CYCU BME
Final presentation

陳信學 (Shin-Shiue, Chen)、古騏銘(Chi-Ming, Ku)、
鄒佳格(Chia-Ke Mia Tsou)、黃天澤(Tien-Tse, Huang)
(Department of Biomedical Engineering, Chung Yuan Christian University)

Mentors: Eason Hung (NVIDIA)

NARLabs 國家實驗研究院
國家高速網路與計算中心
National Center for High-performance Computing

中原大學生物醫學工程學系
Department of Biomedical Engineering

OPEN HACKATHONS
OpenACC

Our application

Target: Otoscopy diagnosis

Mission: Object detection

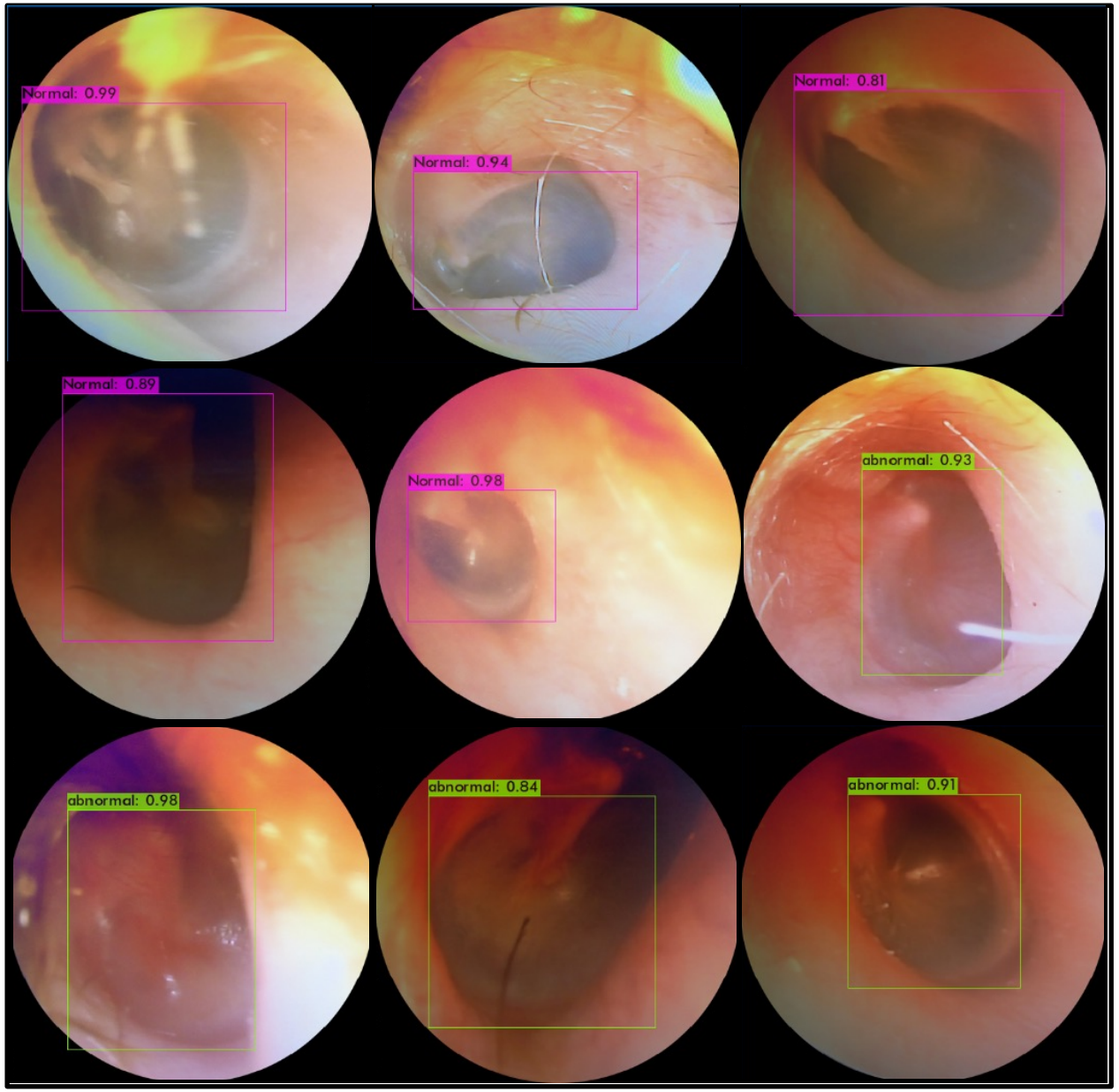
Model:

1. YOLOv4

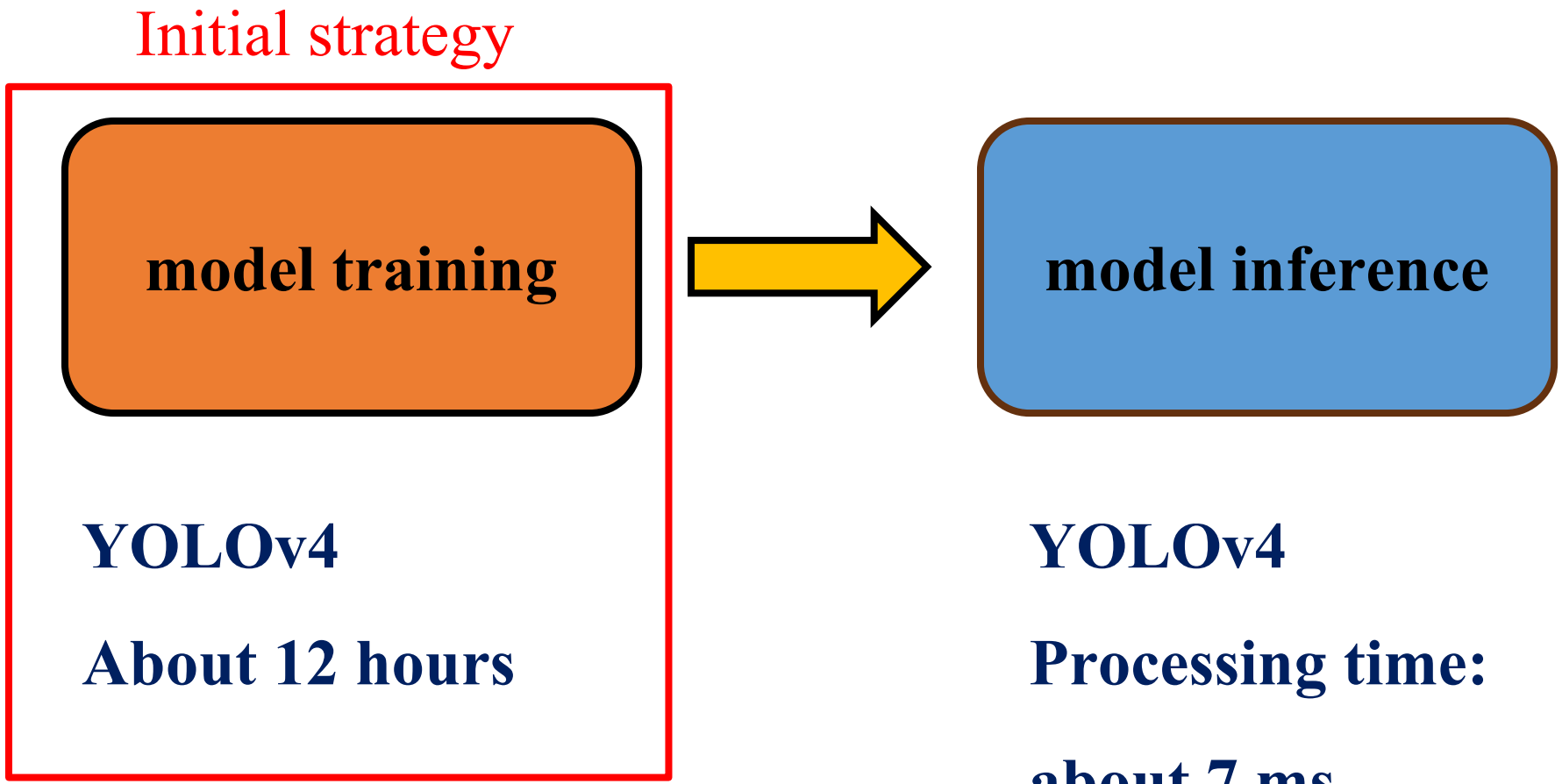
(127.248 BFLOPS)

