

# CLI Usage

**Note:** If using the docker container from the factory image, you don't need to rebuild the environment. The environment is already set up, you just need to enter the docker according to the previous tutorial and run the corresponding function commands to use it.

## 1. Download Source Code

⚠ This step can be skipped if you are using the docker container from the factory image.

```
git clone https://github.com/ultralytics/ultralytics.git
```

## 2. Use CLI for Prediction

**CLI** (Command-Line Interface) is a way for users to interact with computers or software programs. Users run programs or execute tasks by typing text commands, rather than clicking icons or buttons through a graphical user interface (GUI).

### 2.1. CLI Usage Syntax

```
yolo TASK MODE ARGS
```

where TASK (optional) is one of [detect, segment, classify, pose, obb]  
MODE (required) is one of [train, val, predict, export, track, benchmark]  
ARGS (optional) are any number of custom 'arg=value' pairs like 'imgsz=320' that override defaults.

### 2.2. Image Prediction

Use yolo11n.pt to predict images that come with the ultralytics project: If the system cannot find the corresponding model file in the directory where the command is run, it will automatically download (if download fails, you can copy the model in yourself)

Enter the project folder:

```
cd ~/ultralytics/ultralytics
```

Use yolo11n.pt to detect images in the target folder and output results:

```
#Photo path can be customized to your own photo
yolo predict model=yolo11n source='/root/ultralytics/ultralytics/assets/bus.jpg'
```

```
root@raspberrypi:~/ultralytics/ultralytics# yolo predict model=yolo11n source='/root/ultralytics/ultralytics/assets/bus.jpg'
Ultralytics 8.3.154 Python-3.10.12 torch-2.1.2 CPU (Cortex-A76)
YOLO11n summary (fused): 100 layers, 2,616,248 parameters, 0 gradients, 6.5 GFLOPs

image 1/1 /root/ultralytics/ultralytics/assets/bus.jpg: 640x480 4 persons, 1 bus, 537.2ms
Speed: 15.1ms preprocess, 537.2ms inference, 27.2ms postprocess per image at shape (1, 3, 640, 480)
Results saved to runs/detect/predict3
Learn more at https://docs.ultralytics.com/modes/predict
root@raspberrypi:~/ultralytics/ultralytics#
```

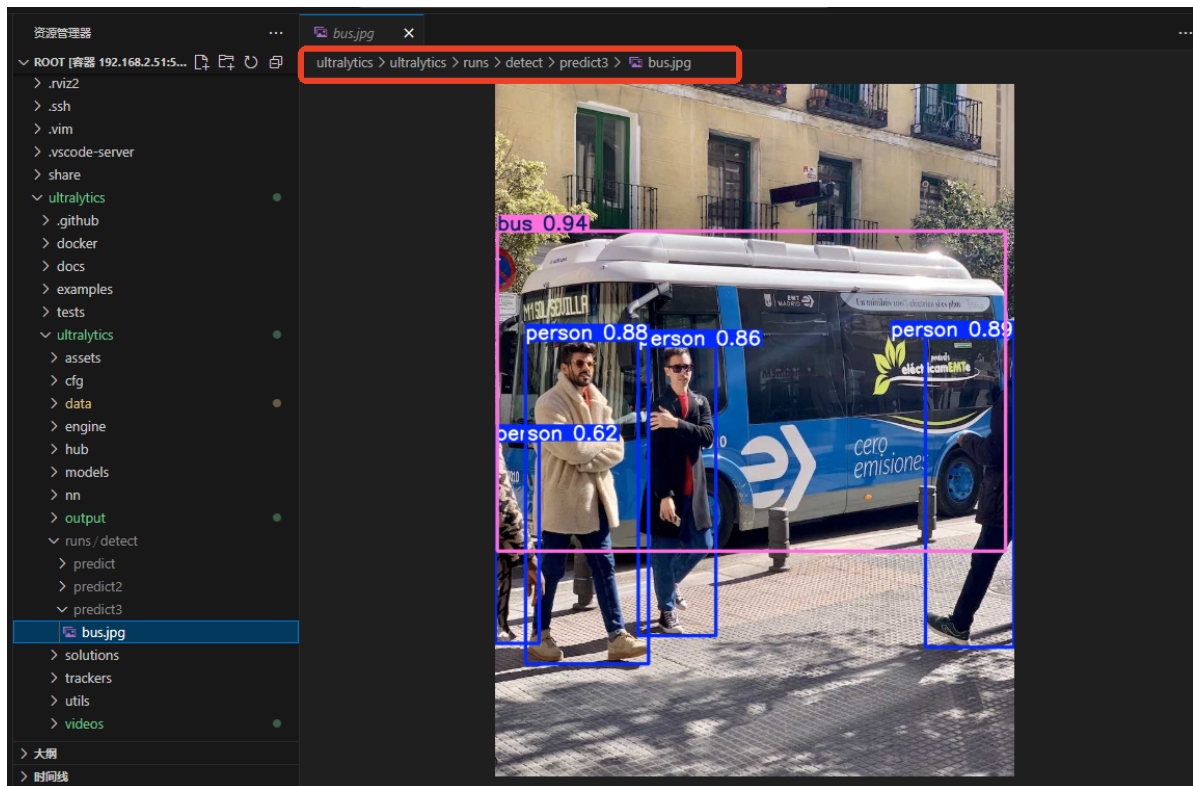
The appearance of the above image indicates successful prediction.

