

개 물림 사고 예방을 위한 강아지 분류 프로그램

견주의 에리켓 점수는?

지금 공개합니다!

<16조 구성원>

팀장 - 이현민

팀원 - 김성진



목차

Table of Contents

01

<PROJECT> 의
Background

- 일상에서부터 시작하자!

02

DataSet 수집 방법과
파일 구조

- 데이터 수집
- 파일 구조

03

Building Model
How? Why?

- YOLOv4
- MobileNetV2

04

Visualization and
Result

- Flow chart
- 시연 영상

05

Improvement point,
Self - Feedback

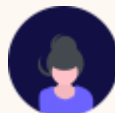
- 개선할 점
- 느낀점

CHAPTER. 1

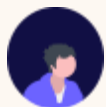
일상에서 찾은 issue가 Project로

<PROJECT> 의 Background

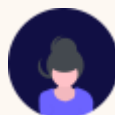
이어오빠~! 나 어제 퇴근 길에
목줄 없는 개랑 마주쳤었어~!!



헐 정말!? 몸은 괜찮아 드림아!?
어디 다친 곳은 없고!?



어 다행이 피해갔는데 하마터면
진짜 큰 일 날 뻔했어~!!





NEWS

요즘뜨 ISSUE

1. CHAPTER
일상에서 찾은
issue가 Project로



영상 출처((비디오머그)

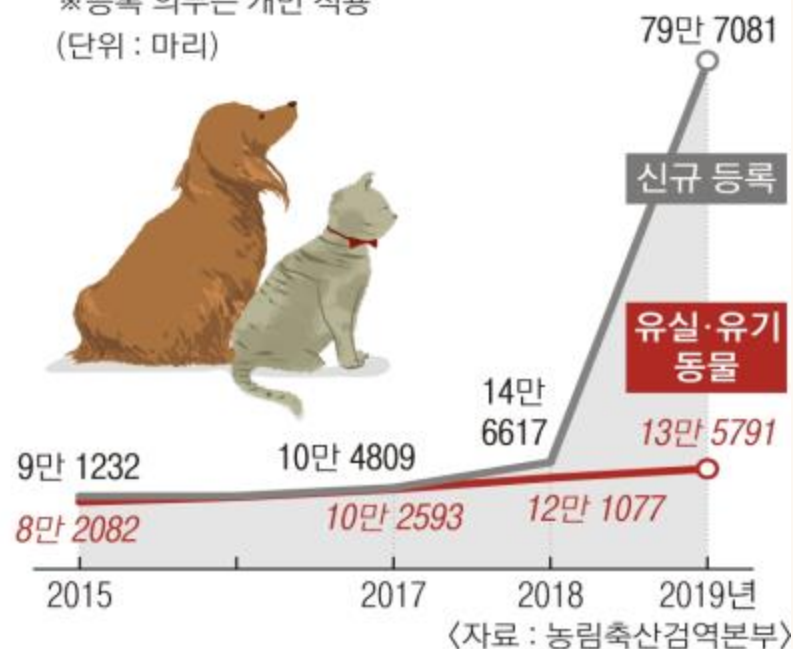
: https://www.youtube.com/watch?v=YFth8h0fsmY&ab_channel=%EB%B9%84%EB%94%94%EC%98%A4%EB%A8%B8%EA%B7%B8-VIDEOMUG

매년 반려견과 유실·유기견이 증가하는 추세

반려동물 신규 등록 및 유실·유기 현황

※등록 의무는 개만 적용

(단위 : 마리)



날이 풀린 달에 가장 사고가 많이 발생

개 물림 사고 많은 달은?

자료 : 소방청 (개 물림 사고 환자 이송 현황) 단위 : 건 (2020년 기준)



최근 잦은 개 물림 사건으로 시민들의 불안 고조

개 물림 사고로 이송된 환자 수

자료 : 소방청

하루 6명꼴



출처 <https://www.yra.co.kr/view/MYH20190904002100038>(연합뉴스)



무서워 못 나갈듯



처벌 강화를..



YearDream3일전

요새 목줄 없이 외출하는 사람들이 너무 많음..

👍 312 🗨️

개파라치들 화이팅!



견주들만 고생..