2. YOLOv4 tiny

(14.502 BFLOPS)



Our initial pain point



Strategy change

CUDA and cuDNN are relatively mature applications

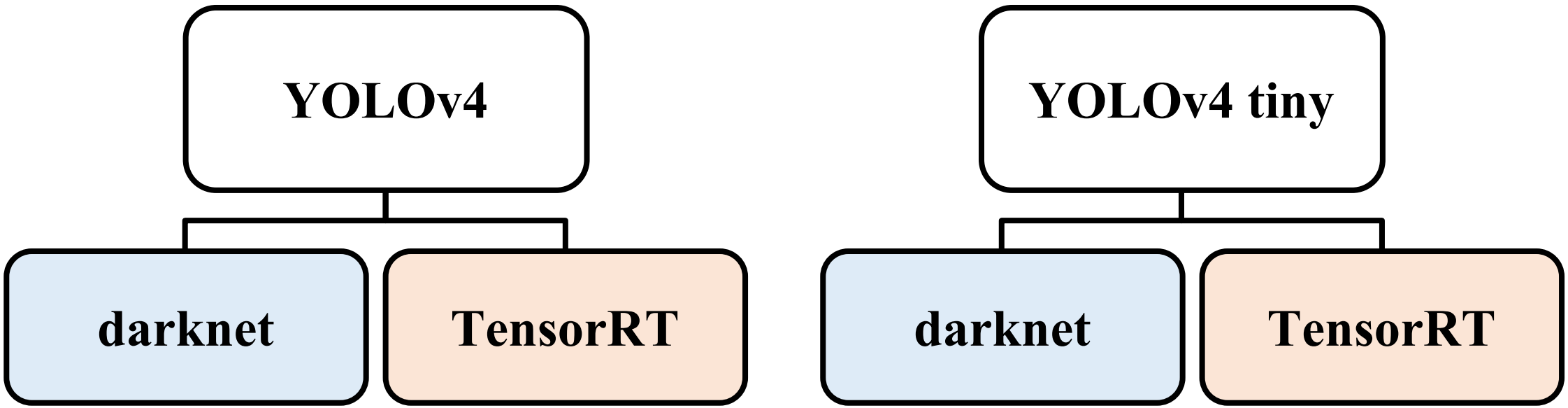
Our environment

GPU: NVIDIA GeForce RTX 3060

Image size: 96*96

Not suitable for the portable device

Our goal



Compare FPS performance and Nsight analysis

Average FPS for one minute video

Problems we encountered

The versions of CUDA and cuDNN are incompatible for tensorRT

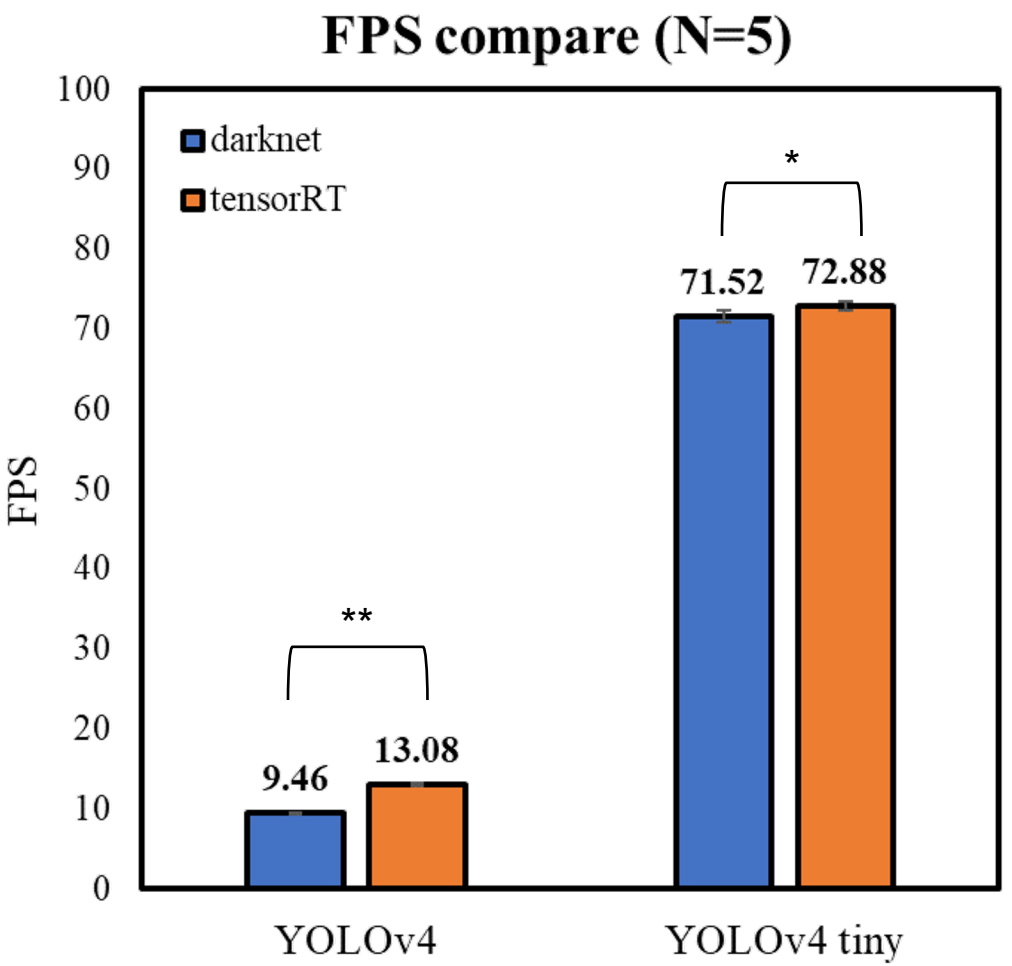
Error occurred when converting the darknet model into tensorRT model (change image size to 608*608)