## Effect Preview

yolo recognition output image location:

```
/root/ultralytics/ultralytics/runs/detect
```

Each time you run, the predict under this folder automatically increments by 1



## 2.3. Video Prediction

Use yolo11n.pt to predict videos in the ultralytics project (not videos that come with ultralytics): If the system cannot find the corresponding model file in the directory where the command is run, it will automatically download (if download fails, you can copy the model in yourself)

Enter the project folder:

```
cd ~/ultralytics/ultralytics
```

Use yolo11n.pt to detect videos in the target folder and output results:

```
#video path can be customized to your own path
yolo predict model=yolo11n
source='/root/ultralytics/ultralytics/videos/people_animals.mp4'
```

```

video 1/1 (frame 65/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 392.9ms
video 1/1 (frame 66/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 426.7ms
video 1/1 (frame 67/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 393.1ms
video 1/1 (frame 68/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 1 sheep, 388.5ms
video 1/1 (frame 69/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 2 sheeps, 391.3ms
video 1/1 (frame 70/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 1 sheep, 387.0ms
video 1/1 (frame 71/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 395.9ms
video 1/1 (frame 72/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 1 sheep, 391.6ms
video 1/1 (frame 73/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 418.7ms
video 1/1 (frame 74/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 1 sheep, 404.9ms
video 1/1 (frame 75/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 439.7ms
video 1/1 (frame 76/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 385.7ms
video 1/1 (frame 77/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 1 sheep, 389.8ms
video 1/1 (frame 78/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 386.8ms
video 1/1 (frame 79/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 406.6ms
video 1/1 (frame 80/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 398.1ms
video 1/1 (frame 81/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 382.9ms
video 1/1 (frame 82/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 2 sheeps, 389.7ms
video 1/1 (frame 83/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 390.9ms
video 1/1 (frame 84/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 1 sheep, 388.9ms
video 1/1 (frame 85/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 419.4ms
video 1/1 (frame 86/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 399.1ms
video 1/1 (frame 87/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 397.8ms
video 1/1 (frame 88/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 421.1ms
video 1/1 (frame 89/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 385.1ms
video 1/1 (frame 90/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 3 dogs, 390.5ms
video 1/1 (frame 91/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 370.0ms
video 1/1 (frame 92/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 412.2ms
video 1/1 (frame 93/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 392.7ms
video 1/1 (frame 94/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 382.2ms
video 1/1 (frame 95/95) /root/ultralytics/ultralytics/videos/people_animals.mp4: 384x640 2 persons, 2 dogs, 399.3ms
Speed: 6.3ms preprocess, 399.4ms inference, 1.5ms postprocess per image at shape (1, 3, 384, 640)
Results saved to runs/detect/predict4
[?] Learn more at https://docs.ultralytics.com/modes/predict
root@raspberrypi:~/ultralytics/ultralytics#

