# 训练

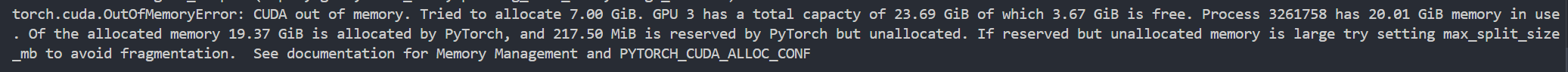
## 显存不足的问题

python -m src.main +experiment=re10k data\_loader.train.batch\_size=14

报错显存不足，改为指定空闲GPU训练：

CUDA\_VISIBLE\_DEVICES=2,3 python -m src.main +experiment=re10k data\_loader.train.batch\_size=14

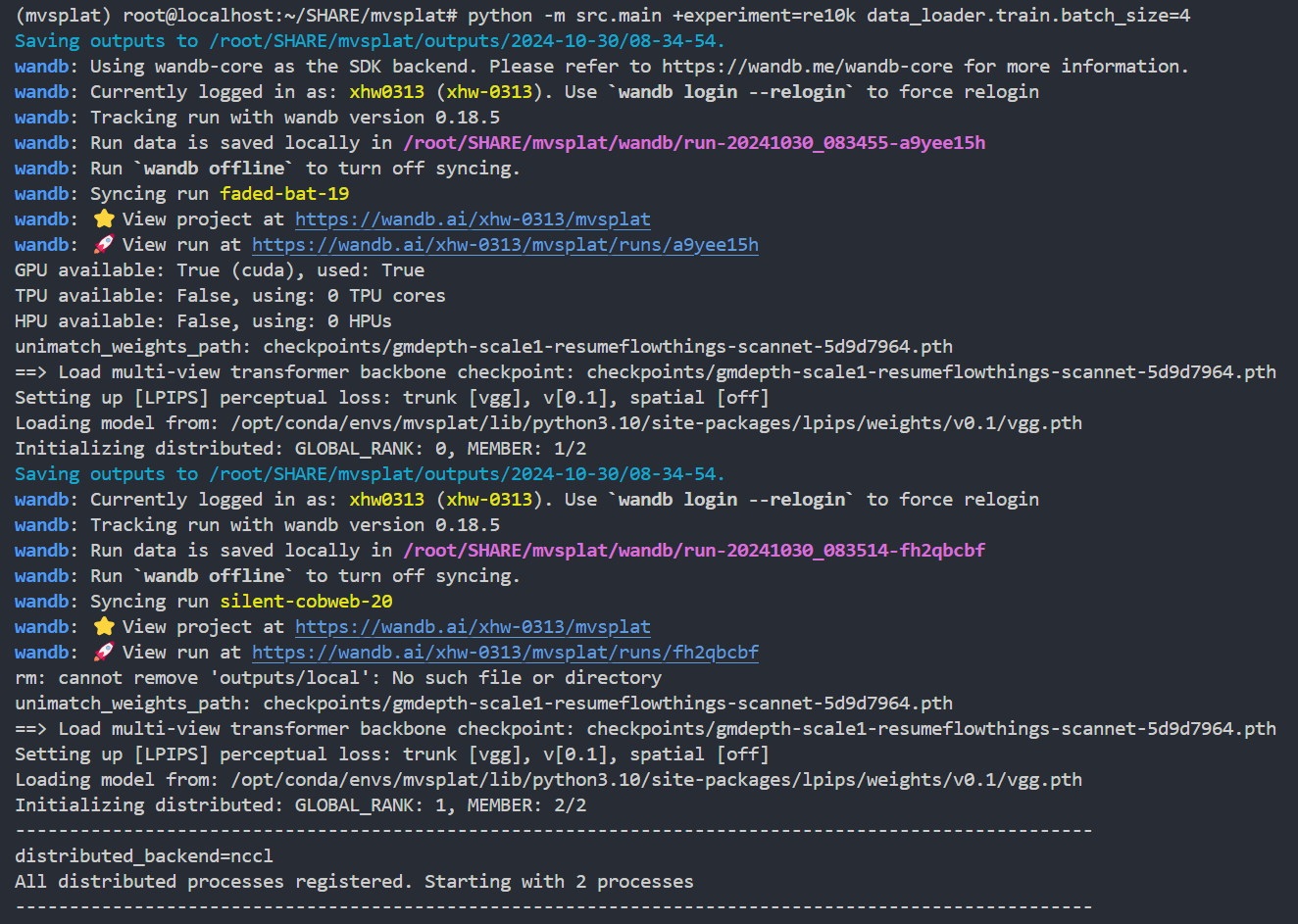
仍然显存不足

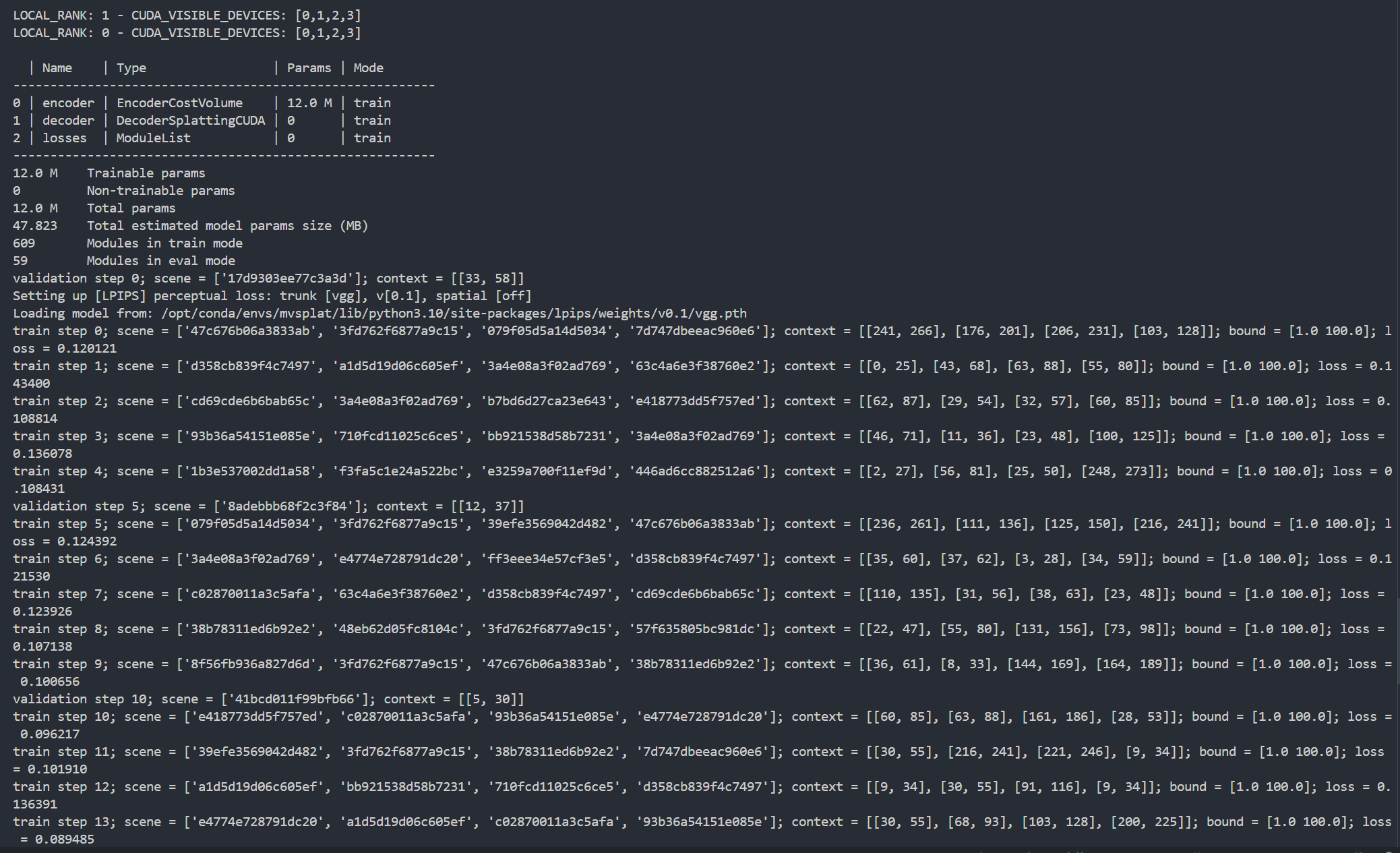


更改为

python -m src.main +experiment=re10k data\_loader.train.batch\_size=4

开始训练，但很慢：





控制台中输出两次 wandb 的信息

是因为正在使用分布式训练模式DDP，Distributed Data Parallel。

在这种模式下，每个 GPU 都会有一个独立的进程，每个进程都会独立地初始化 wandb，从而导致 wandb 的输出信息出现多次。

(mvsplat) root@localhost:~/SHARE/mvsplat# python -m src.main +experiment=re10k data\_loader.train.batch\_size=4

Saving outputs to /root/SHARE/mvsplat/outputs/2024-10-30/08-34-54.

wandb: Using wandb-core as the SDK backend. Please refer to https://wandb.me/wandb-core for more information.

wandb: Currently logged in as: xhw0313 (xhw-0313). Use `wandb login --relogin` to force relogin

wandb: Tracking run with wandb version 0.18.5

wandb: Run data is saved locally in /root/SHARE/mvsplat/wandb/run-20241030\_083455-a9yee15h

wandb: Run `wandb offline` to turn off syncing.

