

# 사물추적 데이터 어그멘테이션

김진용

## 최종 목표 Task

- 드론에 장착된 카메라로 화면 상에 보이는 moving platform에 달려있는 공을 실시간으로 감지하여 안정적으로 접근하고 착륙

## 맡은 역할

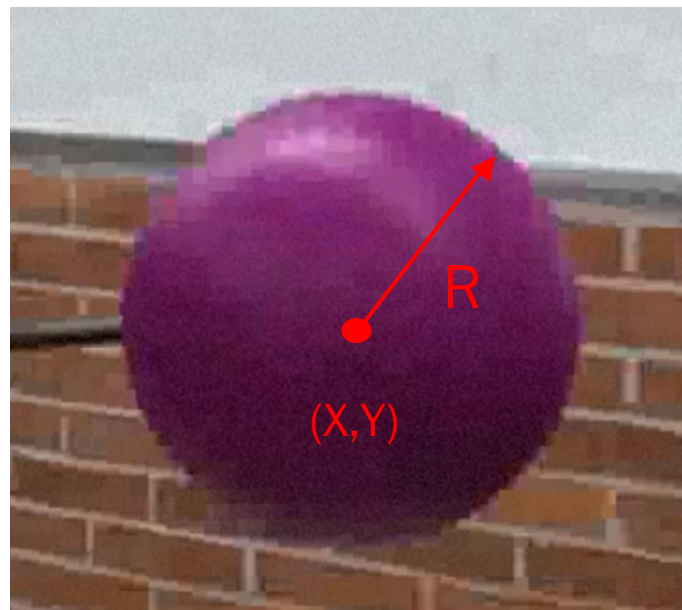
- 공이 포함된 이미지 데이터 수집
- 클래스를 구별하지 않고 공 객체가 있는지 없는지 만을 판별해야 함.
- 공 객체 bounding box를 직접 labeling 함.
- 부족한 데이터 셋을 oversampling 하기 위하여 augmentation함.
- Augmentation한 이미지에 대하여 좌표를 알맞게 변환해주고 csv에 정해진 format으로 저장

TASK

DATA SET

데이터

LABEL



$X,Y,R$

# Image data labeling Tool

## VGG Image Annotator

장점 : 설치 필요없이 웹페이지에서 바로 가능하다.



<http://www.robots.ox.ac.uk/~vgg/software/via/via-1.0.6.html>

## csv 파일

file_attribu	region_cou	region_id	region_shape_attributes	region_attributes		
{}	1	0	{"name":"circle","cx":62,"cy":90,"r":146}	{}		
{}	1	0	{"name":"circle","cx":62,"cy":90,"r":146}	{}		
{}	1	0	{"name":"circle","cx":62,"cy":90,"r":146}	{}		
{}	1	0	{"name":"circle","cx":62,"cy":90,"r":146}	{}		
{}	1	0	{"name":"circle","cx":62,"cy":90,"r":146}	{}		
{}	1	0	{"name":"circle","cx":308,"cy":208,"r":111}	{}		
{}	1	0	{"name":"circle","cx":307,"cy":209,"r":111}	{}		
{}	1	0	{"name":"circle","cx":306,"cy":209,"r":109}	{}		
{}	1	0	{"name":"circle","cx":307,"cy":209,"r":110}	{}		
{}	1	0	{"name":"circle","cx":307,"cy":208,"r":110}	{}		