풍듀르르 3일전

1:14 내 개는 안 그래라는 생각부터 버려야함

👍 854 🗨️

1. CHAPTER
일상에서 찾은
issue가 Project로

반려견과 신혼여행 왔다가...반복되는 개 물림



CHAPTER. 2

Dataset 수집 방법과 파일 구조

구글 크롤링

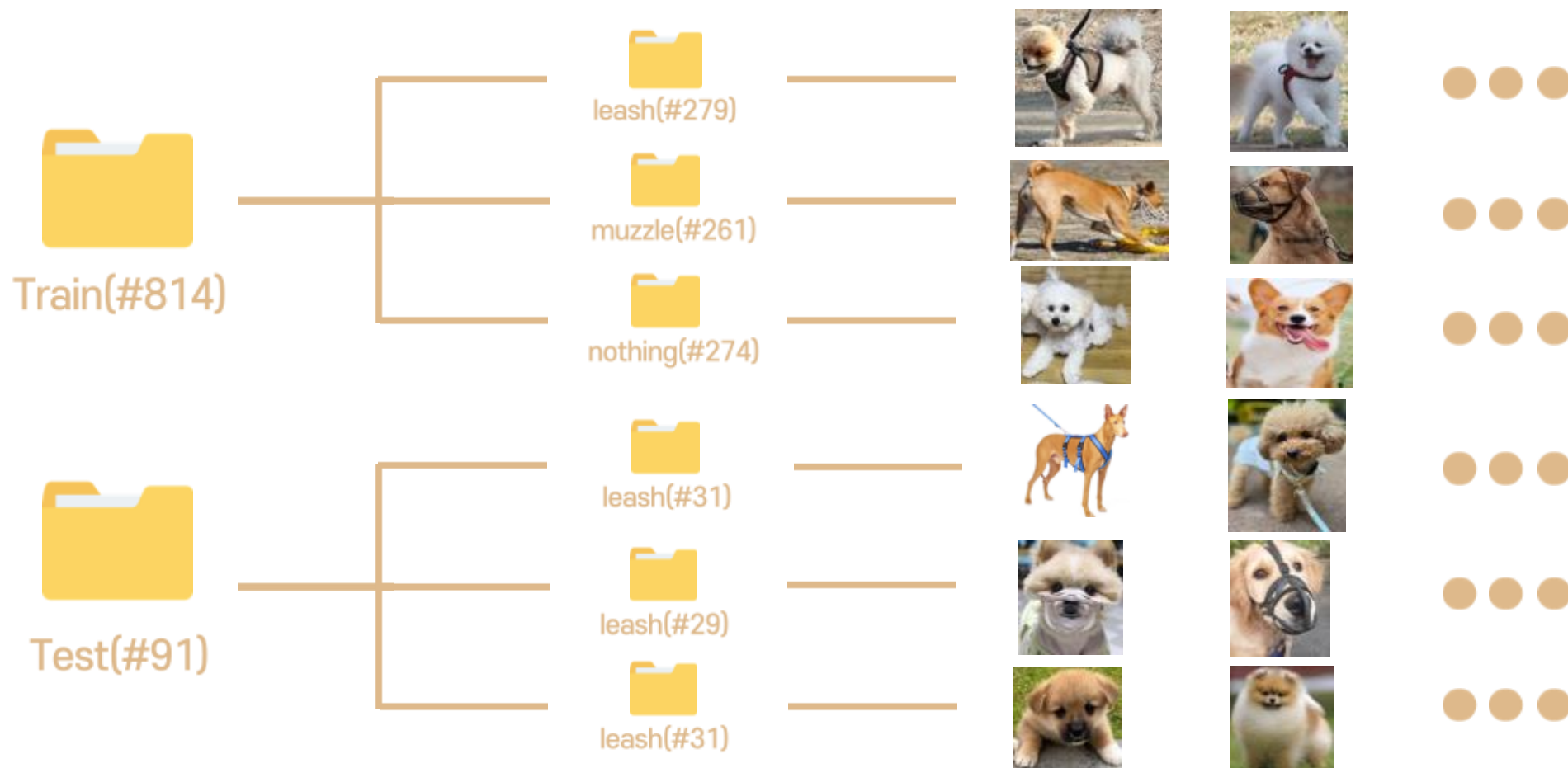


Instalooter



약 2000장

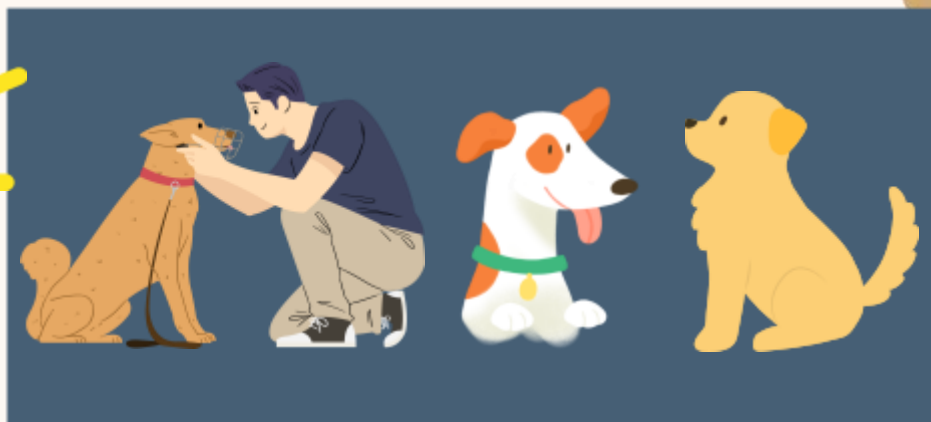
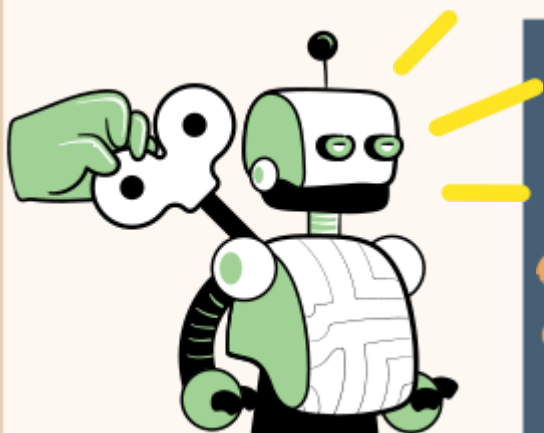
2. CHAPTER Dataset 수집 방법과 파일 구조



CHAPTER. 3

Modeling에 모든것 A to Z

Building Model How? Why?



3-1. YOLOv4 Detection and extract

Image를 YOLOv4의 Input data로 넣어준다.



이때 Image에서 강아지만 detect되도록
utils.py에서 코드 수정 필요



이미지에서 강아지 좌표값이 나오면 그 부분만
cropping 해서 저장



Source cord

10100
00101
00110

```
yolo_model = YOLOv4(weight_path='./yolov4.weights',  
                      class_name_path='./class_names/coco_classes.txt')  
file_list = os.listdir(location)  
for i in range(len(file_list)):  
    df = model.predict(location + file_list[i], random_color=True)  
    #강아지만 검출되도록 수정  
    idx = list(df[df.class_name != 'dog'].index)  
    df.drop(index=idx, inplace=True)  
    #강아지 image만 cropping하기  
    for j in range(len(df)):  
        dog_img = frame[int(df.iloc[j].y1) : int(df.iloc[j].y2),  
                        int(df.iloc[j].x1) : int(df.iloc[j].x2), :]
```

