

# Yolo5 Real time detection

## Yolo5 Real time detection

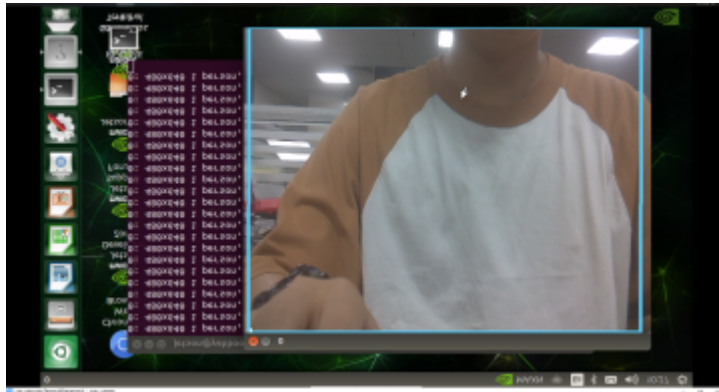
- 1.usage method
- 2.note

## 1.usage method

If you are directly using the YAHBOOM version of the mirror and using a CSI camera  
Run the following command directly

```
cd ~/yolov5 && python3 detect.py --source 0
```

After waiting for a while, the CSI camera turned on  
You can see that the screen will display the recognized object



Press Ctrl+c and turn off the camera screen to end the program  
And store the identified results in the yolov5/runs/detect/exp path (a video)

## 2.note

1. If an error is reported midway due to network issues, it can be placed in the folder of yolov5 from the attachment of the environment setup, yolov5s.pt
2. If you are using a USB camera, you need to make a simple modification to uncomment line 292 in the datasets.py file of ~/yolov5/utils. Add '#' to line 293.

```
datasets.py 6 X
C: > Users > Administrator > Desktop > datasets.py > get_hash
283 for i, s in enumerate(sources):
284     # Start the thread to read frames from the video stream
285     print(f'{i + 1}/{n}: {s}... ', end='')
286     url = eval(s) if s.isnumeric() else s
287     #if 'youtube.com/' in url or 'youtu.be/' in url: # if source is YouTube video
288     #    check_requirements(('pafy', 'youtube_dl'))
289     #    import pafy
290     #    url = pafy.new(url).getbest(preftype="mp4").url
291     #cap = cv2.VideoCapture(url)
292     #cap = cv2.VideoCapture(0)#OPEN USB
293     cap = cv2.VideoCapture(gst_str,cv2.CAP_GSTREAMER) #open CSI
294     assert cap.isOpened(), f'Failed to open {s}'
295     w = int(cap.get(cv2.CAP_PROP_FRAME_WIDTH))
296     h = int(cap.get(cv2.CAP_PROP_FRAME_HEIGHT))
297     self.fps = cap.get(cv2.CAP_PROP_FPS) % 100
298
299     _, self.imgs[i] = cap.read() # guarantee first frame
300     thread = Thread(target=self.update, args=([i, cap]), daemon=True)
301     print(f' success ({w}x{h} at {self.fps:.2f} FPS).')
302     thread.start()
303     print('') # newline
304
305     # check for common shapes
306     s = np.stack([letterbox(x, self.img_size, stride=self.stride)[0].shape for x in self.imgs], 0)
307     self.rect = np.unique(s, axis=0).shape[0] == 1 # rect inference if all shapes equal
308     if not self.rect:
309         print('WARNING: Different stream shapes detected. For optimal performance supply similar')
310
311     def update(self, index, cap):
312         # Read next stream frame in a daemon thread
313         n = 0
314         while cap.isOpened():
315             n += 1
316             # _, self.imgs[index] = cap.read()
317             cap.grab()
```

3. If it is a self built image that is not configured using the YAHBOOM version, you need to rewrite the datasets. py file yourself. Please refer to the link below

<https://blog.csdn.net/AlwaysNoError/article/details/123298884>

If the image you have built still experiences SPPF errors, you can refer to the tutorial

[https://blog.csdn.net/m0\\_50004939/article/details/126739291](https://blog.csdn.net/m0_50004939/article/details/126739291)