## Json 파일

```
{'1.jpg29339': {'fileref': '', 'size': 29339, 'filename': '1.jpg', 'base64_img_data': '',
'file_attributes': {}, 'regions': {'0': {'shape_attributes': {'name': 'circle', 'cx': 59, 'cy': 87, 'r': 150},
'region_attributes': {}}}
```

```
1  # -*- coding: utf-8 -*-
2  """
3  Created on Fri Nov 13 19:56:12 2020
4
5  @author: user
6  """
7
8  import numpy as np
9  import json
10 #import pandas as pd
11 import csv
12
13 file = 'E:/김진용/AIMS연구실/201117미팅/via_region_data.json'
14 f=open(file, 'r', encoding='utf-8')
15 data=json.load(f)
16 f.close()
17
18
19
20
21
22 label_x=[]
23 label_y=[]
24 label_r=[]
25 file_name=[]
26
27 for i in data.keys():
28     label_x=np.append(label_x,int(data[i]['regions']['0']['shape_attributes']['cx']))
29     label_y=np.append(label_y,int(data[i]['regions']['0']['shape_attributes']['cy']))
30     label_r=np.append(label_r,int(data[i]['regions']['0']['shape_attributes']['r']))
31     file_name=np.append(file_name, data[i]['filename'])
32
33
34
35 label=[label_x, label_y, label_r, file_name]
36 label=np.transpose(label)
37
38 with open('E:/김진용/AIMS연구실/201117미팅/datalabeling.csv', 'w', newline='') as f:
39     wt=csv.writer(f)
40     wt.writerows(label)
41
```

data의 type  
dictionary

# X Y R img

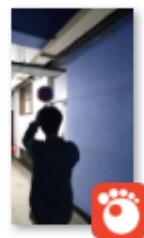
H PC > 저장용 (E:) > 김진용 > AIMS연구실 > datalabeling

이름	날짜	유형
1번영상	2020-11-18 오후 11:29	파일 폴더
2번영상	2020-11-18 오후 11:29	파일 폴더
3번영상	2020-11-19 오전 9:19	파일 폴더
4번영상	2020-11-19 오전 9:19	파일 폴더
5번영상	2020-11-19 오전 9:20	파일 폴더
extract_video.py	2020-11-19 오전 9:51	PY 파일

PC > 저장용 (E:) > 김진용 > AIMS연구실 > datalabeling > 1번영상



data



1.mp4



datalabeling.csv

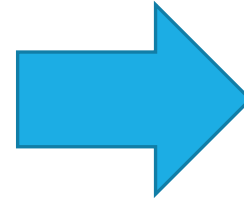


via\_region\_data.json

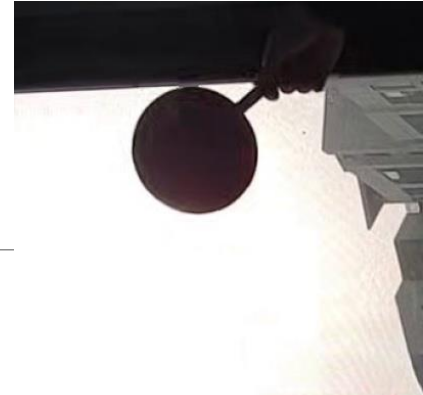
	A	B	C	D
1	580	497	80	1.jpg
2	586	470	82	2.jpg
3	588	441	83	4.jpg
4	581	421	84	5.jpg
5	581	431	84	6.jpg
6	581	492	78	7.jpg
7	615	570	79	8.jpg
8	635	635	75	9.jpg
9	633	715	81	10.jpg
10	616	762	86	11.jpg
11	598	827	87	12.jpg
12	596	813	83	13.jpg
13	581	728	80	14.jpg
14	579	633	81	15.jpg
15	574	548	79	16.jpg
16	538	513	81	17.jpg
17	492	489	76	18.jpg
18	460	477	71	19.jpg
19	452	467	79	20.jpg
20	441	441	83	21.jpg
21	448	450	79	22.jpg
22	475	463	77	23.jpg
23	533	479	79	24.jpg