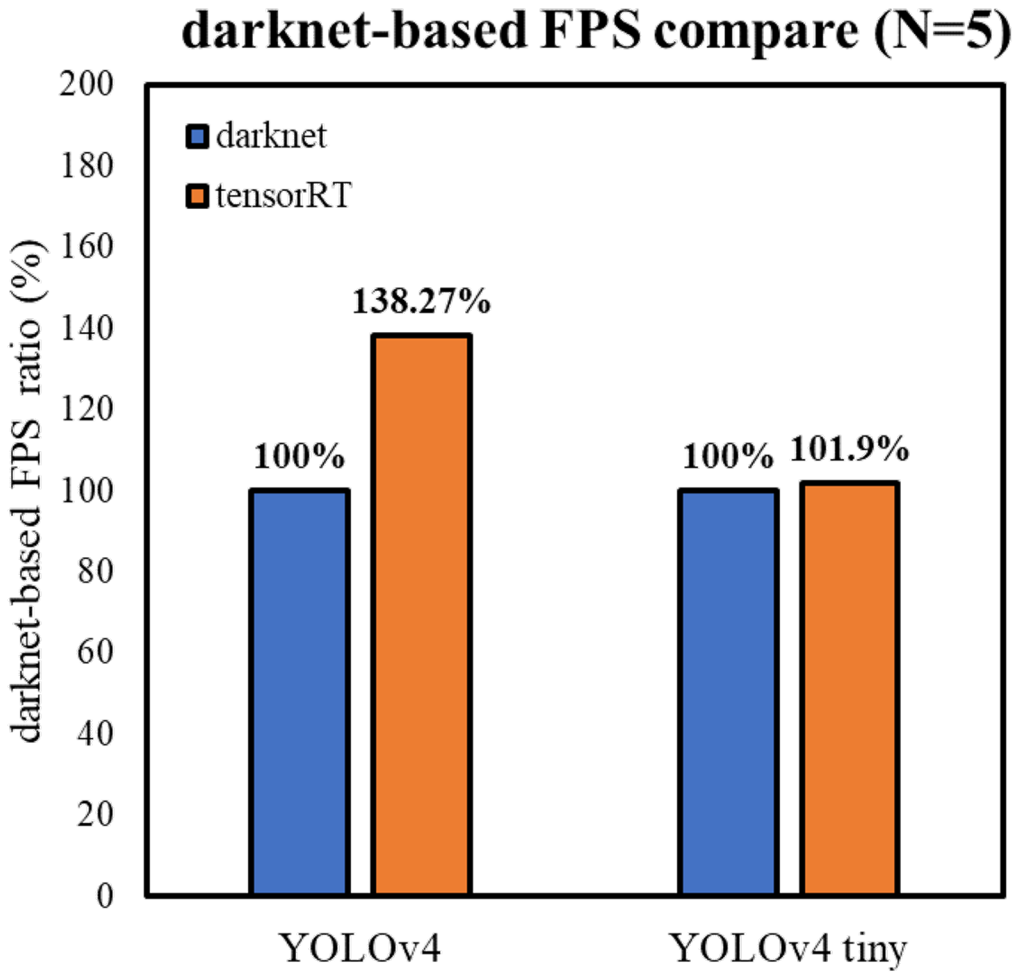
Training new model on A100 (4 hours for YOLOv4)

Environment of our local machine

Item	Content
CPU	Intel i7-6700
GPU	NVIDIA GeForce GTX 1050ti
RAM	16 GB DDR4
OS	Ubuntu 20.04
CUDA	CUDA Toolkit version 12.0
cuDNN	Version 8.8.0

