

Piano Keyboard Detection Experiments using YOLOv5

Aug 3rd, 2023

To, professor

| Index_

1. Datasets
2. Training model
3. Comparison of the results between models
4. Inference

| 1. Datasets

1. Datasets

There are 2,605 images for model training.

First, Dataset was split at a ratio 6:2:2 by sckit learn for model training.

type	train	validation	test
keyboard	1563	521	521

```
# train_test_split
from sklearn.model_selection import train_test_split # train, validation split
train_paths, test_paths = train_test_split(image_paths, test_size=0.2, random_state=42)
train_paths, val_paths = train_test_split(train_paths, test_size=0.25, random_state=42)
```

훈련 데이터 개수 : 1563
검증 데이터 개수 : 521
테스트 데이터 개수 : 521
2605

1. Datasets

Second, I moved image files and labeled files to the appropriate path by coded.

```
# 파일 옮기기
import shutil
import os

def move_files_to_destination(file_paths, destination_folder):
    for file_path in file_paths:
        filename = os.path.basename(file_path)
        destination_path = os.path.join(destination_folder, filename)
        shutil.move(file_path, destination_path)

# 이동할 폴더 경로 설정
train_destination = './keyboard/images/train'
val_destination = './keyboard/images/val'
test_destination = './keyboard/images/test'

# 훈련 데이터 이동
move_files_to_destination(train_paths, train_destination)
# 검증 데이터 이동
move_files_to_destination(val_paths, val_destination)
# 테스트 데이터 이동
move_files_to_destination(test_paths, test_destination)

print("파일 이동이 완료되었습니다.")
```

```
import shutil
import os

def move_files_with_matching_names(source_folder, destination_folder, file_list):
    # destination_folder가 존재하지 않으면 생성
    os.makedirs(destination_folder, exist_ok=True)

    for file_name in file_list:
        source_path = os.path.join(source_folder, file_name)
        destination_path = os.path.join(destination_folder, file_name)

        # 파일 이름이 일치하면 이동
        if os.path.exists(source_path):
            shutil.move(source_path, destination_path)

# 폴더 경로 설정
source = './keyboard/labels/label'
train_destination = './keyboard/labels/train'
val_destination = './keyboard/labels/val'
test_destination = './keyboard/labels/test'

# 훈련 데이터 이동
move_files_with_matching_names(source, train_destination, [os.path.basename(path) for path in label_paths if os.path.basename(path) == 'train'])
# 검증 데이터 이동
move_files_with_matching_names(source, val_destination, [os.path.basename(path) for path in label_paths if os.path.basename(path) == 'val'])
# 테스트 데이터 이동
move_files_with_matching_names(source, test_destination, [os.path.basename(path) for path in label_paths if os.path.basename(path) == 'test'])

print("파일 이동이 완료되었습니다.")
```

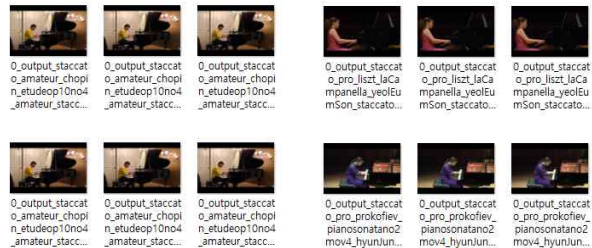
파일 이동이 완료되었습니다.