## Data Augmentation

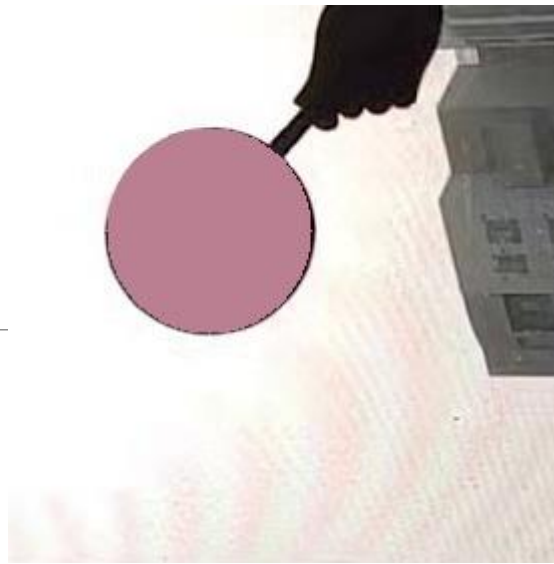
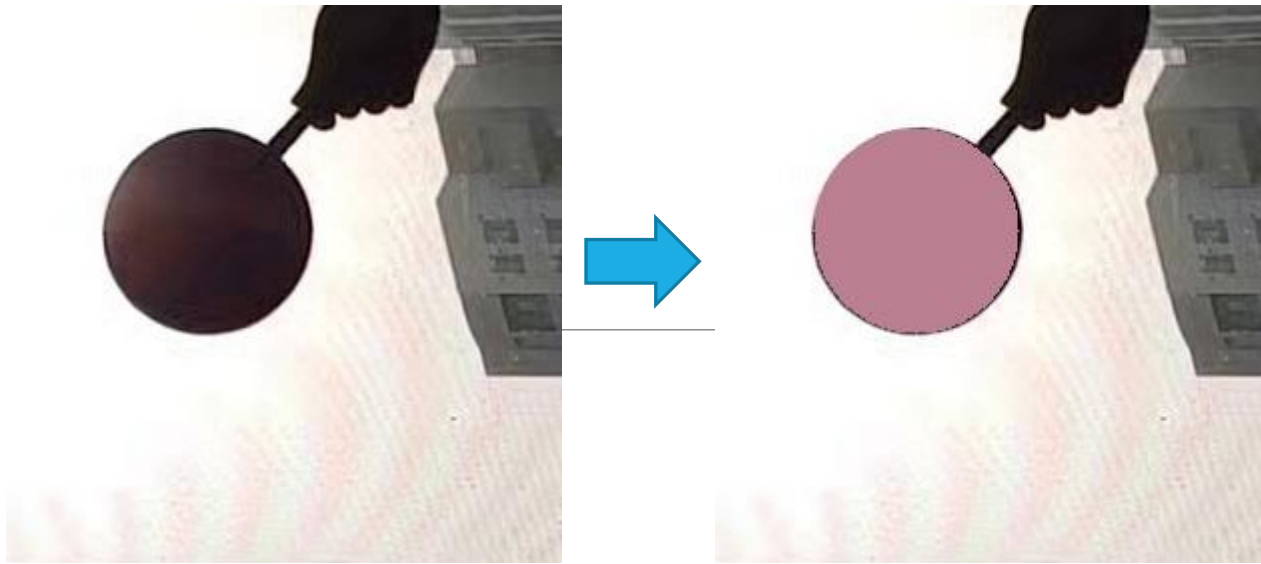
원본



crop







수작업으로 얻은  
데이터 set 활용

RGB (145,128,186) 원 색칠



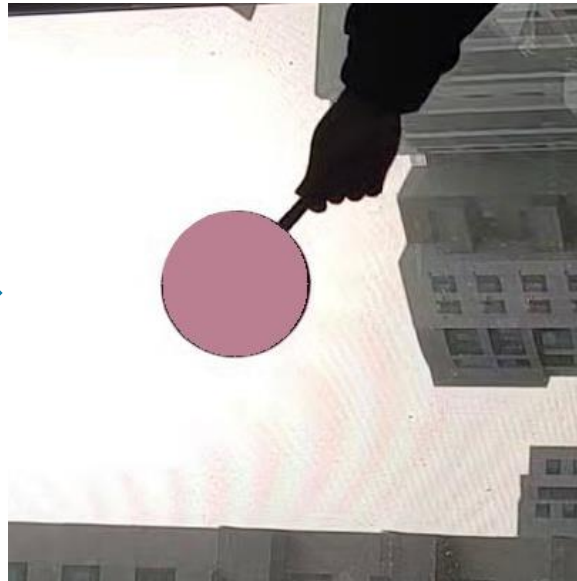
HSV -> Trackbars로 찾음.