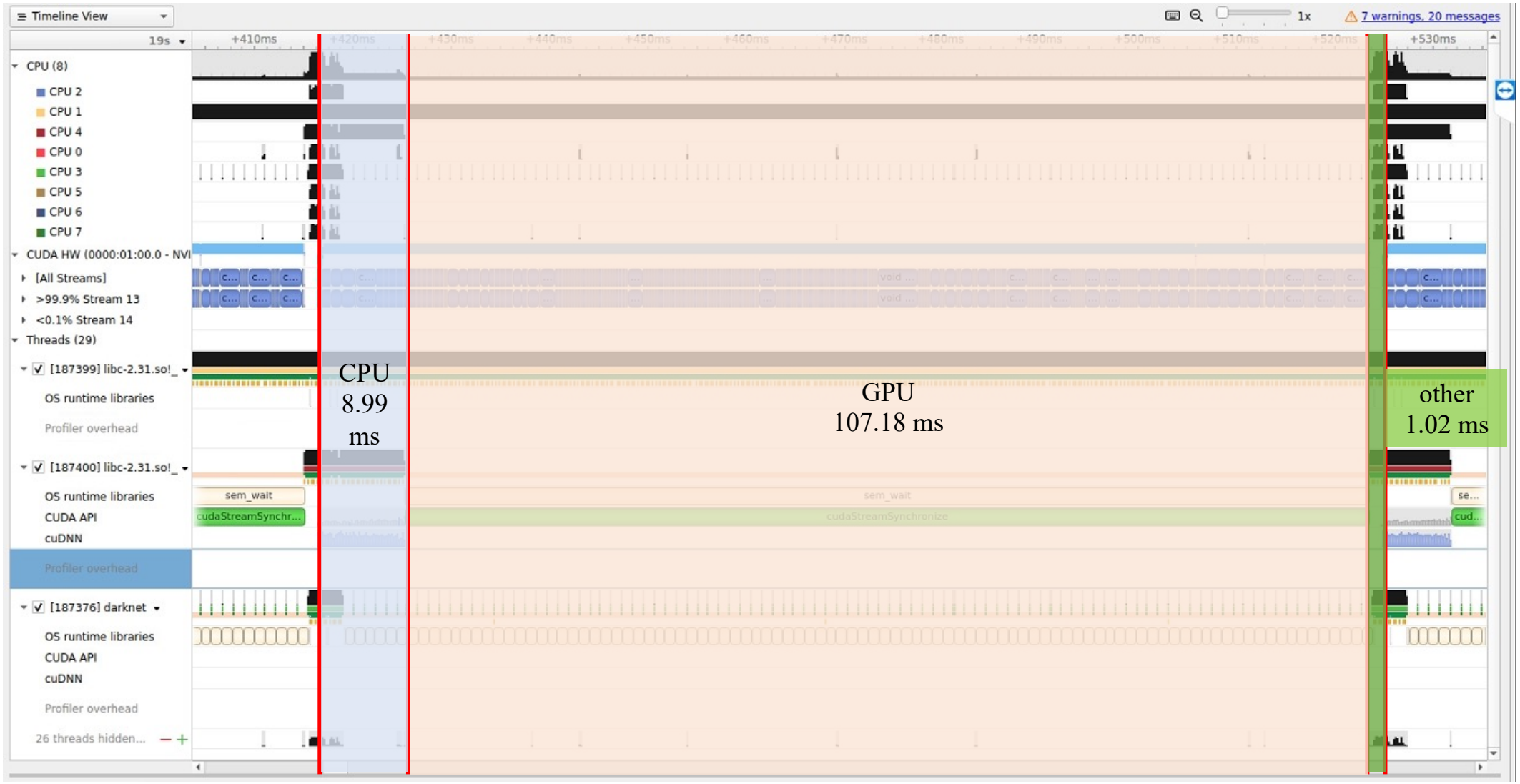
Result-FPS comparison



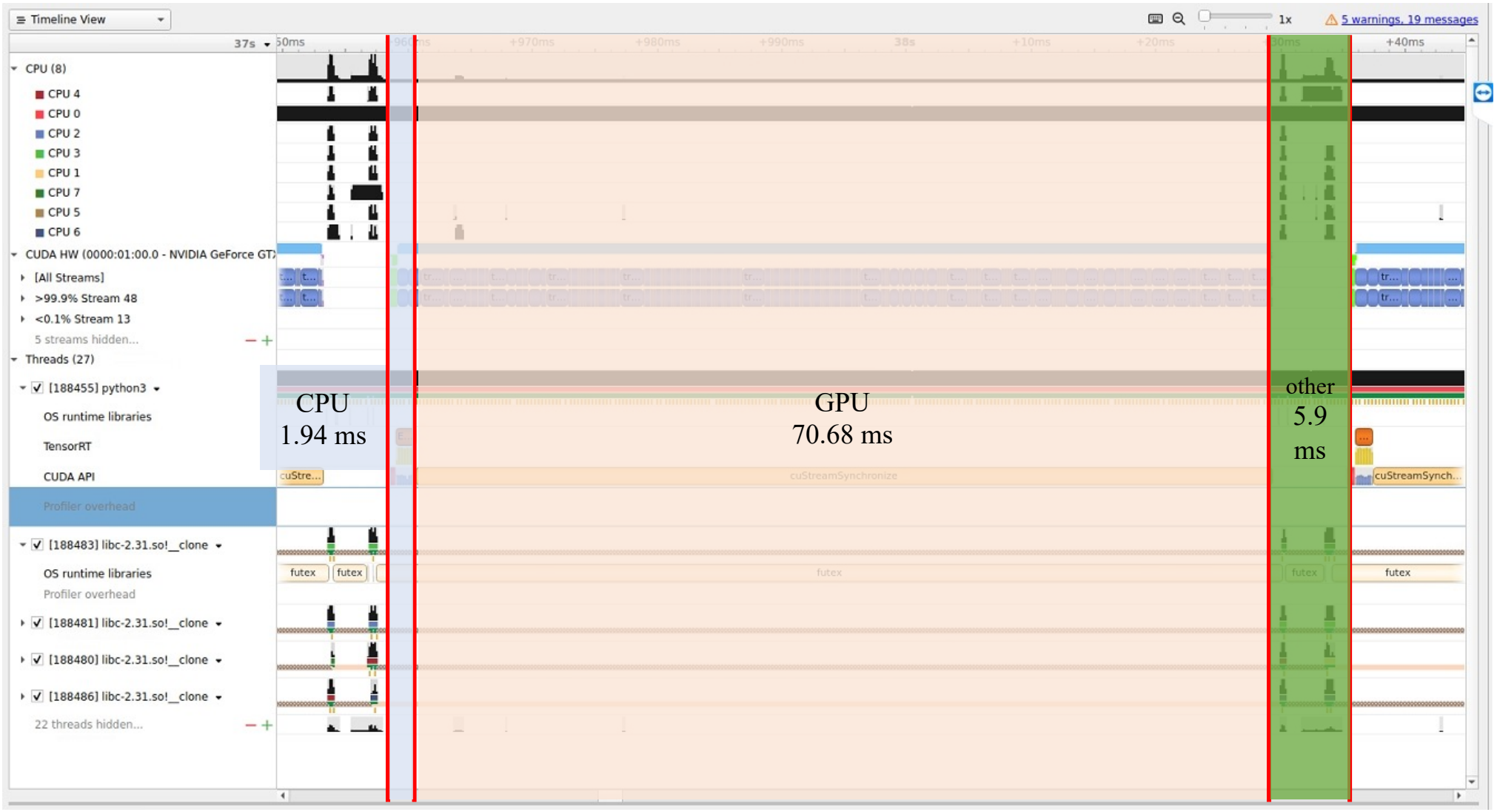


*: $P<0.05$; **: $P<0.01$; P values were obtained with the Mann-Whitney test

