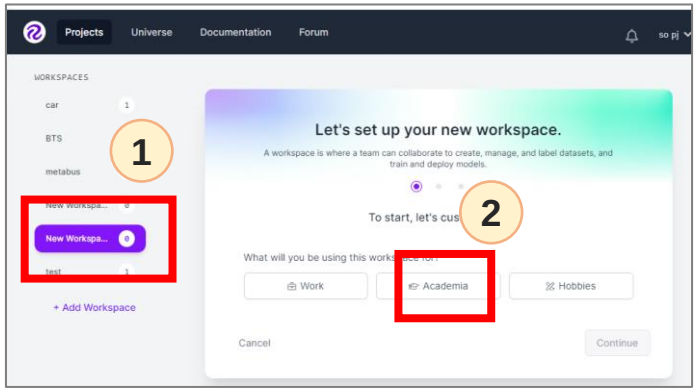
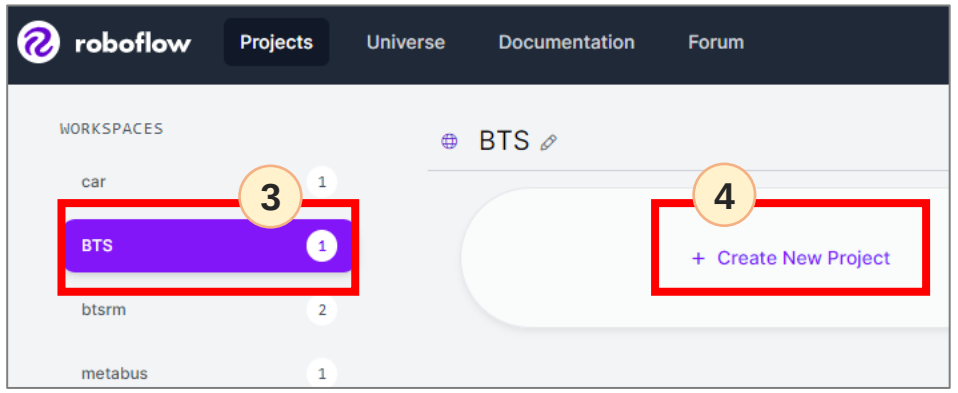
욜로가 미리 학습한 coco데이터셋이 아닌경우 - **Part1: 학습데이터 제작**