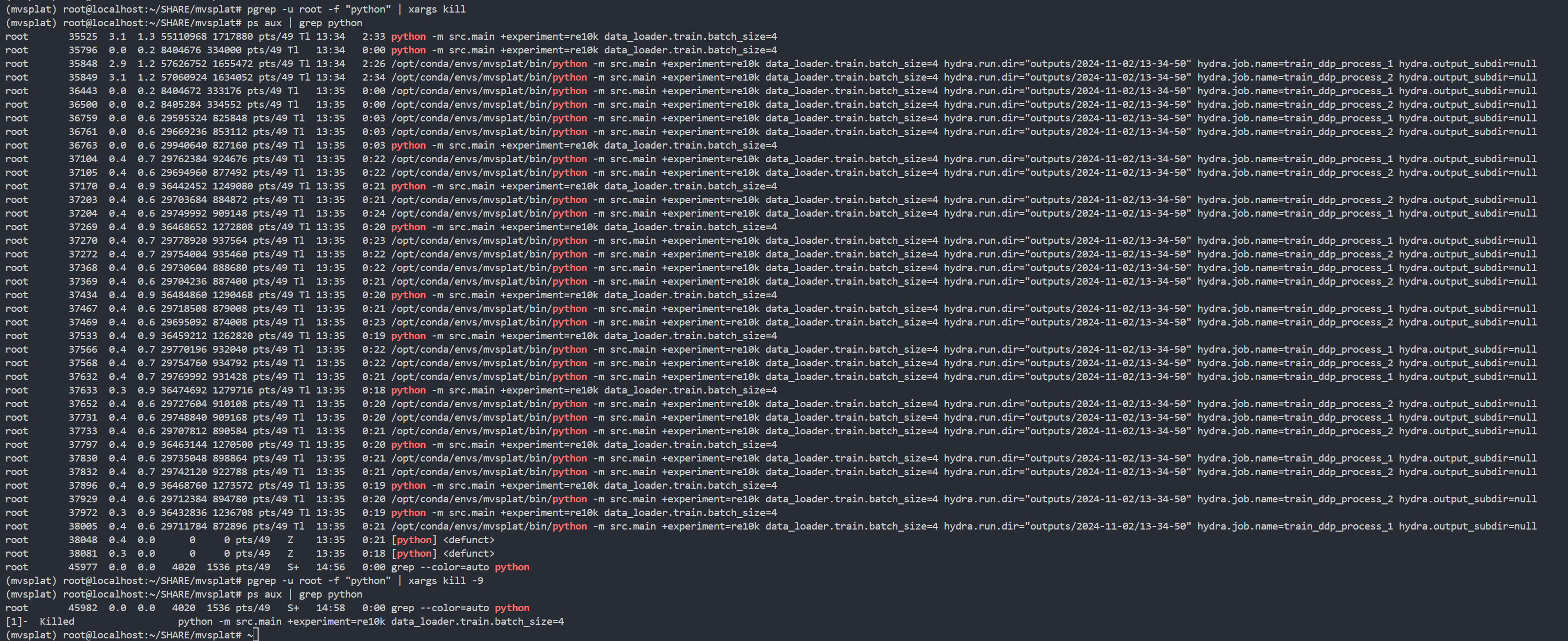
wandb: Syncing run faded-bat-19