```

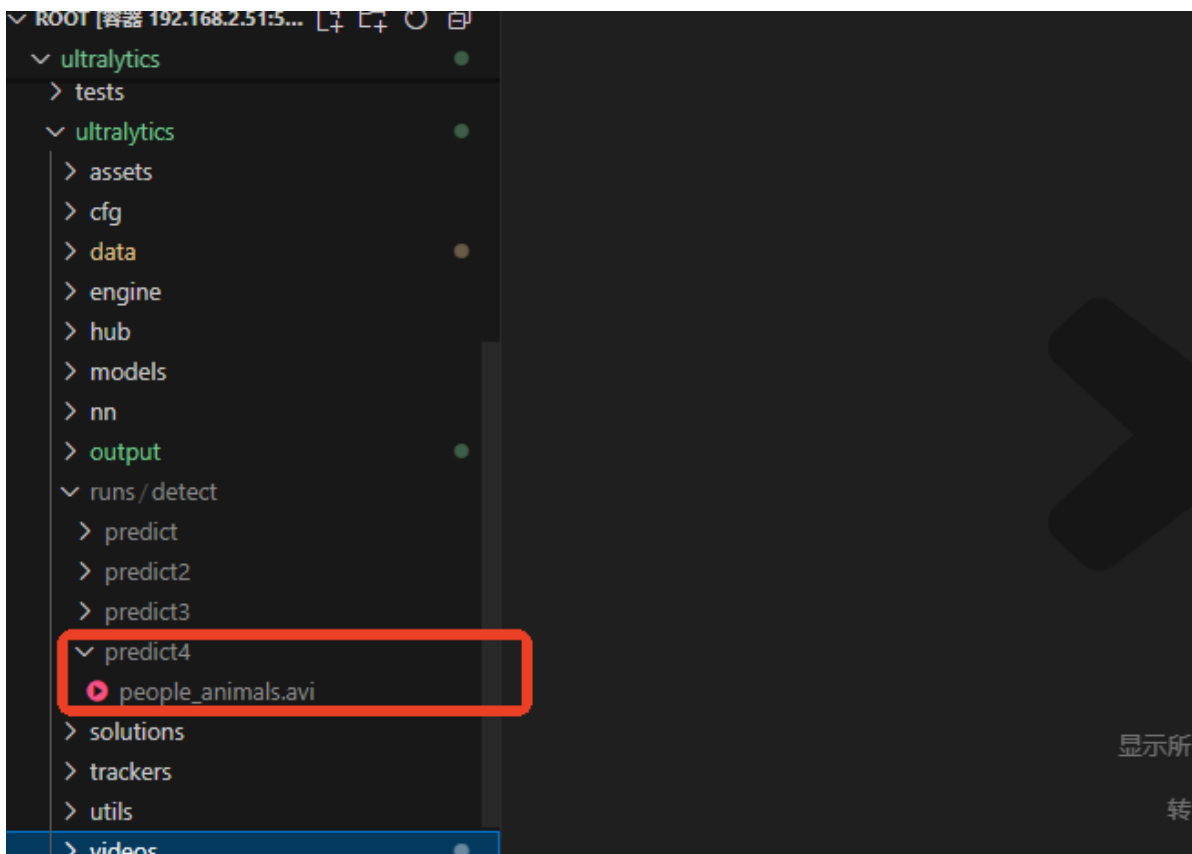
The appearance of the above image indicates successful prediction.

## Effect Preview

yolo recognition output video location:

```
/root/ultralytics/ultralytics/runs/detect/
```

Each time you run, the predict under this folder automatically increments by 1



You can use ffmpeg to view videos, use the following command to install:

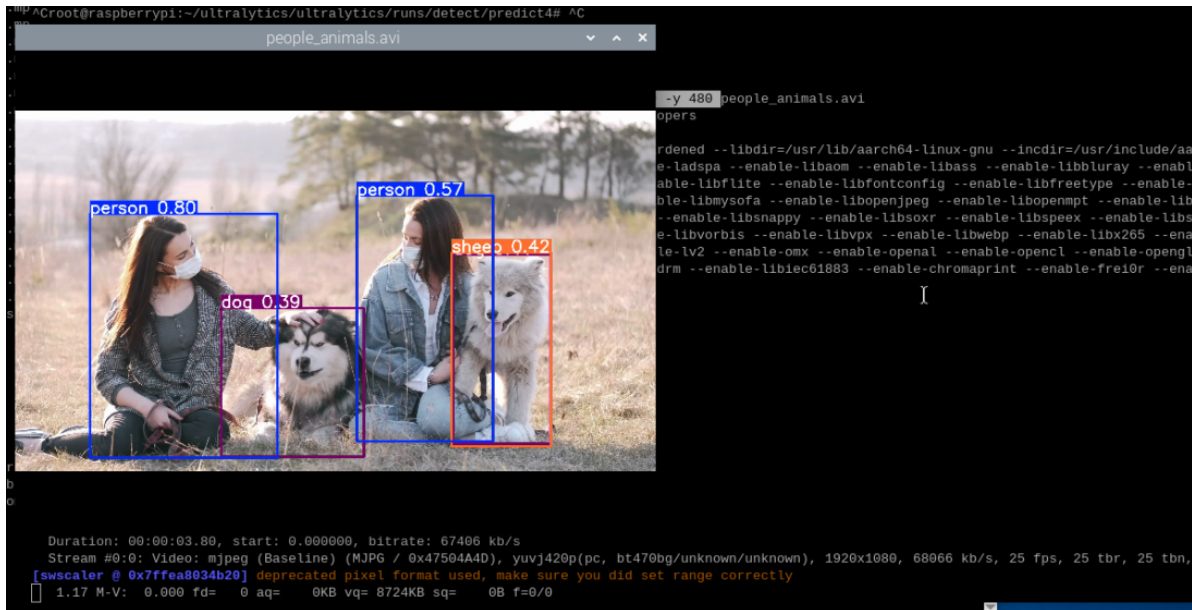
```

sudo apt update
sudo apt install ffmpeg -y

