Qingling Duan, GitHub Link

Documentation

The solution I use is the object detection model, specifically Ultralytics YOLO. The solution will cover **data labeling, custom model training, and staff exclusion**. The programming language is **Python**. Software and devices I use: roboflow, google colab.

Data Labelling

Data Preprocessing: The video sample sample.mp4 was extracted into video frames, and a total of 1341 images were obtained for later data annotation. See codes/video2img.py for details.

To train a custom model on our own data set, we must annotate the data set to obtain reference labels for model training. In this case, the objects we want to mark are people (**sticking out their chests**) and name tags. The software I use is Roboflow, which can directly get the format required by the yolo model.

Suppose employees are required to wear name tags on their chests.

And sticking out your chest makes it easy to spot name tags.

For employees, sticking out their chests to expose as much of their upper body as possible, as well as facing the camera, are encouraged.

Employees should not face away from the camera, nor should they bury their heads on the desk, which can easily obscure the name tag.

For the label, I draw a bounding box to label it as ‘label’.

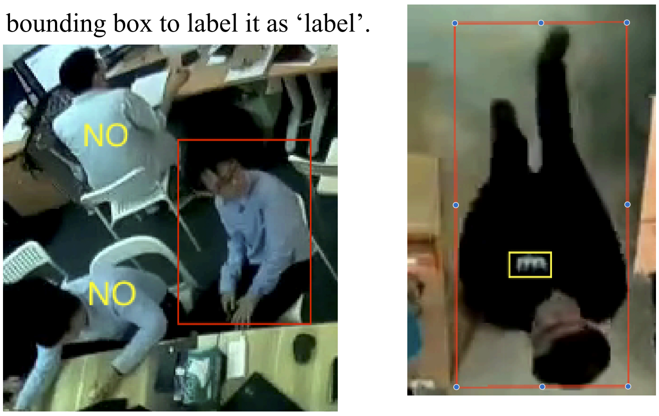
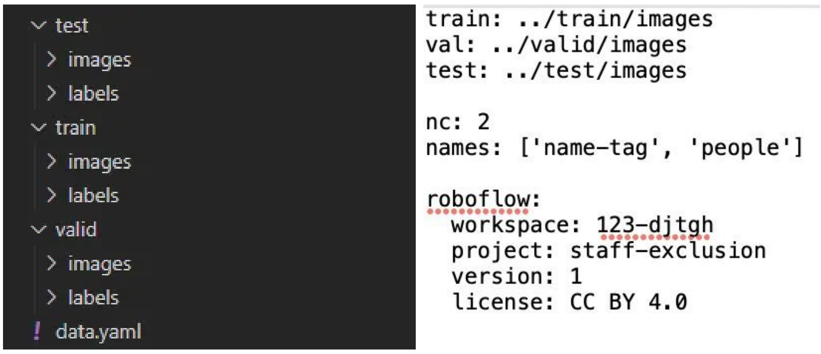


Figure 1: Label Labelling



```
test
  images
  labels
train
  images
  labels
valid
  images
  labels
data.yaml
```

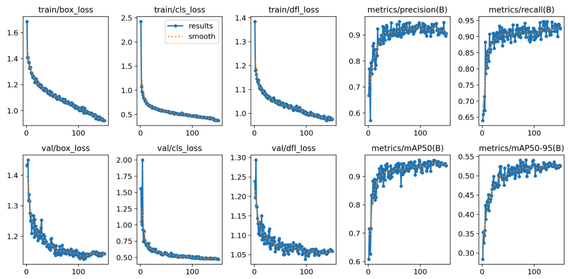
```
train: ../train/images
val: ../valid/images
test: ../test/images

nc: 2
names: ['name-tag', 'people']

roboflow:
  workspace: 123-djtgh
  project: staff-exclusion
  version: 1
  license: CC BY 4.0
```

Figure 2: Dataset Folder format & yaml file format

Model training

I use google calab, switch to gpu environment, use YOLO library to train my database.