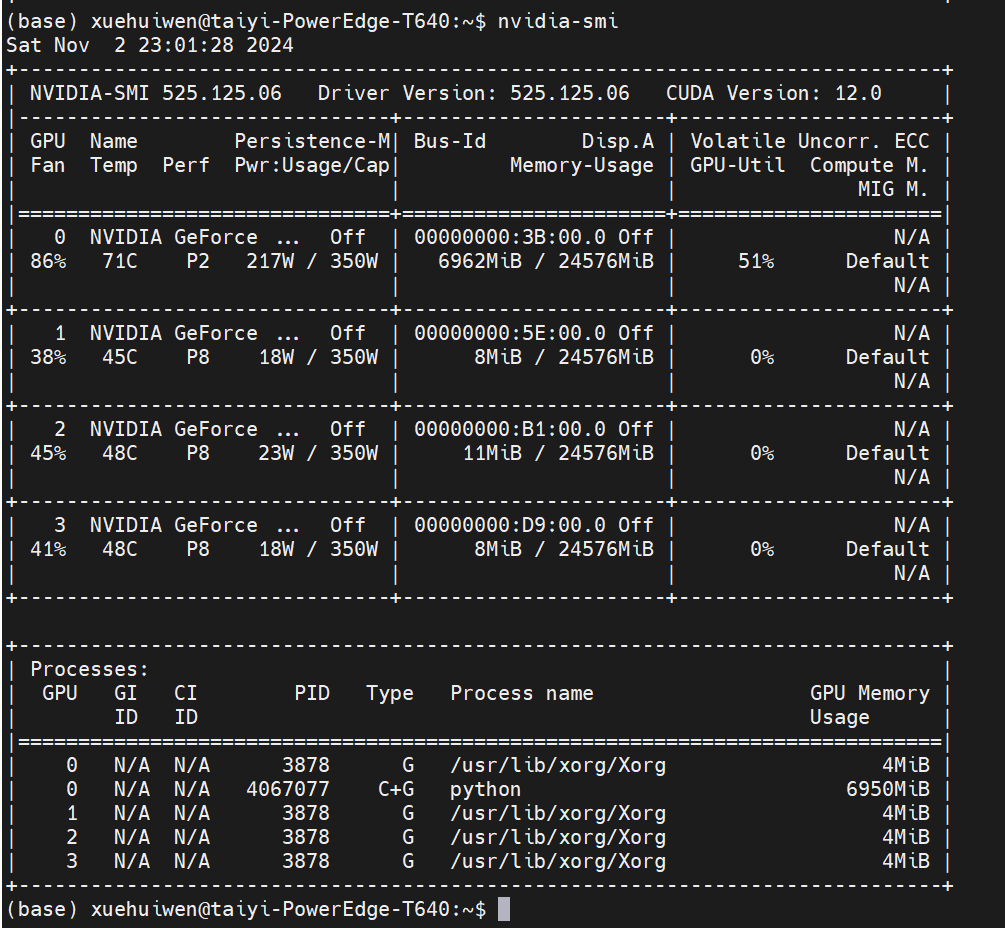
wandb: ⭐️ View project at https://wandb.ai/xhw-0313/mvsplat