purple\_lower = (158, 46, 124)

purple\_upper = (255, 163, 206)



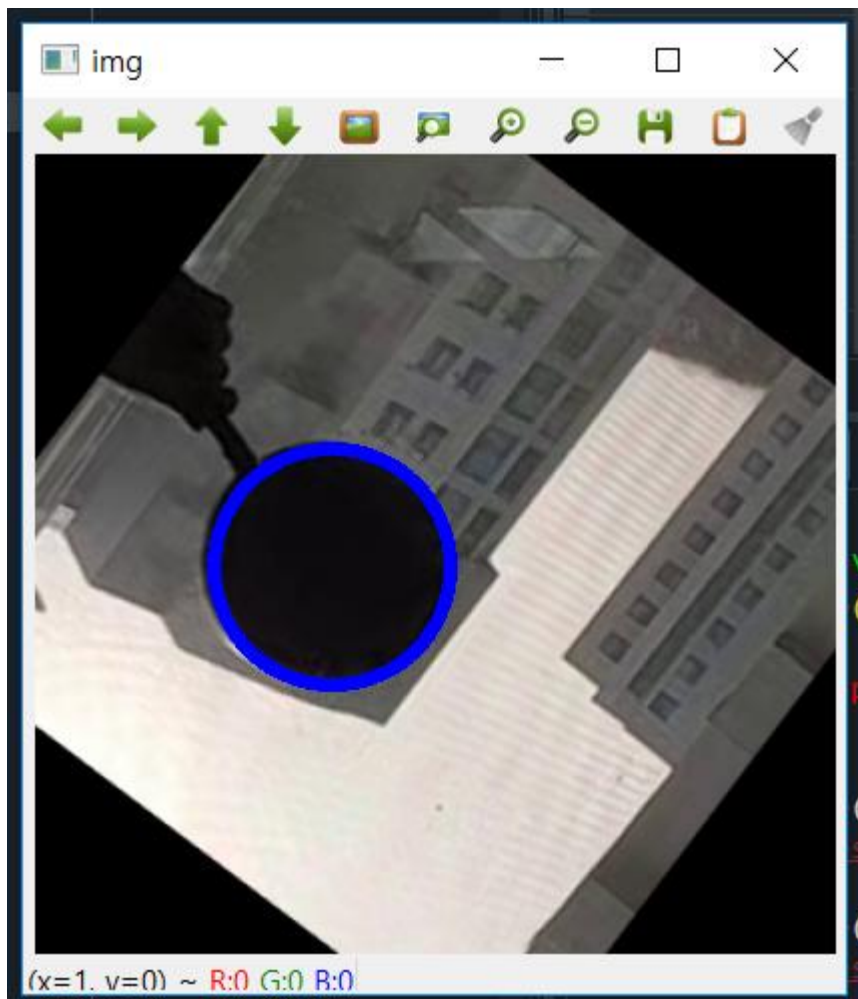
Target Contour 하여  
hsv tracking을 통해  
좌표 값 찾음.