3-2. Image_Augmentation

수집한 데이터가 충분하지 못 해,
DATA_Augmentation을 진행

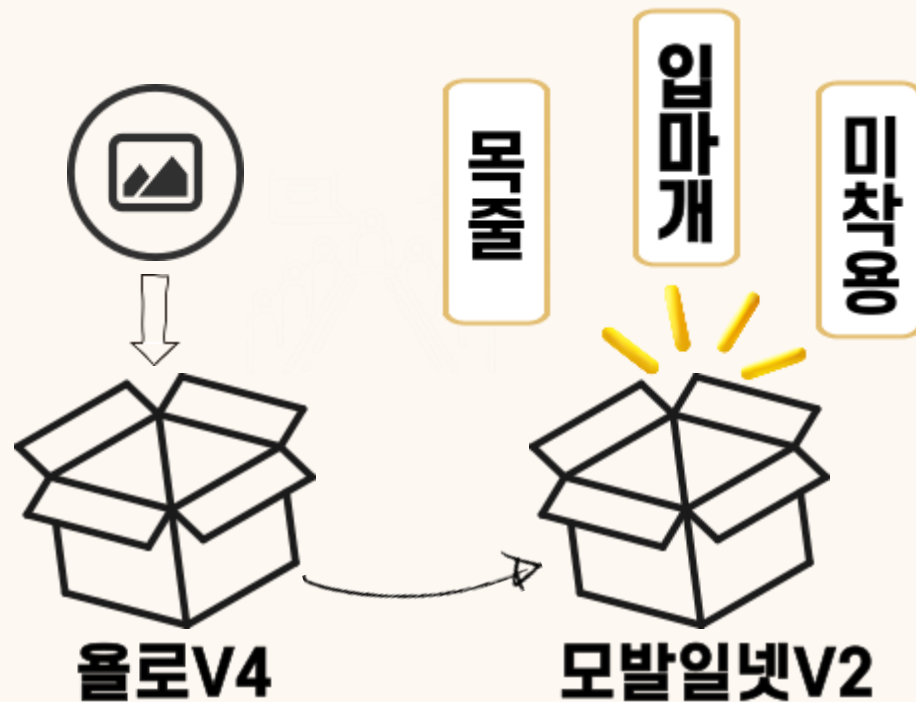
```
DATAGEN_TRAIN = ImageDataGenerator(  
    rescale=1./255,  
    rotation_range=20,  
    width_shift_range=0.2,  
    height_shift_range=0.2,  
    shear_range=0.2,  
    zoom_range=0.2,  
    horizontal_flip=True,  
    vertical_flip=True,  
    featurewise_center=True,  
    featurewise_std_normalization=True,  
    data_format="channels_last",  
    validation_split=0.10) # Train / Validation
```



```
DATAGEN_TEST = ImageDataGenerator(  
    rescale=1./255,  
    featurewise_center=True,  
    featurewise_std_normalization=True,  
    data_format="channels_last")
```



3-3 MobileNetV2 사용 및 이유



Source cord

```
batch_size = 32
epochs = 150

LearningRate = 1e-3 # 0.001
Decay = 1e-6

img_width = 224
img_height = 224
```

```
MobileNetV2Model= tf.keras.applications.MobileNetV2(
    input_shape=(img_width,img_height,3),
    include_top=False,
    weights="imagenet",
    input_tensor=None,
    pooling=None)
x = GlobalAveragePooling2D()(MobileNetV2Model.output)
x = Flatten()(x)
x = Dense(256, activation="relu")(x)
predictions = Dense(3, activation='sigmoid', name = "output_node")(x)
# Input ~ Output 연결해주기
DeepLearning = Model(inputs=MobileNetV2Model.input, outputs=predictions)
```

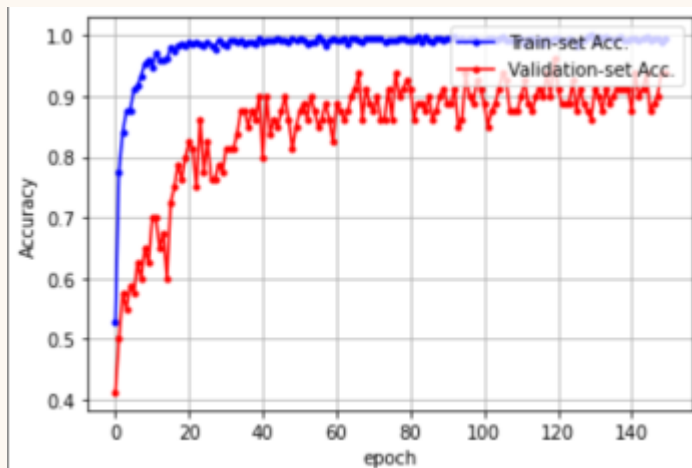
Model	Size (MB)	Top-1 Accuracy	Top-5 Accuracy	Parameters	Depth	Time (ms) per inference step (CPU)	Time (ms) per inference step (GPU)
MobileNet	16	70.4%	89.5%	4.3M	55	22.6	3.4
MobileNetV2	14	71.3%	90.1%	<u>3.5M</u>	105	25.9	3.8