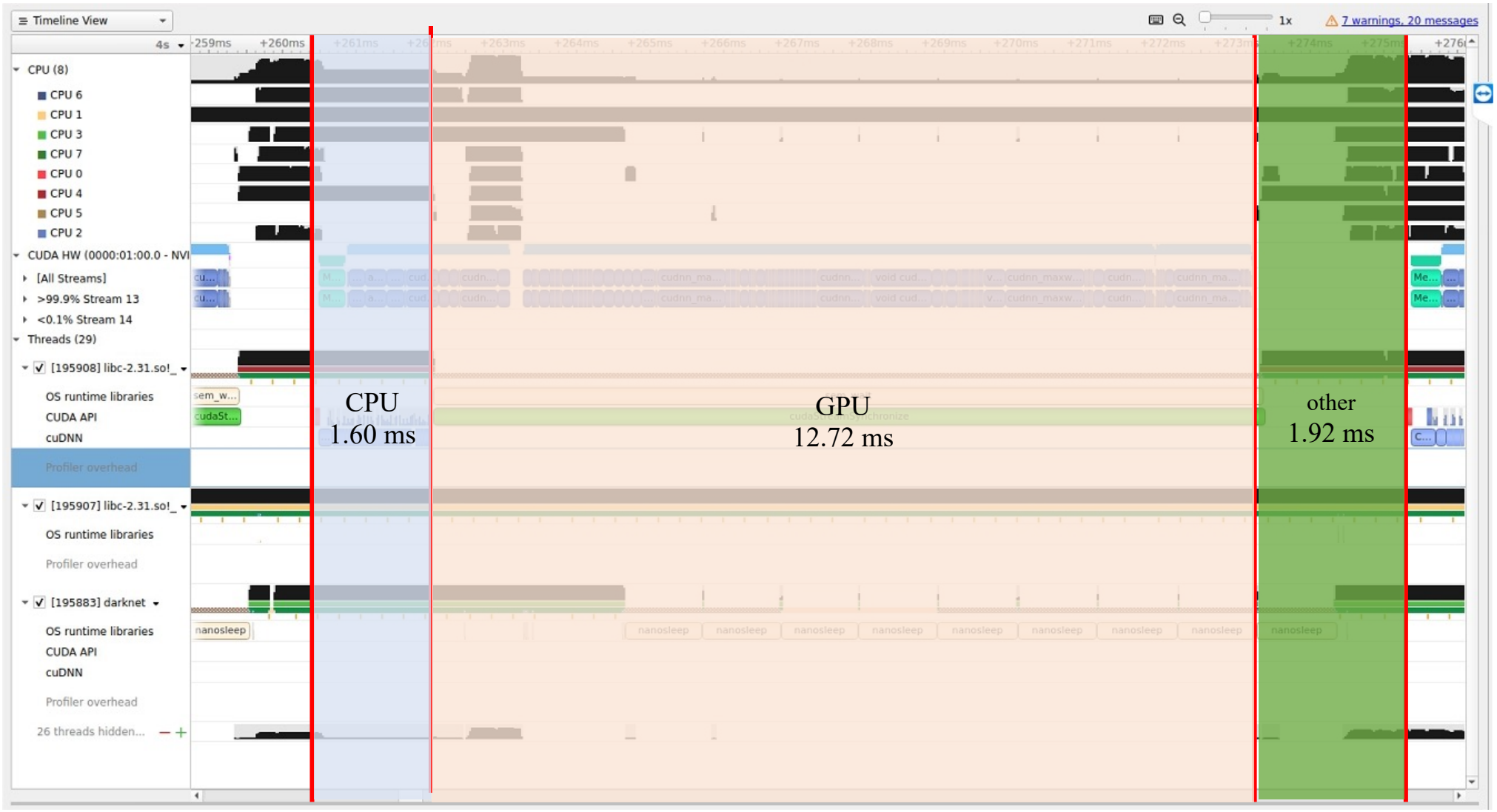
Profiler output: YOLOv4-darknet



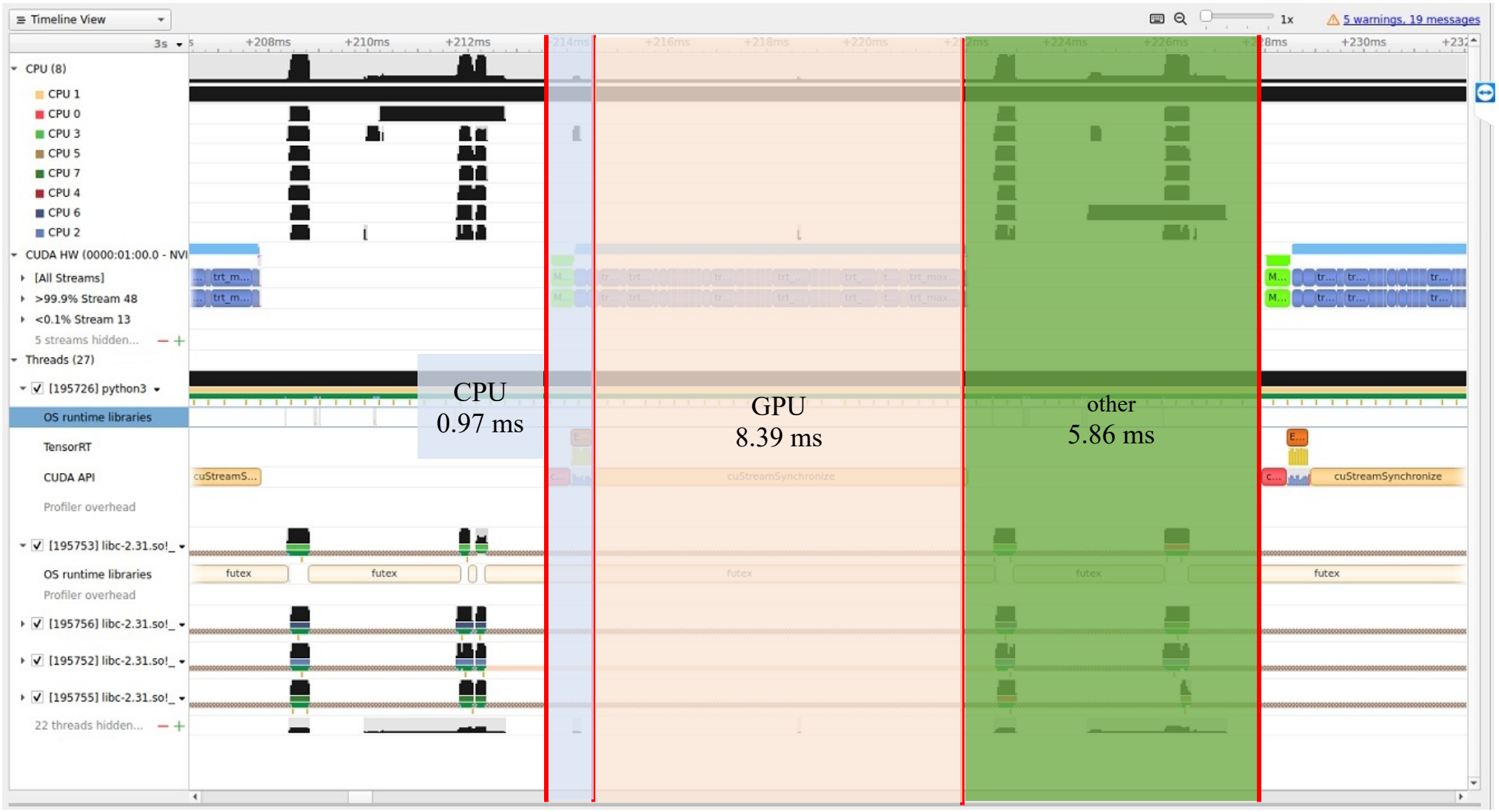
Profiler output: YOLOv4-tensorRT