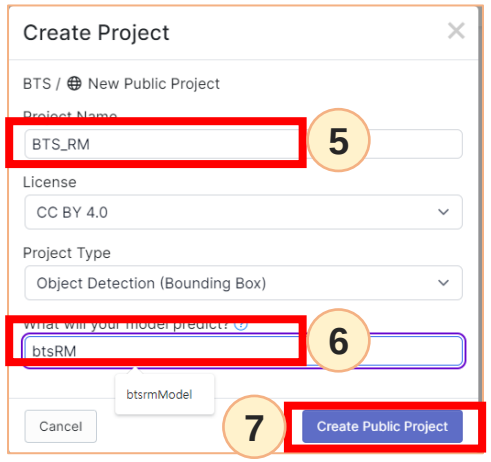
1. 로보플로우 사이트, 로그인, 워크스페이스 추가



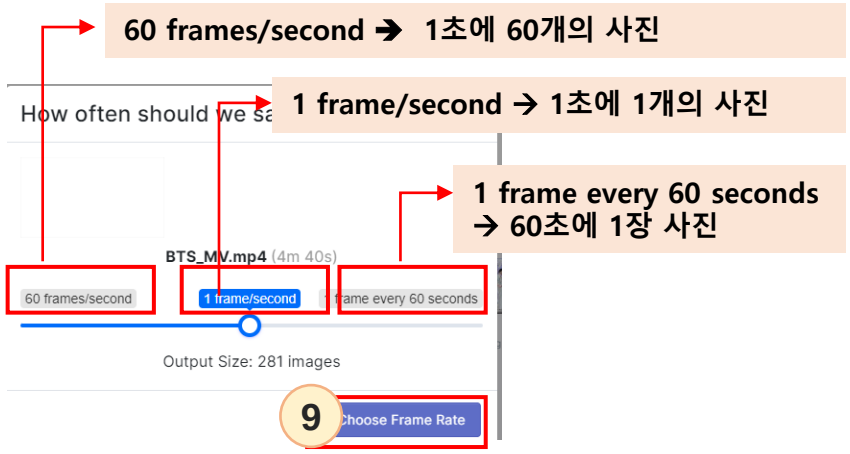
2. 추가된 워크스페이스에 NewProject 추가



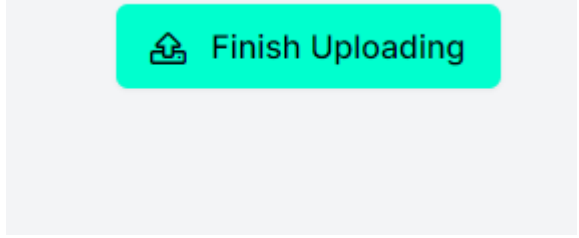
3. 프로젝트이름 및 모델이름추가



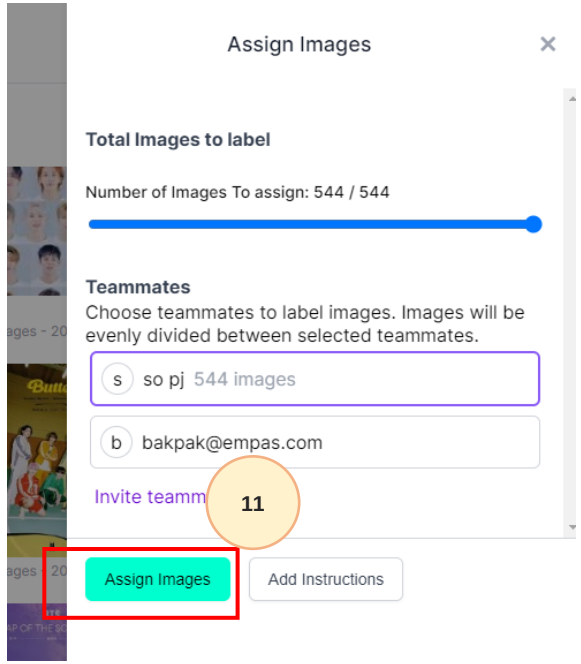
4. 동영상 추가시 8



5. 우측 상단의 Finish Uploading 10

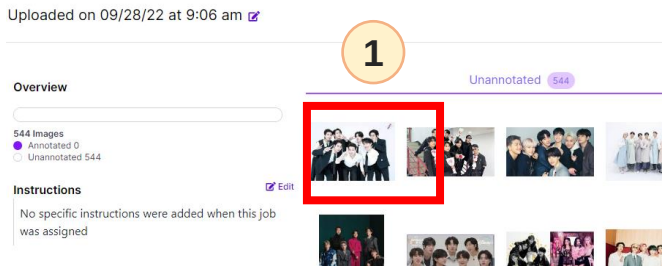


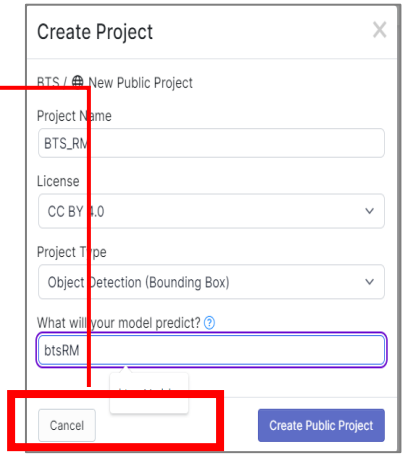
6. Assign images



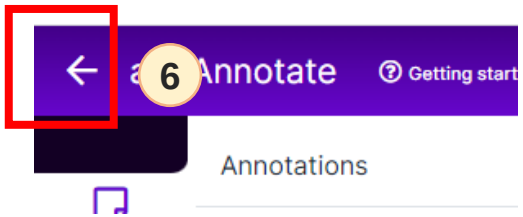
욜로가 미리 학습한 coco데이터셋이 아닌경우 - **Part1: 학습데이터 제작**