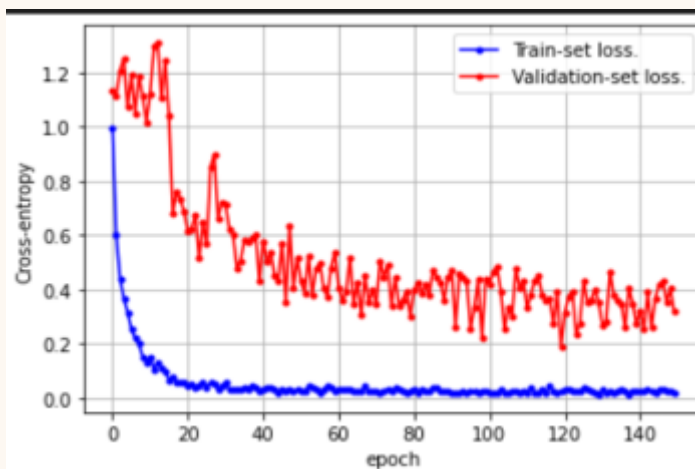
Source cord

3-4 MobileNetV2 성능 결과

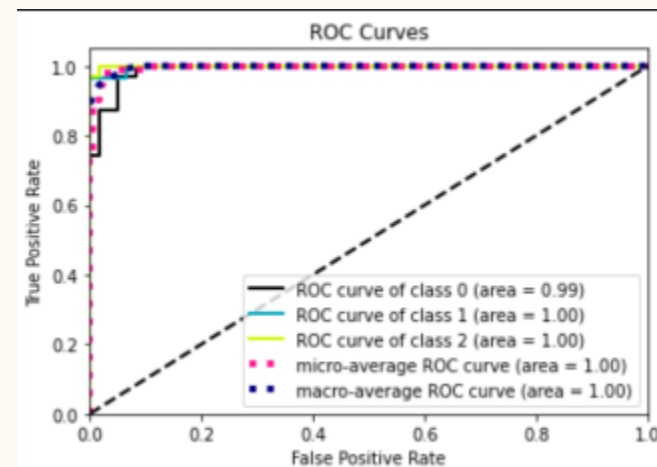
최종적으로 가장 잘 학습된 모델을 평가지표를
사용해서 측정해보았다.