1. Datasets

- Directories

YOLOV5 > C3Pap_openpose > keyboard > images	
이름	수정된 날짜
test	2023-08-03 오후
train	2023-08-03 오후
val	2023-08-03 오후

YOLOV5 > C3Pap_openpose > keyboard > images > train



YOLOV5 > C3Pap_openpose > keyboard > labels	
이름	수정된 날짜
test	2023-
train	2023-
val	2023-

YOLOV5 > C3Pap_openpose > keyboard > labels > train

이름	수정된 날짜
0_output_staccato_amateur_chopin_etud...	2023-08-03
0_output_staccato_amateur_chopin_etud...	2023-08-03
0_output_staccato_amateur_chopin_etud...	2023-08-03

YOLOV5 > C3Pap_openpose > keyboard > labels > val

이름	수정된 날짜
1_output_staccato_amateur_chopin_etud...	2023-08-03
1_output_staccato_amateur_chopin_etud...	2023-08-03
1_output_staccato_amateur_liszt_laCam...	2023-08-03

| 2. Training model

2. Training model

The hyperparameters used to train the model are:

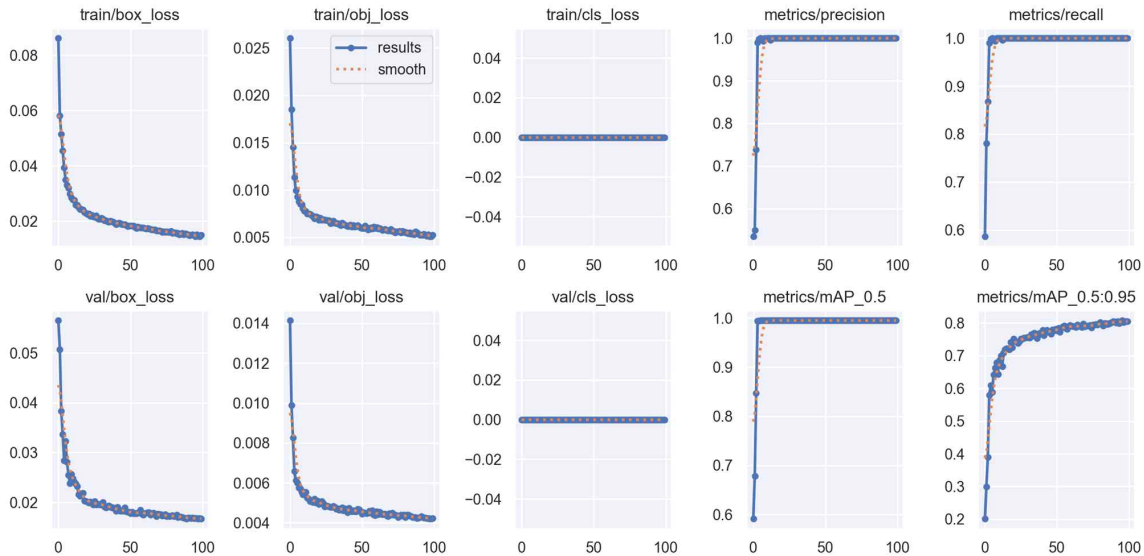
1. `--batch 16 --epochs 100 --weights yolov5s.pt`
2. `--batch 16 --epochs 50 --weights yolov5s.pt`
3. `--batch 16 --epochs 500 --weights yolov5s.pt` (However, the training ended at the 368th training.)

In summary, only the number of epochs had changes.