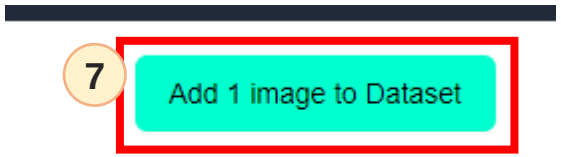
7. 이미지를 더블클릭해서 어노테이션 작업을 시작

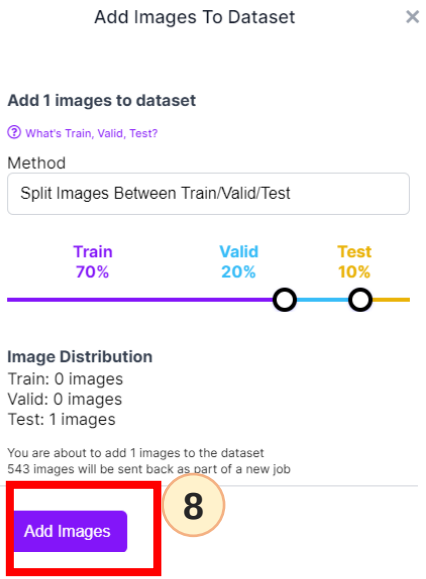




8. 박스 작업이 다 종료되면 왼쪽 상단의 ← 를 클릭(아래그림의 6번), 우측상단의 DataSet제작(아래그림의 7번)을 클릭함

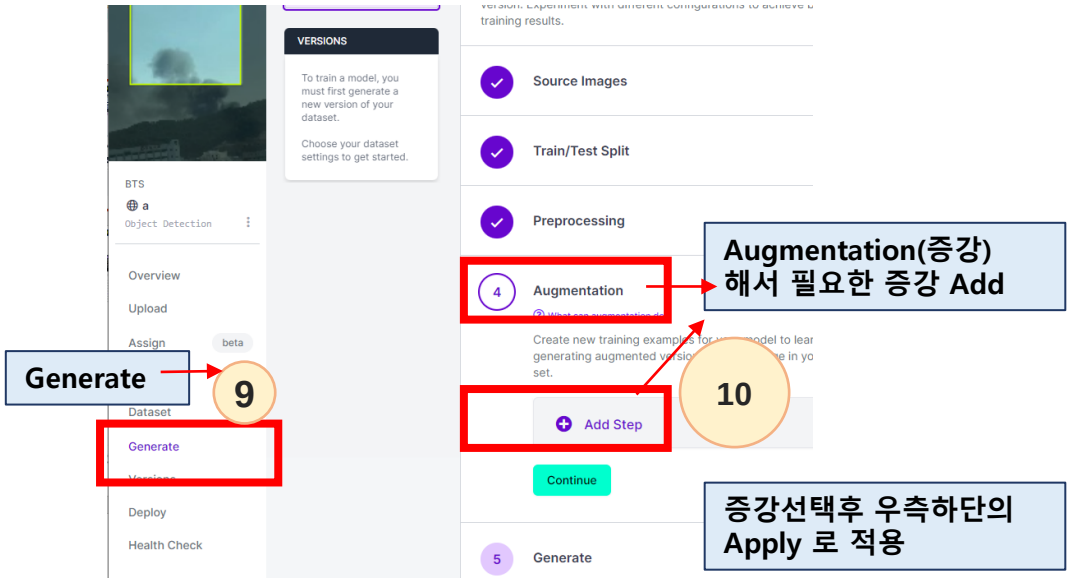




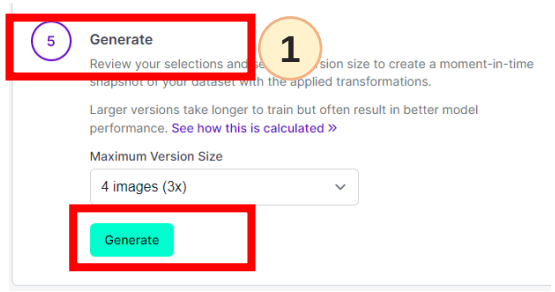


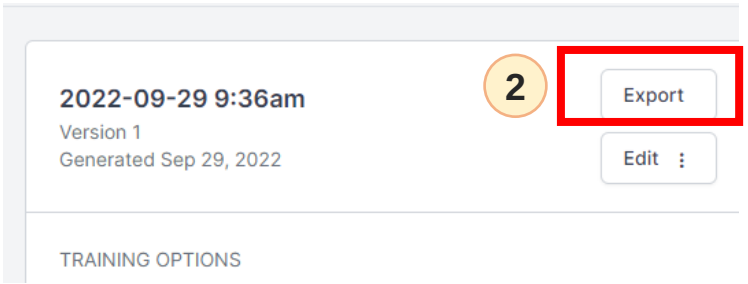
욜로가 미리 학습한 coco데이터셋이 아닌경우 - **Part1: 학습데이터 제작**

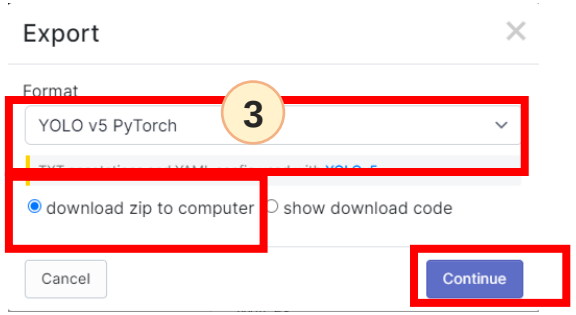
9. 훈련,검증,테스트 비율을 설정(일반적으로 제공되는 설정사용함) , 증강등 이미지 추가