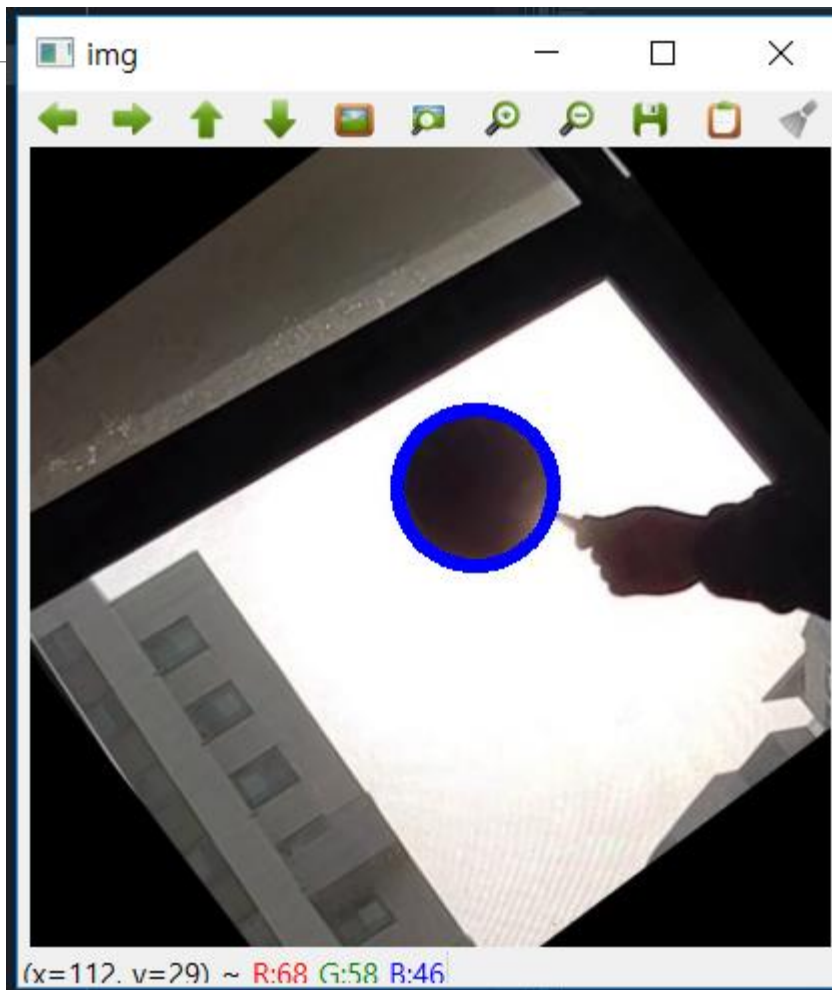
x	y	r	img
424	287	51	1.jpg
369	220	49	2.jpg
326	156	51	3.jpg
307	108	51	4.jpg
323	116	51	5.jpg
381	164	50	6.jpg
457	218	49	7.jpg
540	274	50	8.jpg
608	320	49	9.jpg
595	367	50	10.jpg
502	393	50	11.jpg
399	384	50	12.jpg
324	355	49	13.jpg
295	352	51	14.jpg
303	390	51	15.jpg

## 좌표 변환

147.6673 205.8182 59.25689 233.jpg

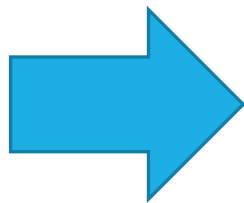
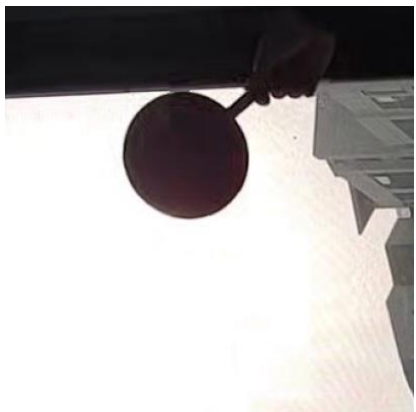