3. Comparison of the results between models

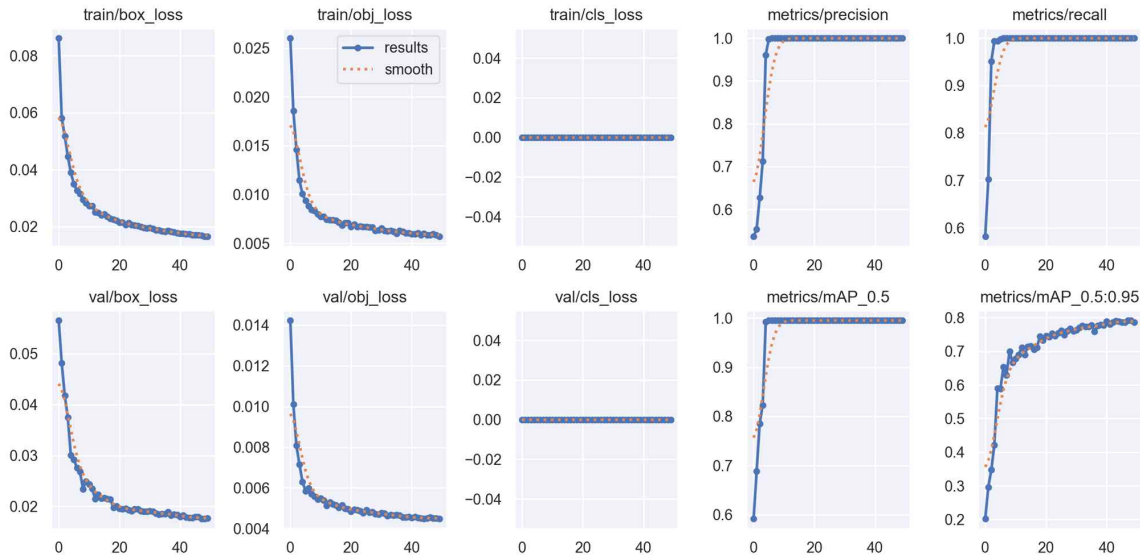
3. Comparison of the results between models

1. A model trained by epochs 100



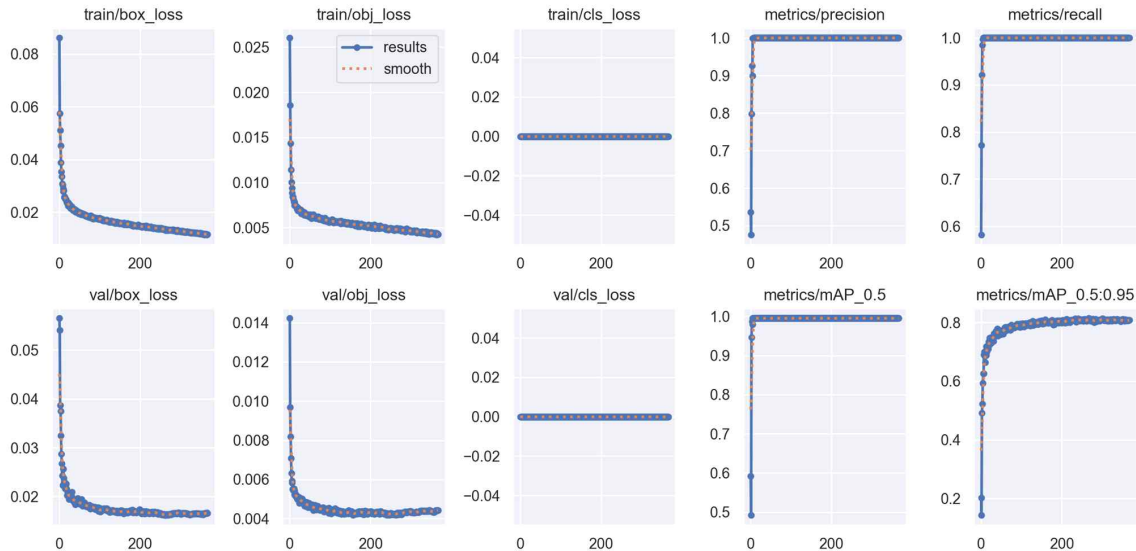
3. Comparison of the results between models