10. (5)번의 Generate의 하단의 초록색 버튼 클릭하여 작업 마무리 후 [Export] 클릭하여 이미지와 바운딩 박스 처리한 레이블 내보내기
forma은 [Yolo v5 PyTorch]로 , 내 컴퓨터 다운로드함. (Show Download code 하여서 이 자료를 압축한 url 주소만 받을수도 있음. 여기서는 내컴으로 다운로드함)

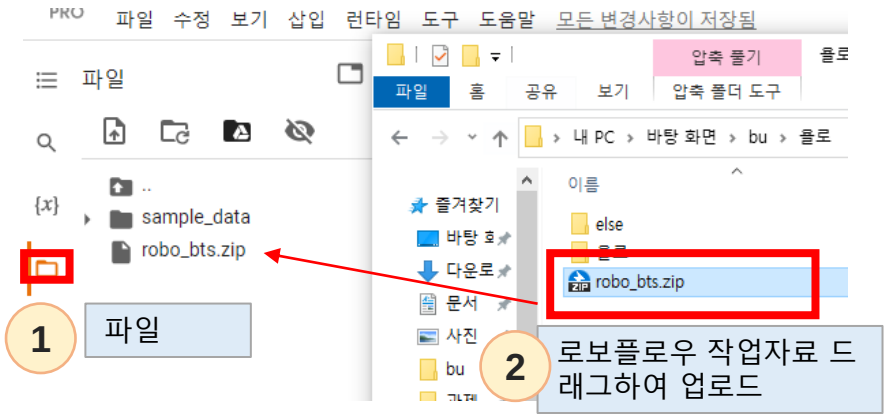






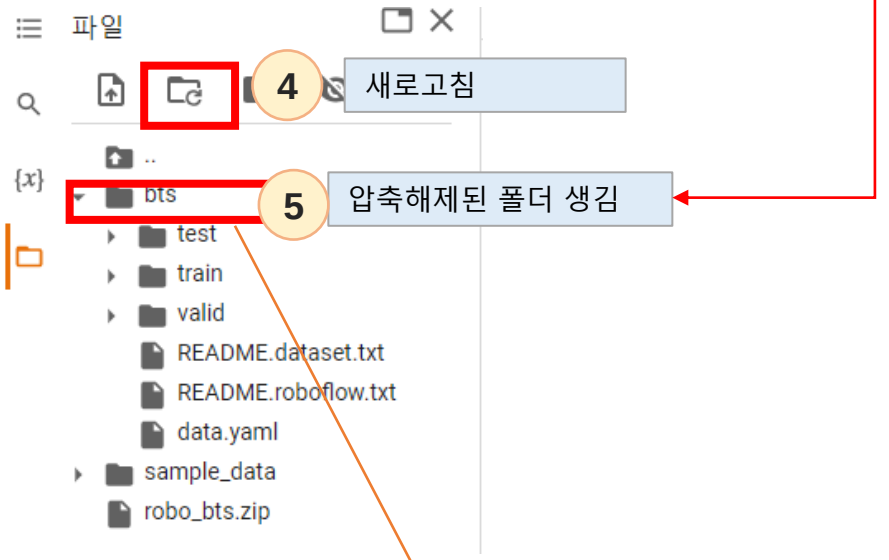
욜로가 미리 학습한 coco데이터셋이 아닌경우 – **Part2: 욜로트레이닝 하기, 내 데이터로 가중치 제작**

1. 다운로드한 자료를 작업할 코렙에 올리고, 기존의 욜로를 내 자료로 훈련시켜야 함.
코렙을 실행하고, [런타임-유형변경] 에서 GPU로 변경한다음에 로보플로우에서다운받은 zip파일을 드래그하여서 코렙에 올림.



2. zip파일을 압축해제함

3 Code `!unzip -uq "/content/robo_bts.zip" -d "/content/bts/"`



3. git에서 욜로 자료 다운로드함

Code
```
!git clone https://github.com/ultralytics/yolov5
```

4. 욜로 폴더로 이동한뒤 requirements.txt를 실행시켜 필요한 모듈을 다운로드받음

Code
```
%cd /content/yolov5/
!pip install -r requirements.txt
```

5. 로보플로우에서 작업한 이미지와 레이블이 있는 폴더명을 입력함
훈련과 검증, 테스트 데이터셋의 위치를 지정하는 작업임

Code
```
import yaml
folder='bts'
with open('/content/' + folder + '/data.yaml', 'r') as f:
  data = yaml.load(f,Loader=yaml.FullLoader)

print(data)

data['train'] = '/content/' + folder + '/'
data['test'] = '/content/' + folder + '/'
data['val'] = '/content/'+ folder + '/'

with open('/content/' + folder + '/data.yaml', 'w') as f:
  yaml.dump(data, f)

print(data)
```

