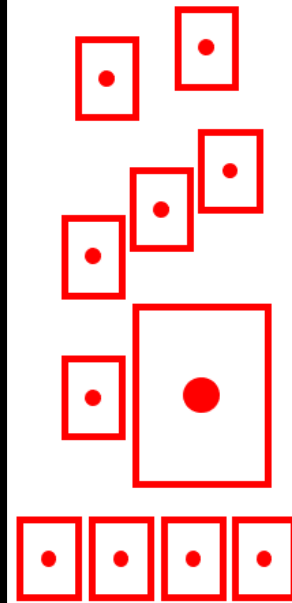
번호판 부분을 수월히 인식하기 위해서 각 숫자들을 감싸는 사각형을 찾아낸다.

x, y좌표와 너비, 높이 정보를 리스트에 저장한다.

각 사각형의 중심좌표도 저장한다.

[3] 번호판 부위 인식

여러 BoundingRect들 중에서 번호판의 모양새를 한 Rect를 찾아낸다.
번호판 숫자 및 문자 폰트는 크기가 일정하다.
우측과 같은 기준을 적용한다.



1. 번호판에서 숫자 Rect들 중심좌표 사이의 거리가 숫자 1개 가로의 3배 이내에 드는지 Check!
2. 번호판 숫자 Rect들 사이의 각이 차이가 너무 많이 나면 out!
3. 번호판 숫자 Rect들의 넓이차이가 너무 크면 out!
4. 번호판 숫자 Rect이 붙어있는 개수가 최소 7개 이상이면 Check!

Project Contents

[3] 번호판 부위 인식

B. 이미지 정렬(Affine Transform)

번호판이 비대칭으로 보이거나 삐뚤게 보일 때도 인식을 할 수 있어야 한다.

가. 방법1

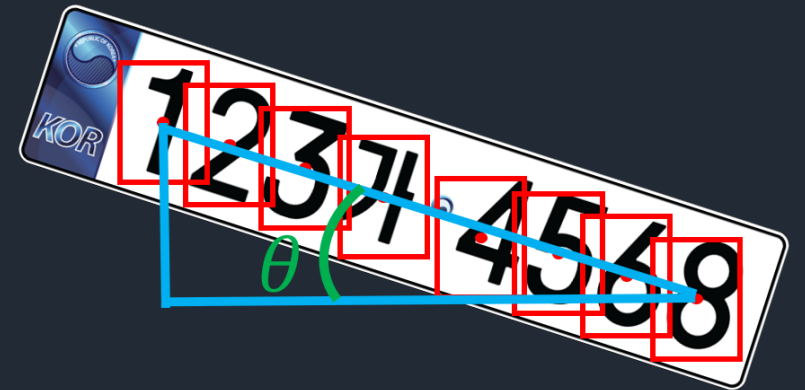
번호판은 왼쪽부터 읽게 되므로 x방향에 순차적으로 정렬을 해준다. 숫자 사각형들이 삐뚤어져 있다면, 수평과의 사이각을 구한다(arctan함수 사용).

```
cv2.getRotationMatrix2D(center, angle, scale)
```

위의 함수를 사용하여 수평면과 중심을 이은 선을 일치시킨다.

```
cv2.getRectSubPix(img_rotated, box_size, center)
```

회전된 이미지를 바탕으로 번호판 부분 이미지를 자른다.



Project Contents

[3] 번호판 부위 인식

나. 방법2

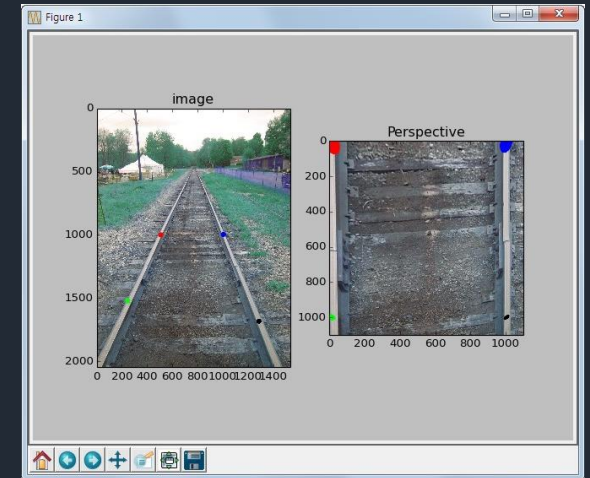
번호판의 맨 첫 번째 숫자의 왼쪽 두 꼭짓점과 맨 마지막 숫자의 오른쪽 두 꼭짓점을 4개의 점으로 잡고 아래의 함수를 사용한다.



```
cv2.getPerspectiveTransform()
```

```
cv2.warpPerspective()
```

위에서 얻은 행렬을 사용하여 영상(이미지)을 회전시킨다.



[4] Pytesseract를 이용한 숫자 감지 및 출력

Tesseract는 OCR 이미지로부터 텍스트를 인식 및 추출하는 소프트웨어이다, 번호판 이미지가 깔끔하게 정렬 되면 Tesseract-OCR을 통해서 그 영역의 글자가 무엇인지 인식 및 추출한다.

```
pytesseract.images_to_string(img, lang='kor', config='--psm7 --oem')
```

이미지를 문자열로 바꿔주며, 언어는 한국어, 설정값은 psm7로 문자열이 한줄로 나열되어 있음을 의미한다. 뒤의 oem은 테서렉트 엔진 버전이다.



Tesseract OCR

1. 사람이 번호판 모형을 들고 카메라 앞을 돌아다닌다.
2. 번호판의 각각 숫자와 글자에 직사각형 테두리가 쳐진다.
3. 인식한 번호가 모니터에 출력된다.



번호판 모형은 하드보드지에 붙여서 만든다.

03

Project Schedule

	3M	4M					5M		
	5w	1w	2w	3w	4w	5w	1w	2w	3w
실시간 영상 받아오기			중간고사						
Gray Scale, 필터, 이진화 처리 etc									
Contours로 숫자 부분 Bindary Rectangle 인식									
패키지로 글자 인식 및 출력									
디버깅 및 테스트									

Q&A

Thank You for Listening