2. A model trained by epochs 50



3. Comparison of the results between models

3. A model trained by epochs 368



3. Comparison of the accuracy between models

1. A model trained by epochs 100

- precision: 0.999
- recall: 1
- mAP_0.5: 0.995
- mAP_0.5:0.95: 0.80534

2. A model trained by epochs 50

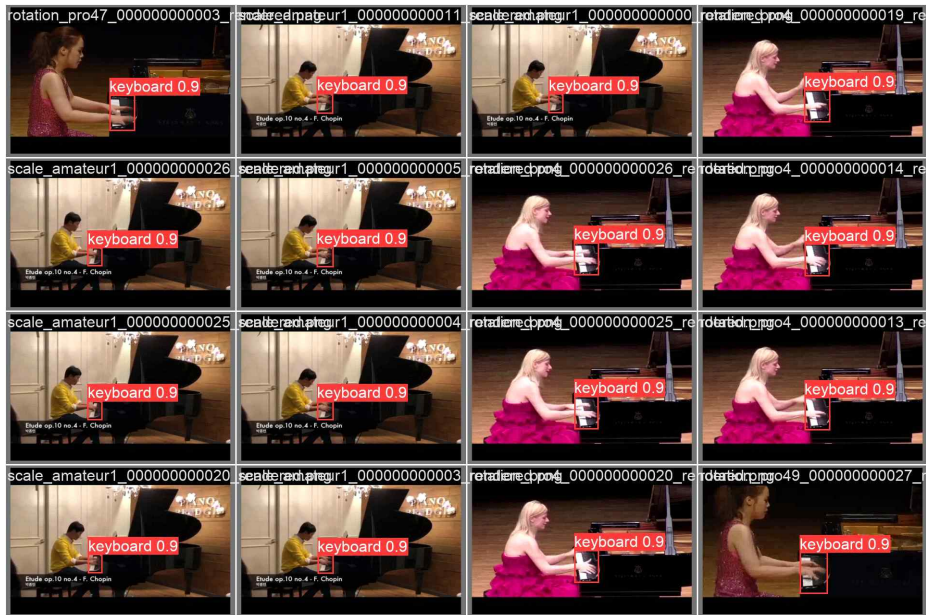
- precision: 0.999
- recall: 1
- mAP_0.5: 0.995
- mAP_0.5:0.95: 0.78612

3. A model trained by epochs 368

- precision: 0.999
- recall: 1
- mAP_0.5: 0.995
- mAP_0.5:0.95: 0.808

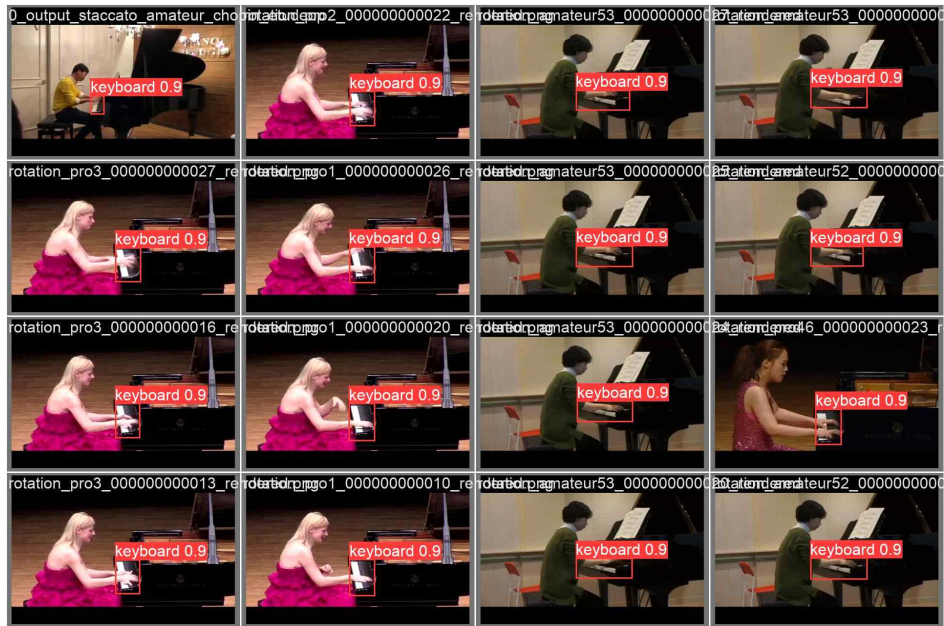
3. Comparison of the results between models