폴더명만 변경함

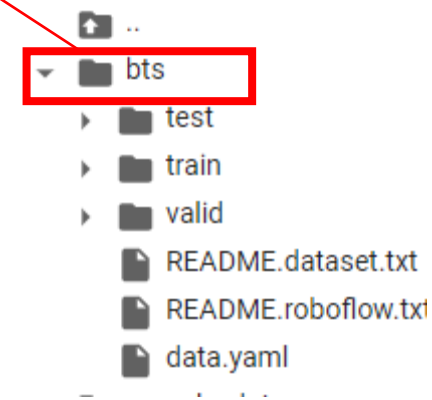
욜로가 미리 학습한 coco데이터셋이 아닌경우 – Part2: 욜로트레이닝 하기, 내 데이터로 가중치 제작

6. 실행 (주의: 반드시 데이터 경로가 있는 폴더명으로 변경해야함), 아래의 1,2,3 번은 사용자가 지정해야함.

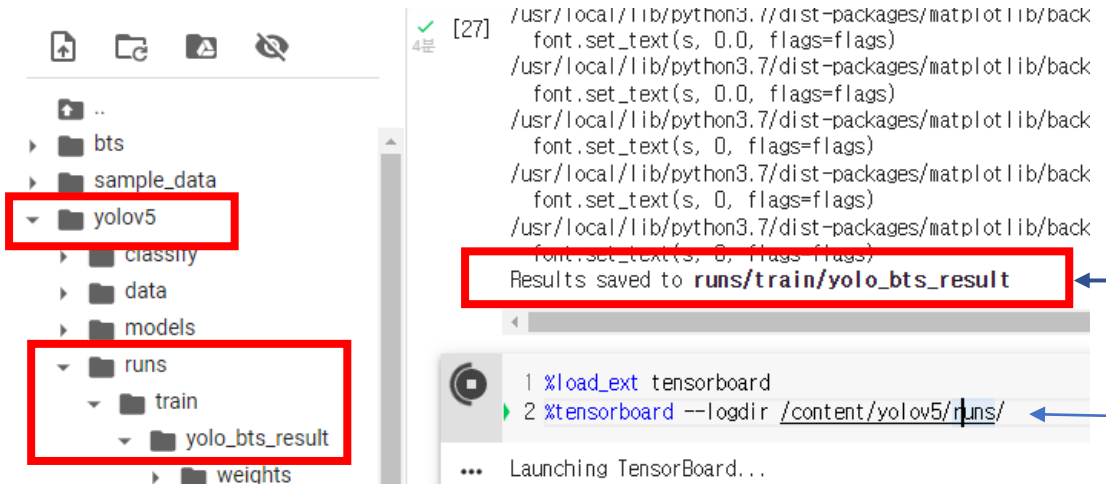
Code

```
!python train.py --img 416 --batch 16 --epochs 50 --data /content/bts/data.yaml --cfg ./models/yolov5s.yaml --weights yolov5s.pt --name yolo_bts_result
```

1 2 3



7. 훈련하여 새로 생성된 가중치값과 여러상황은 runs/train 에 사용자가 지정한 이름으로폴더가 생성됨
텐서보드를 이용하여 훈련의 결과를 확인할수 있음.



Code

```
%load_ext tensorboard
%tensorboard --logdir /content/yolov5/runs/
```

욜로가 미리 학습한 coco데이터셋이 아닌경우 – Part3. 학습된 가중치 사용하기

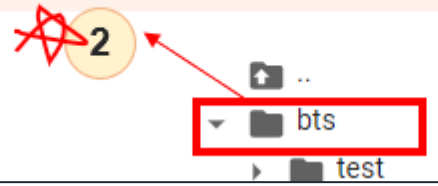
1. 샘플이미지를 업로드한뒤 가중치 경로를 설정하고 욜로프로그램을 실행해봄

Code

```
!python train.py --img 416 --batch 16 --epochs 50 --data /content/bts/data.yaml --cfg ./models/yolov5s.yaml --weights yolov5s.pt --name yolo_bts_result
```

1

2



3

슬라이드 5번에서 작업한 폴더명과 일치해야함.

Code

```
weights_path = '/content/yolov5/runs/train/yolo_bts_result/weights/best.pt'
!python detect.py --weights "{weights_path}" --img 416 --conf 0.1 --source /content/bts샘플.jpg
```