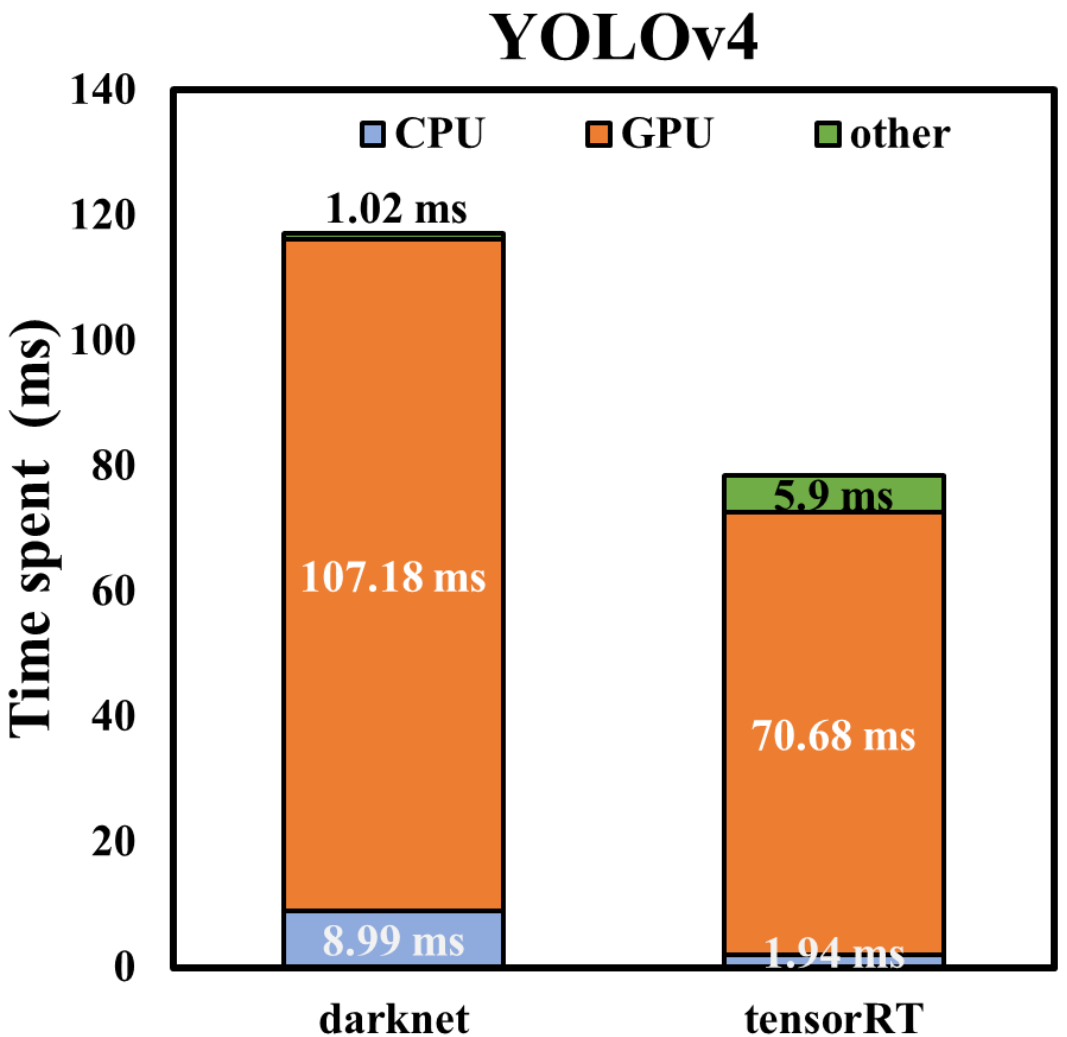
Profiler output: YOLOv4 tiny-darknet

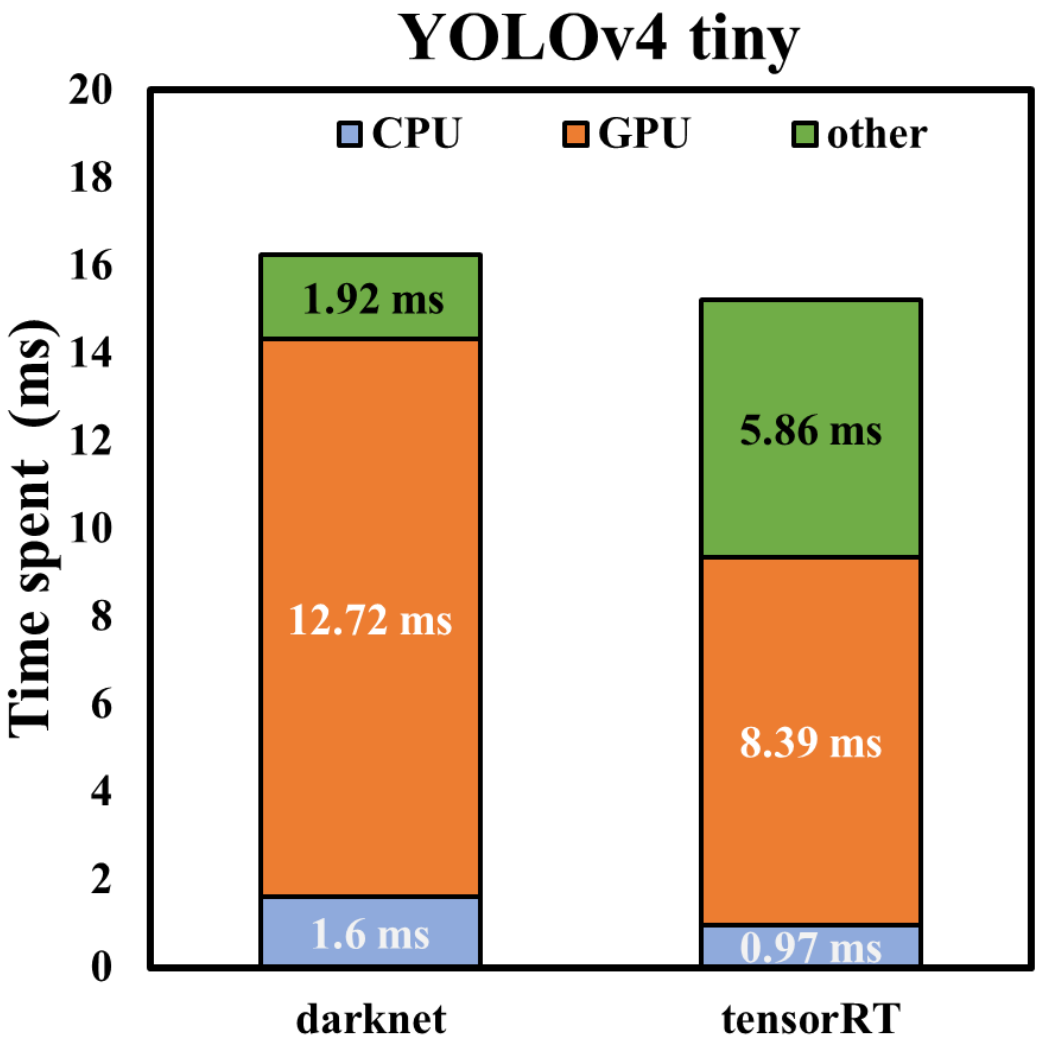


Profiler output: YOLOv4 tiny-tensorRT