```

Play video:

```
#Need to go to the corresponding path, here is people_animals.avi in the predict4
folder
cd /root/ultralytics/ultralytics/runs/detect/predict4/
#Play video
ffplay -x 640 -y 480 people_animals.avi
```



## 2.4. Real-time Prediction

⚠ This function only supports USB cameras, depth cameras are not suitable for CLI

Use yolo11n.pt to predict USB camera feed. If the system cannot find the corresponding model file in the directory where the command is run, it will automatically download

Enter the project folder:

```
cd ~/ultralytics/ultralytics
```

Use yolo11n.onnx to detect camera feed and output results:

```
yolo predict model=yolo11n.onnx source=0 save=False show # Object detection
# Instance segmentation: yolo predict model=yolo11n-seg.onnx source=0 save=False
show
# Image classification: yolo predict model=yolo11n-cls.onnx source=0 save=False
show
# Pose estimation: yolo predict model=yolo11n-pose.onnx source=0 save=False show
# Oriented object detection: yolo predict model=yolo11n-obb.onnx source=0
save=False show
```

Click on the terminal and press "Ctrl + C" shortcut key to terminate the program!

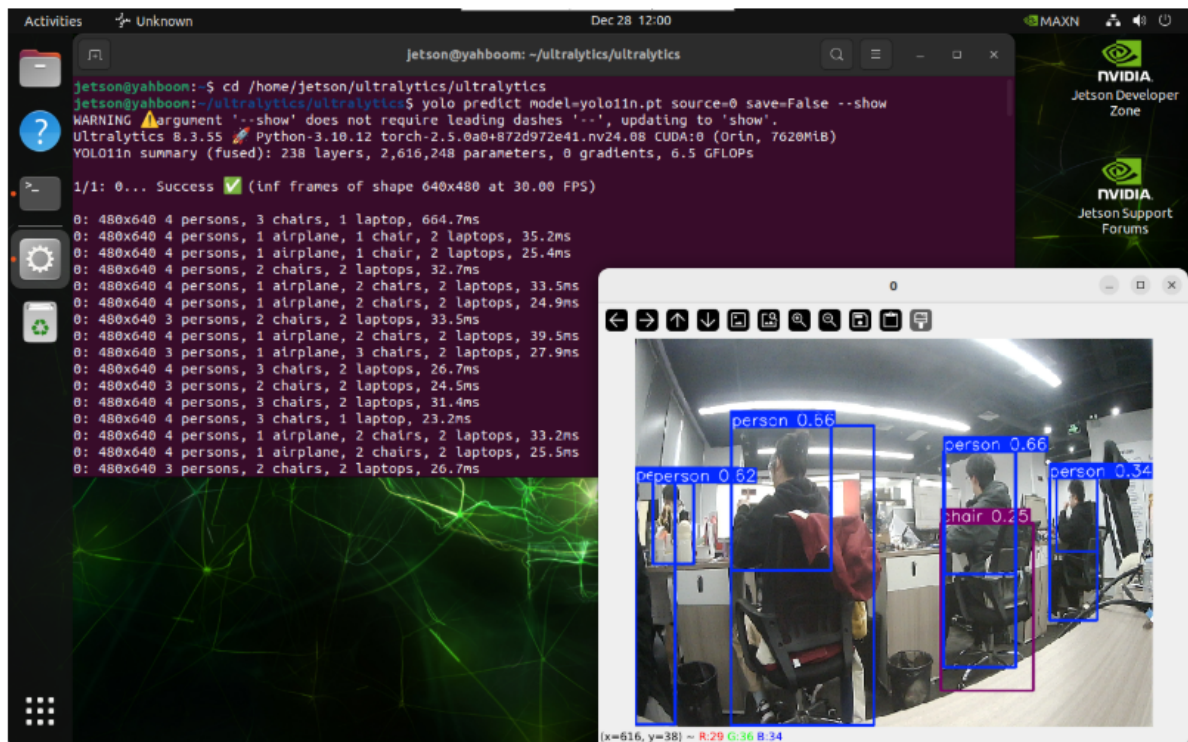
### Parameter Description

**model** : Specify YOLO model

**source** : Specify recognition source: multiple cameras can switch numbers

**save=False** : Disable saving results

**show** : Real-time display



## References

<https://github.com/ultralytics/ultralytics>

<https://docs.ultralytics.com/usage/cli/>