Train/Val acc



Train/Val loss



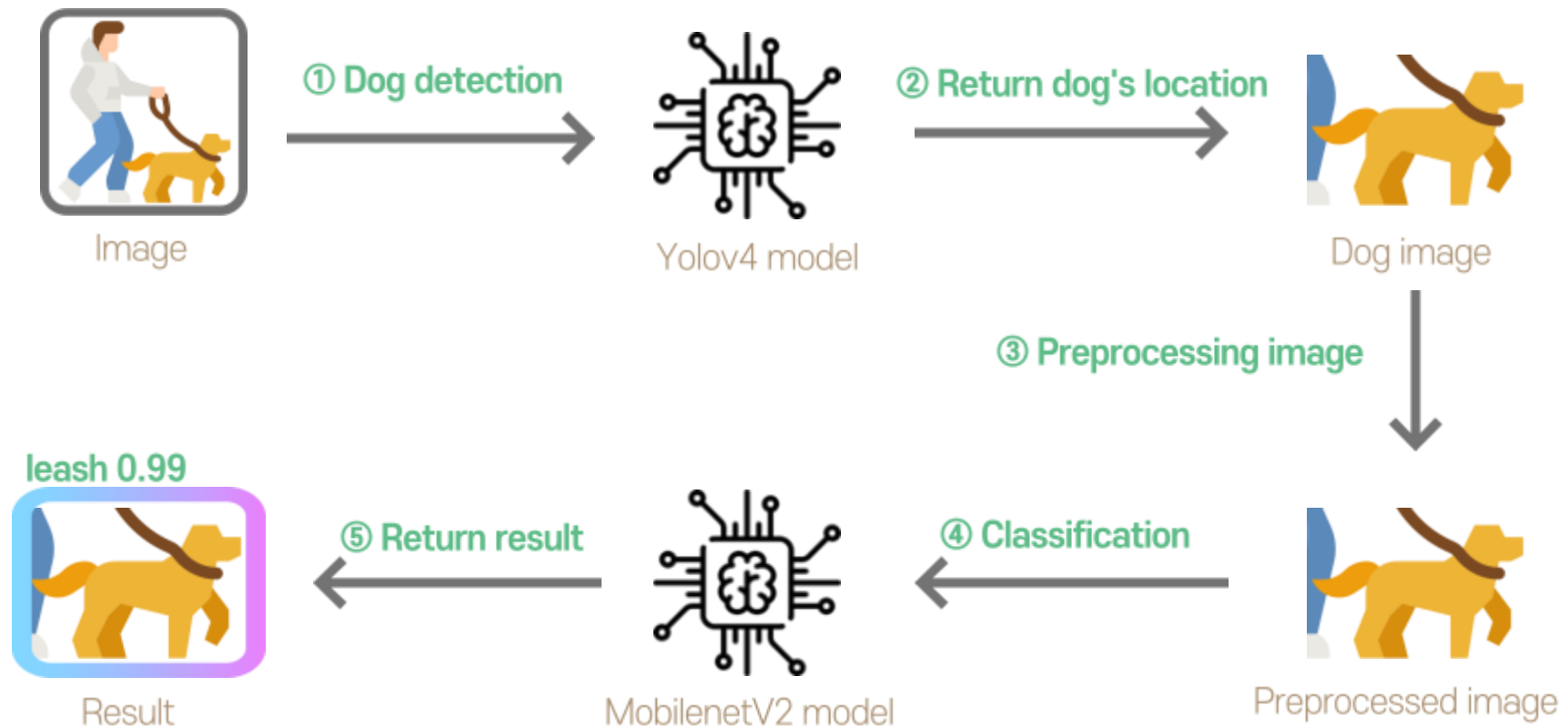
ROC Curves



CHAPTER. 4

Visualization and Result

Flow chart



① Dog detection

```
# get dog location in one frame
def get_dog_location(model, frame):
    df = model.predict_img(frame)

    # set dog score threshold
    df = df[df.score>0.5]
    return df
```

② Dog classification

```
def get_results(pred):
    if pred[0] > 0.7 and pred[1] > 0.7:
        class_txt = "leash_muzzle"
        score = (pred[0]+pred[1]) / 2

    else:
        class_num = pred.argmax(axis=-1)
        score = pred.max()

        if class_num == 0:
            class_txt = "leash"
        elif class_num == 1:
            class_txt = "muzzle"
        else:
            class_txt = "nothing"

    return score, class_txt
```

③ Draw bounding box

```
# draw bbox
scale = max(frame.shape[0:2]) / 416
line_width = int(2 * scale)