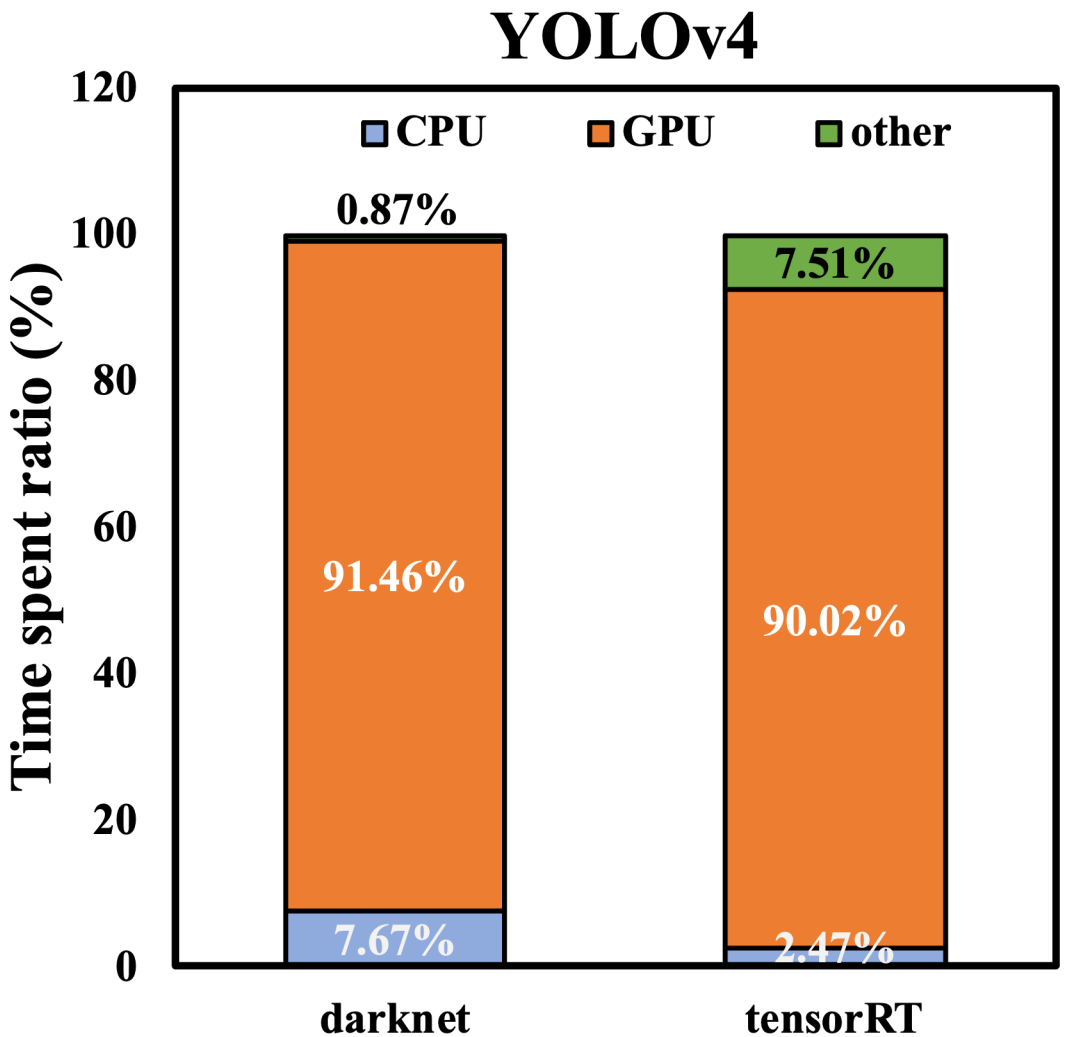


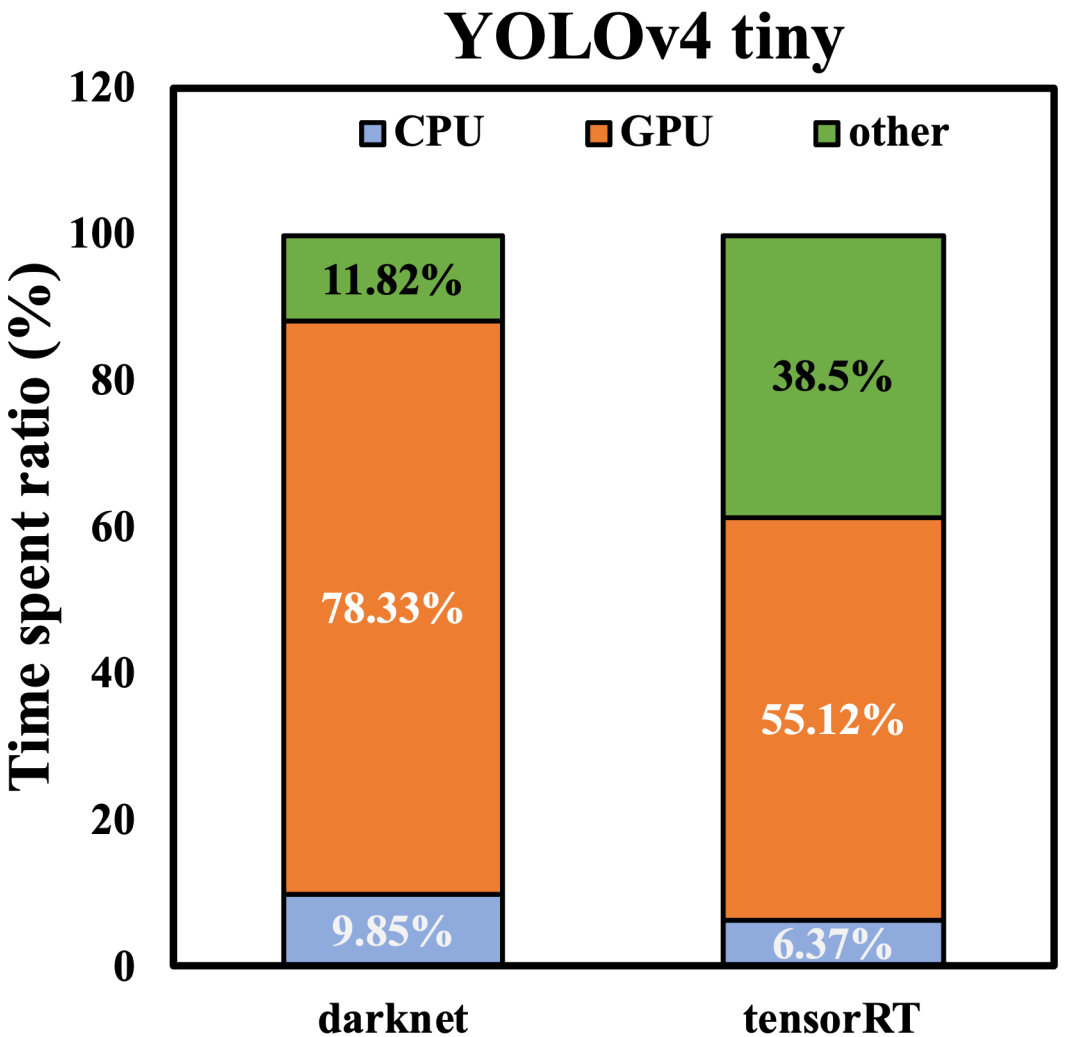
Result-Nsight comparison (ms)





Result-Nsight comparison (ratio)