- install : `!pip install ultralytics`
- train: `!yolo train model=yolov8s.pt data="data.yaml" epochs=150 imgsz=640`



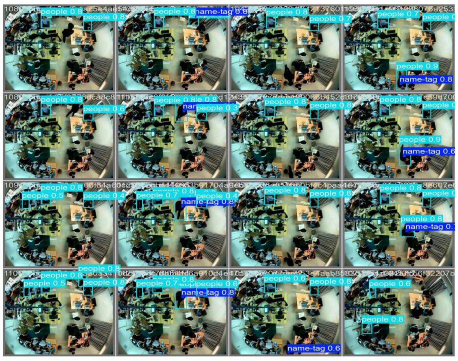


Figure 3: YOLOv8 train result

Staff Exclusion

The task is to identify staff wearing name tags. To predict staff: If the entire bounding box of the name tag is inside the people bounding box, then people are staff.

For the results of the YOLO model, the bounding box coordinates of the top right and bottom left corner of the target are output. Therefore, we can determine whether the entire bounding box of the name tag is inside the people bounding box.

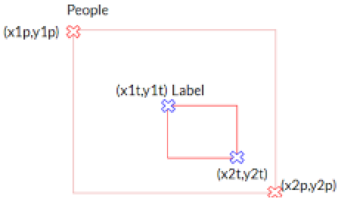
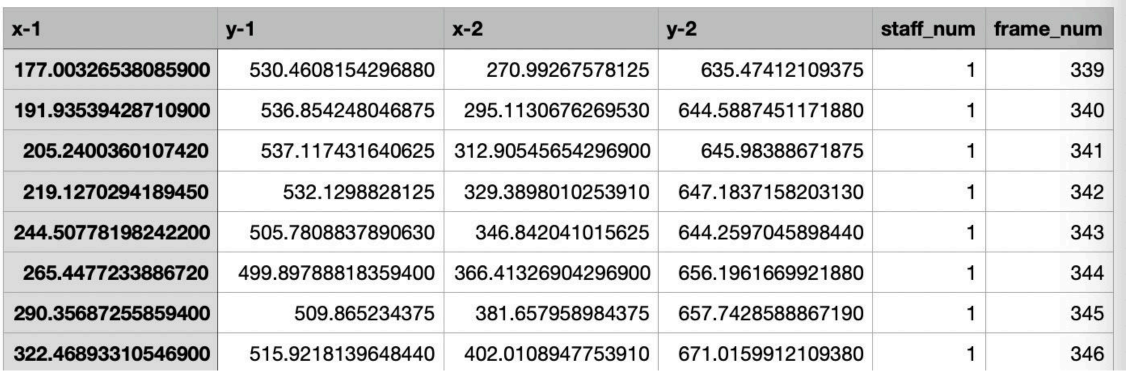


Figure 4: Example of Staff Identification

```
model = YOLO('yolov8s.pt')
model = YOLO('best.pt')

results = model(frame)
boxes = results[0].boxes

for name_tag in boxes:
    if name_tag.cls[0] == 0:
        for box in boxes:
            if (name_tag.xyxy[0][0] >= box.xyxy[0][0] and
                name_tag.xyxy[0][1] >= box.xyxy[0][1] and
                name_tag.xyxy[0][2] <= box.xyxy[0][2] and
                name_tag.xyxy[0][3] <= box.xyxy[0][3] and
                box.cls[0] == 1.0):
                staff_coordinate.append(box.xyxy[0].tolist())
```



x-1	y-1	x-2	y-2	staff_num	frame_num
177.00326538085900	530.4608154296880	270.99267578125	635.47412109375	1	339
191.93539428710900	536.854248046875	295.1130676269530	644.5887451171880	1	340
205.2400360107420	537.117431640625	312.90545654296900	645.98388671875	1	341
219.1270294189450	532.1298828125	329.3898010253910	647.1837158203130	1	342
244.50778198242200	505.7808837890630	346.842041015625	644.2597045898440	1	343
265.4477233886720	499.89788818359400	366.41326904296900	656.1961669921880	1	344
290.35687255859400	509.865234375	381.657958984375	657.7428588867190	1	345
322.46893310546900	515.9218139648440	402.0108947753910	671.0159912109380	1	346

Figure 5 Output frame number and Coordinates of Staff