222.44 169.8586 39.40926 387.jpg

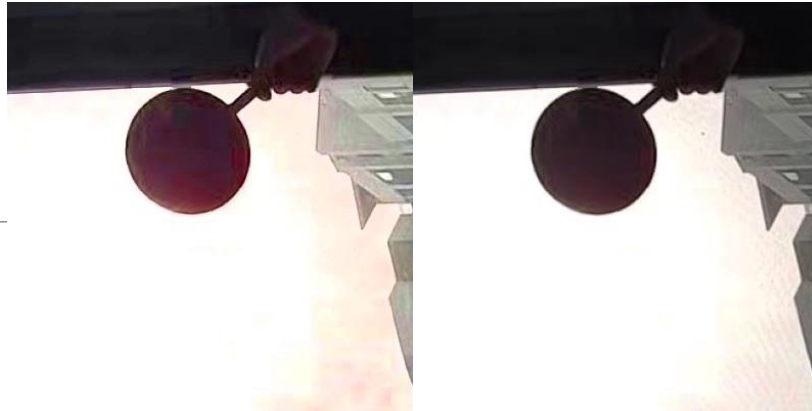


## Data Augmentation

crop

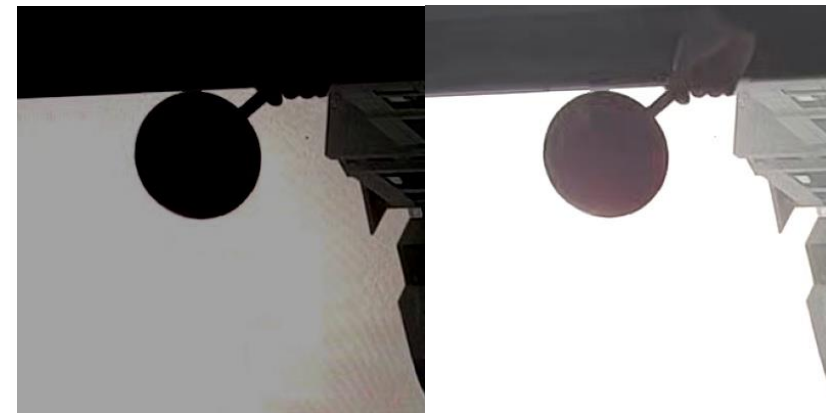


s,v만 변경(2개)

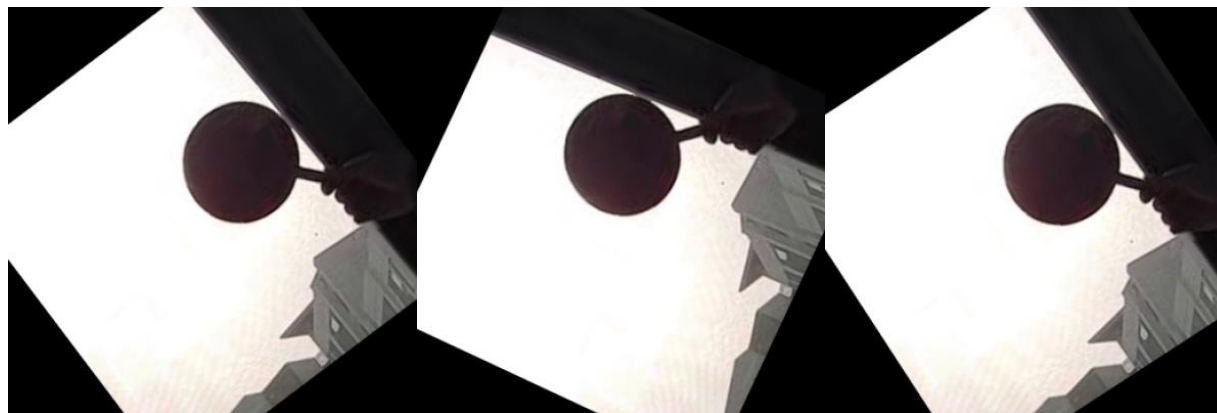


랜덤한 값 할당

R,G,B 변경(2개)



Rotation(3개)



73개의 image



1648장 augmentation