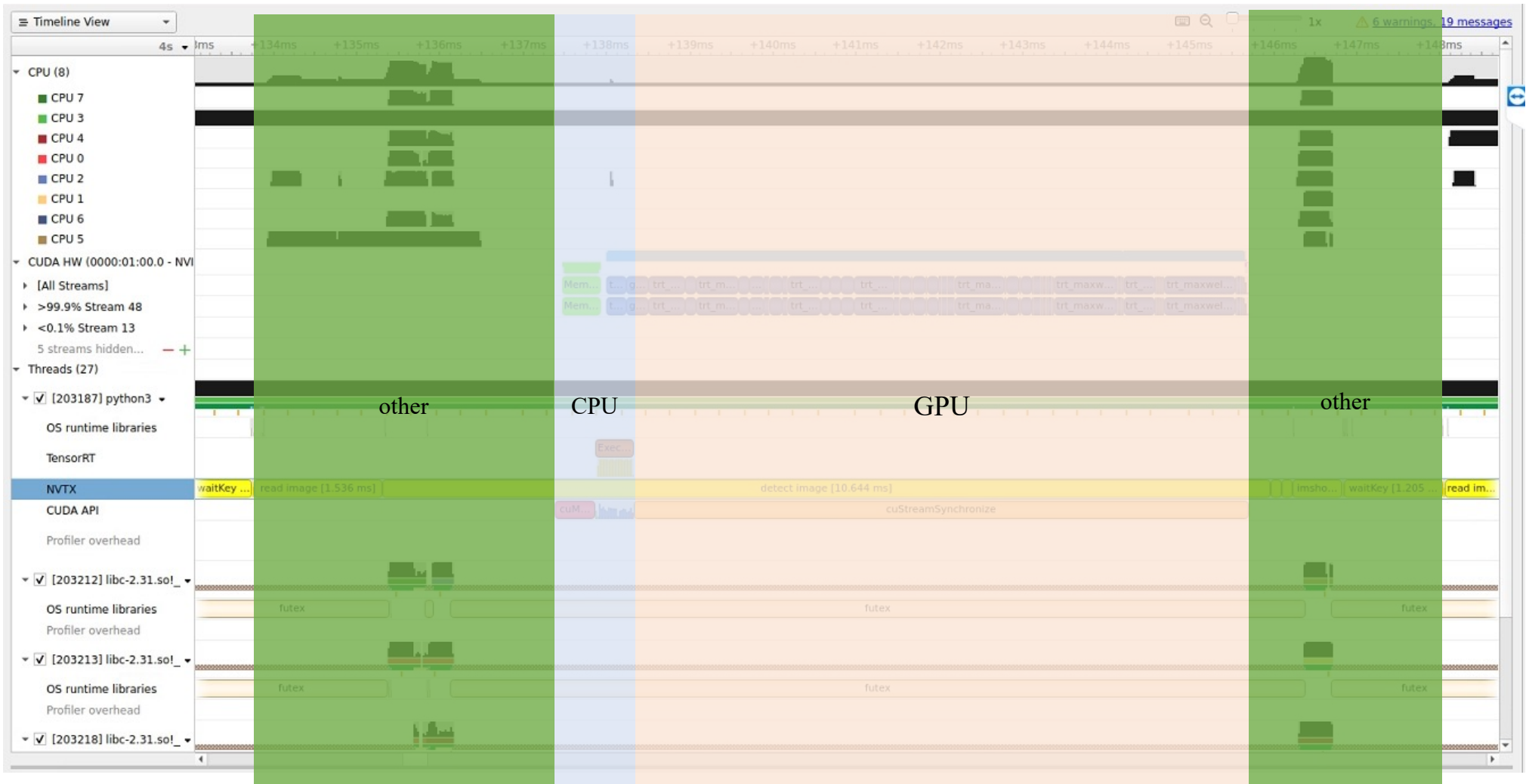


Nsight analysis with NVTX

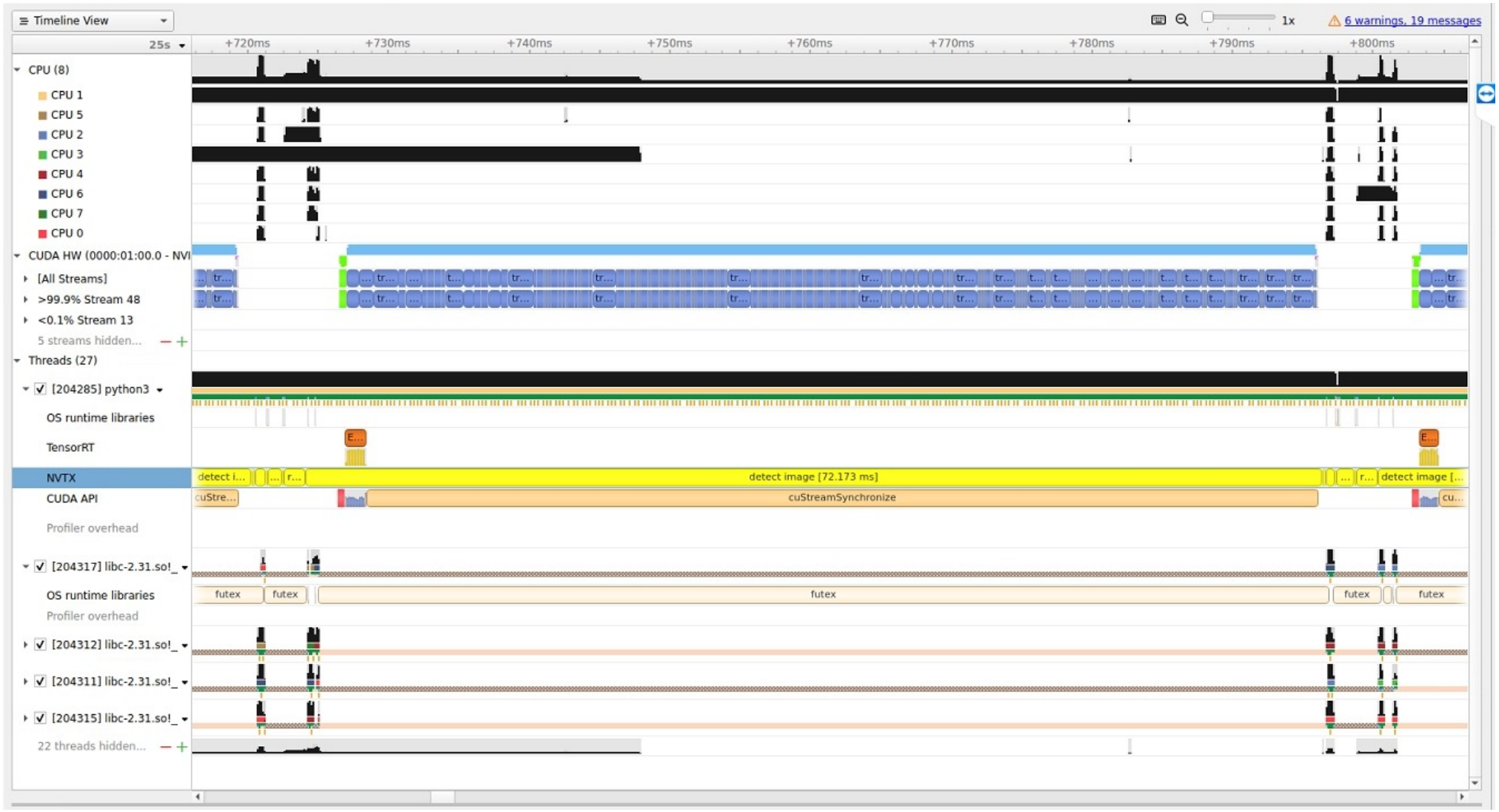
```
while True:
    with nvtx.annotate("read image", color="yellow"):
        if cv2.getWindowProperty(WINDOW_NAME, 0) < 0:
            break
        img = cam.read()
        if img is None:
            break
    with nvtx.annotate("detect image", color="yellow"):
        boxes, confs, clss = trt_yolo.detect(img, conf_th)
    with nvtx.annotate("draw_bboxes", color="yellow"):
        img = vis.draw_bboxes(img, boxes, confs, clss)
    with nvtx.annotate("show_fps", color="yellow"):
        img = show_fps(img, fps)
    with nvtx.annotate("imshow", color="yellow"):
        cv2.imshow(WINDOW_NAME, img)
    with nvtx.annotate("curr_fps", color="yellow"):
        toc = time.time()
        curr_fps = 1.0 / (toc - tic)
    with nvtx.annotate("fps", color="yellow"):
        avg_fps = avg_fps + curr_fps
        # calculate an exponentially decaying average of fps number
        fps = curr_fps if fps == 0.0 else (fps*0.95 + curr_fps*0.05)
        tic = toc

    with nvtx.annotate("waitKey", color="yellow"):
        key = cv2.waitKey(1)
    with nvtx.annotate("key", color="yellow"):
        if key == 27:  # ESC key: quit program
            break
        elif key == ord('F') or key == ord('f'):  # Toggle fullscreen
            full_scrn = not full_scrn
            set_display(WINDOW_NAME, full_scrn)
return avg_fps
```

Profiler output: YOLOv4 tiny-tensorRT (NVTX)



Profiler output: YOLOv4-tensorRT (NVTX)



Summary of our team's achievements during this Hackathon

Model inference optimization implementation based on tensorRT in Linux environment

Implementation of Nsight analysis, and apply it to understand the performance of model inference

Future work

Understand why the darknet model is faster in the "other" block

Thank you for your listening

This project thanks

Children's Hearing Foundation and NVIDIA

for their guidance and support.