1. A model trained by epochs 100 – val



3. Comparison of the results between models

2. A model trained by epochs 50 – val



3. Comparison of the results between models

3. A model trained by epochs 368 – val



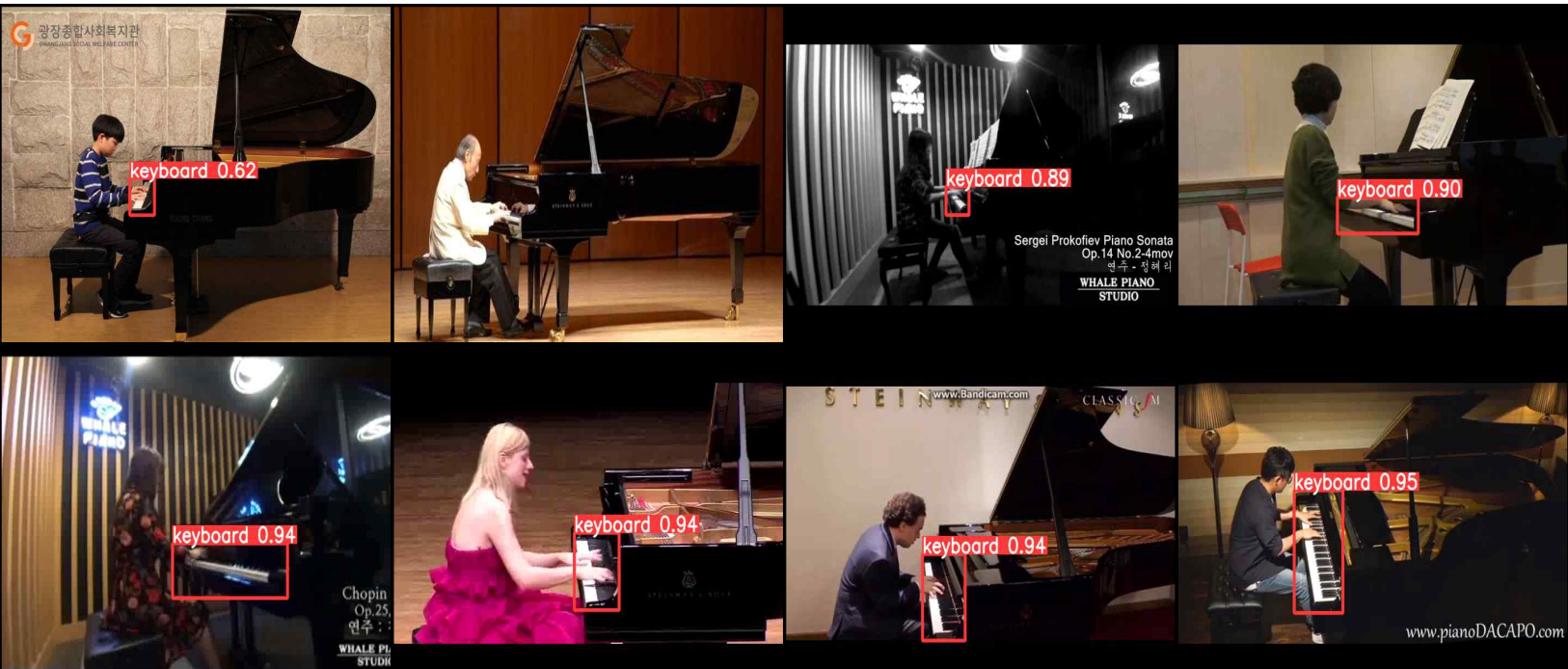
| 4. Inference

For the test, I conducted a test with 10 images already had and 10 images collected by searching for 'playing piano' on Google.