cv2.rectangle(frame, (int(df.iloc[i].x1), int(df.iloc[i].y1)), (int(df.iloc[i].x2),
                                                                int(df.iloc[i].y2)), color, line_width)

text = f'{class_txt} {score:.2f}'
font = cv2.FONT_HERSHEY_DUPLEX
font_scale = max(0.3 * scale, 0.3)
thickness = max(int(1 * scale), 1)

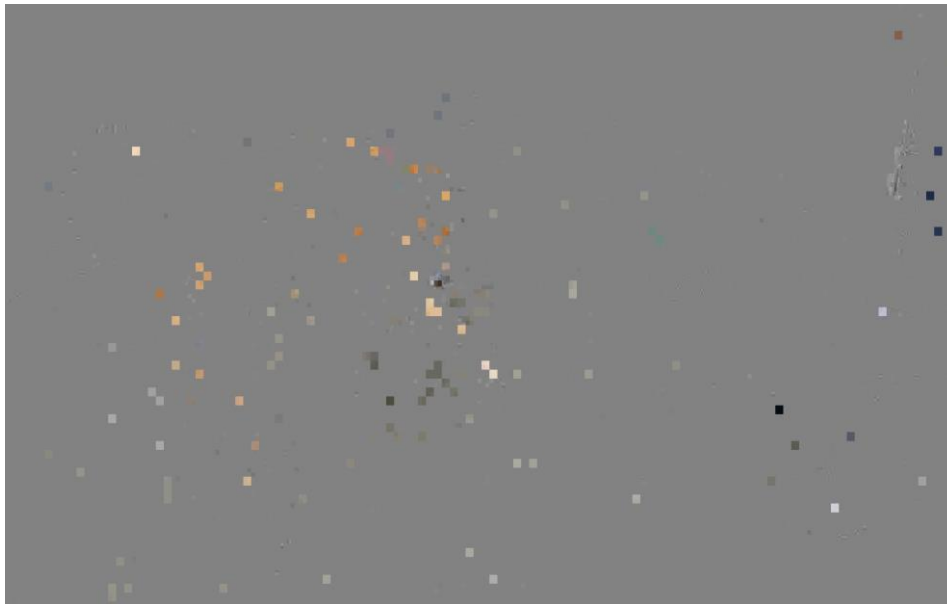
(text_width, text_height) = cv2.getTextSize(text, font,
                                             fontStyle=font_scale, thickness=thickness)[0]
cv2.rectangle(frame, (int(df.iloc[i].x1) - line_width//2,
                      int(df.iloc[i].y1) - text_height), (int(df.iloc[i].x1) + text_width,
                                                            int(df.iloc[i].y1)), color, cv2.FILLED)
cv2.putText(frame, text, (int(df.iloc[i].x1), int(df.iloc[i].y1)), font, font_scale,
             (255, 255, 255), thickness, cv2.LINE_AA)
```



<Sample result Image>

시연 영상

① 동영상



출처: <https://www.youtube.com/watch?v=b-6V8QS7YpU&t=249s>

[1] leash -> leash_muzzle

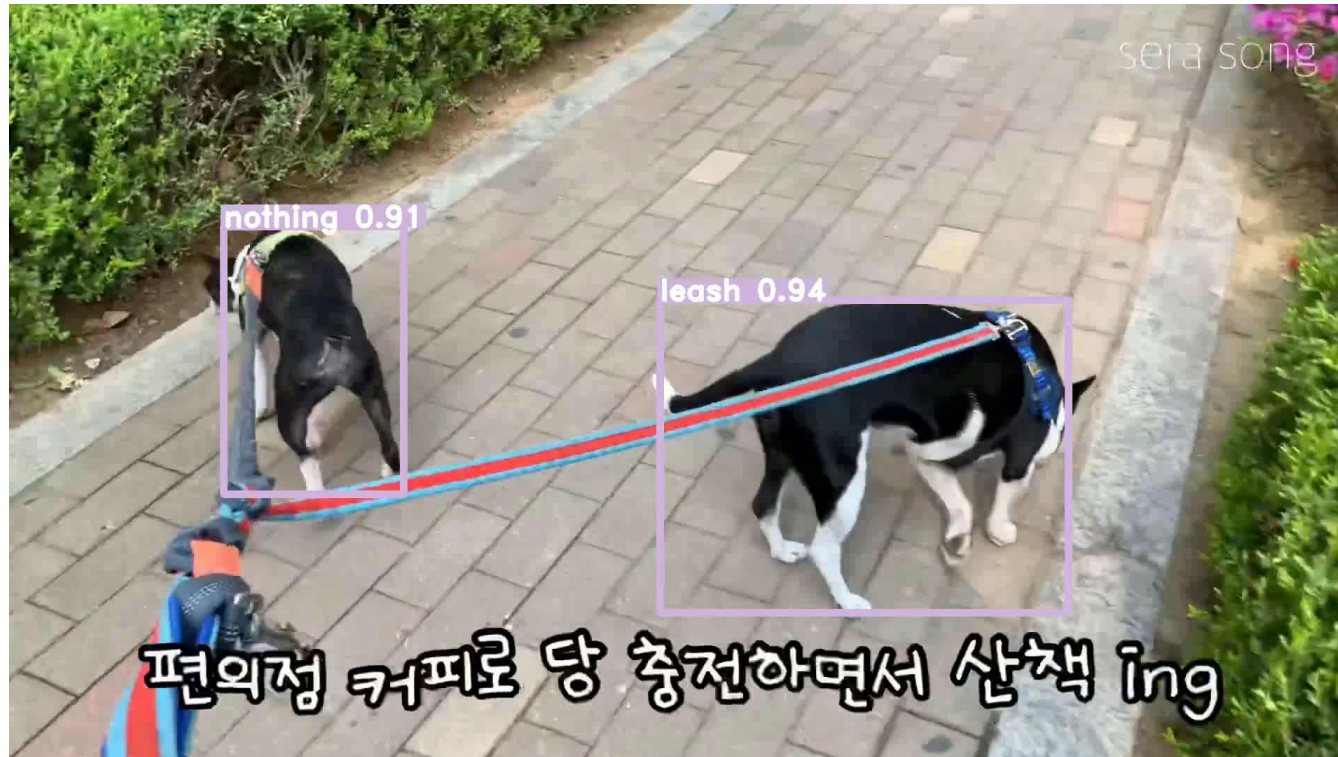


출처: <https://www.youtube.com/watch?v=cxCICB9MLA0&t=208s>

[2] 빠르게, 느리게 걷는 강아지

시연 영상

① 동영상

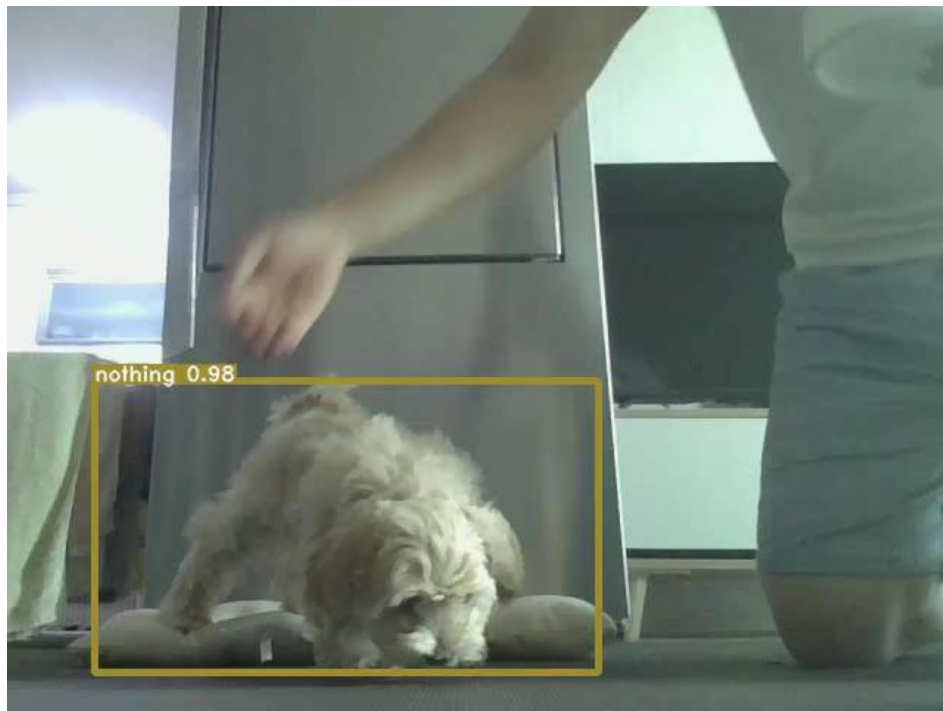


출처: <https://www.youtube.com/watch?v=xH9sN00c8GY&t=183s>

[3] 두 마리 이상의 강아지

시연 영상

① 웹캠(실시간)



[1] nothing



[2] leash

CHAPTER. 5

Improvement point & Self - Feedback





양질의 데이터셋

- 다양한 형태의 강아지 사진
- 해상도가 높은 이미지
- 보다 많은 데이터셋



이미지 분류 속도

- 실시간에서 보다 빠르게 분류할 수 있도록
flow 최적화 작업 필요



모델의 서비스화

- 실제로 CCTV에 연결
- "nothing"으로 나온 결과 값을 저장하고
관리할 수 있는 페이지 만들기



팀원의 중요성

- 팀원간 활발한 대화
- 같은 목표를 갖고 나아가는 팀원



코드 분석 능력의 중요성

- 구글링한 내용 적용
- YOLOv4 모델 커스텀



데이터 전처리의 중요성

- 영상의 화질
- Model에 맞게 이미지 전처리

감사합니다