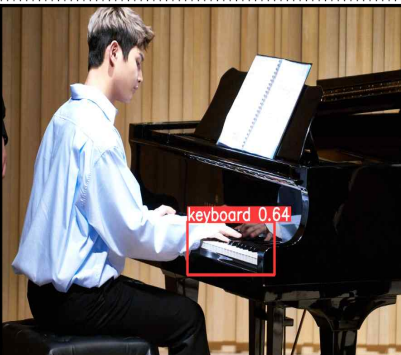
4. Inference – A model trained by epochs 100 (1)



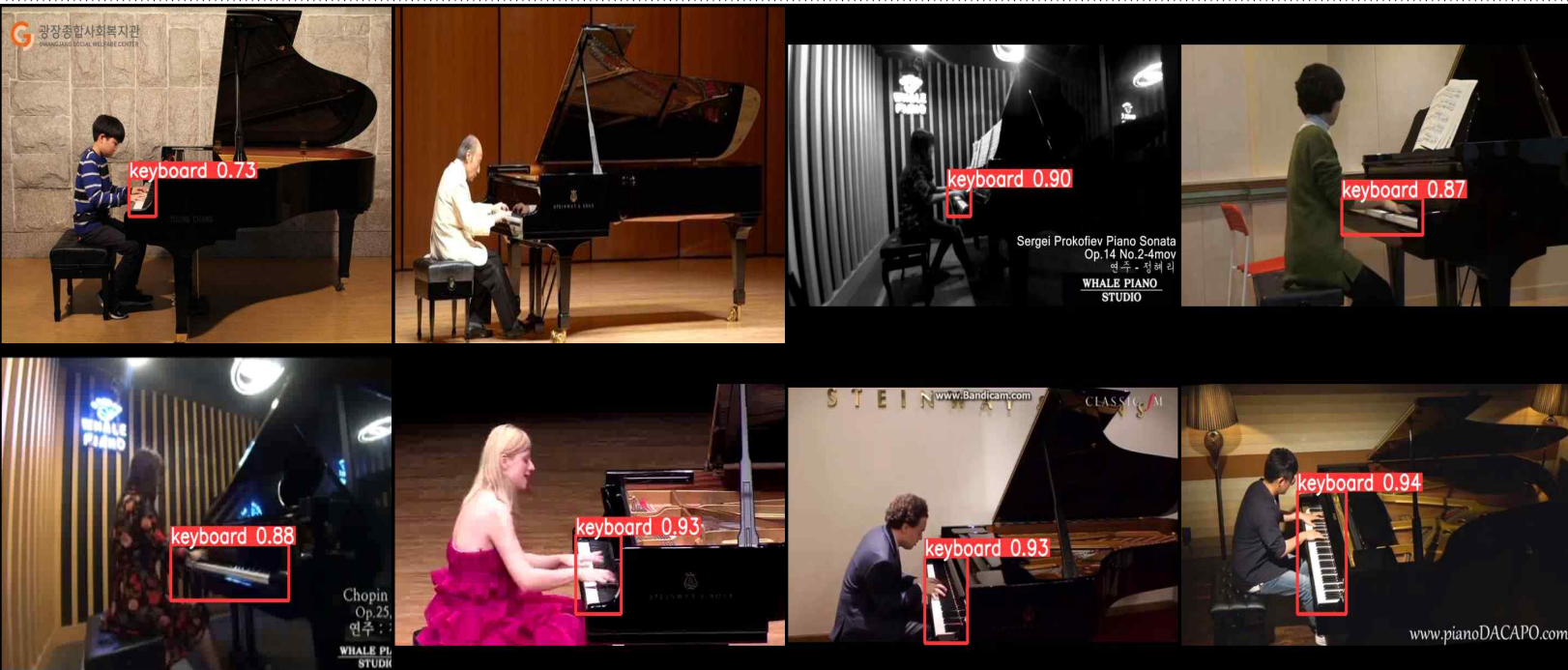
4. Inference – A model trained by epochs 100 (2)



4. Inference – A model trained by epochs 50 (1)



4. Inference – A model trained by epochs 50 (2)



4. Inference – A model trained by epochs 368 (1)



4. Inference – A model trained by epochs 368 (2)