wandb: 🚀 View run at <https://wandb.ai/xhw-0313/mvsplat/runs/a9yee15h>

## 使用tmux工具，避免中断

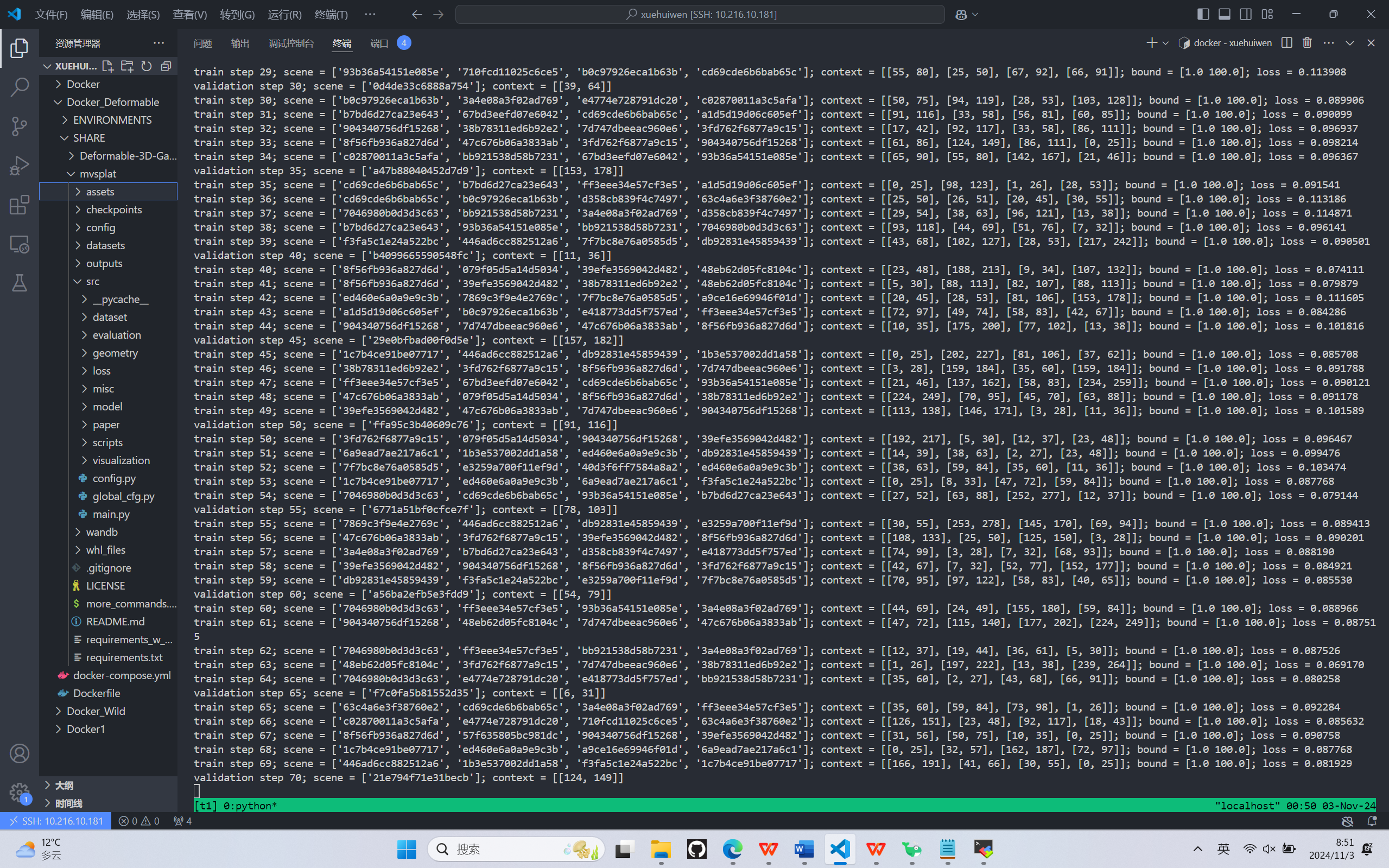
同时，查看进程，强制kill掉root下的所有python进程





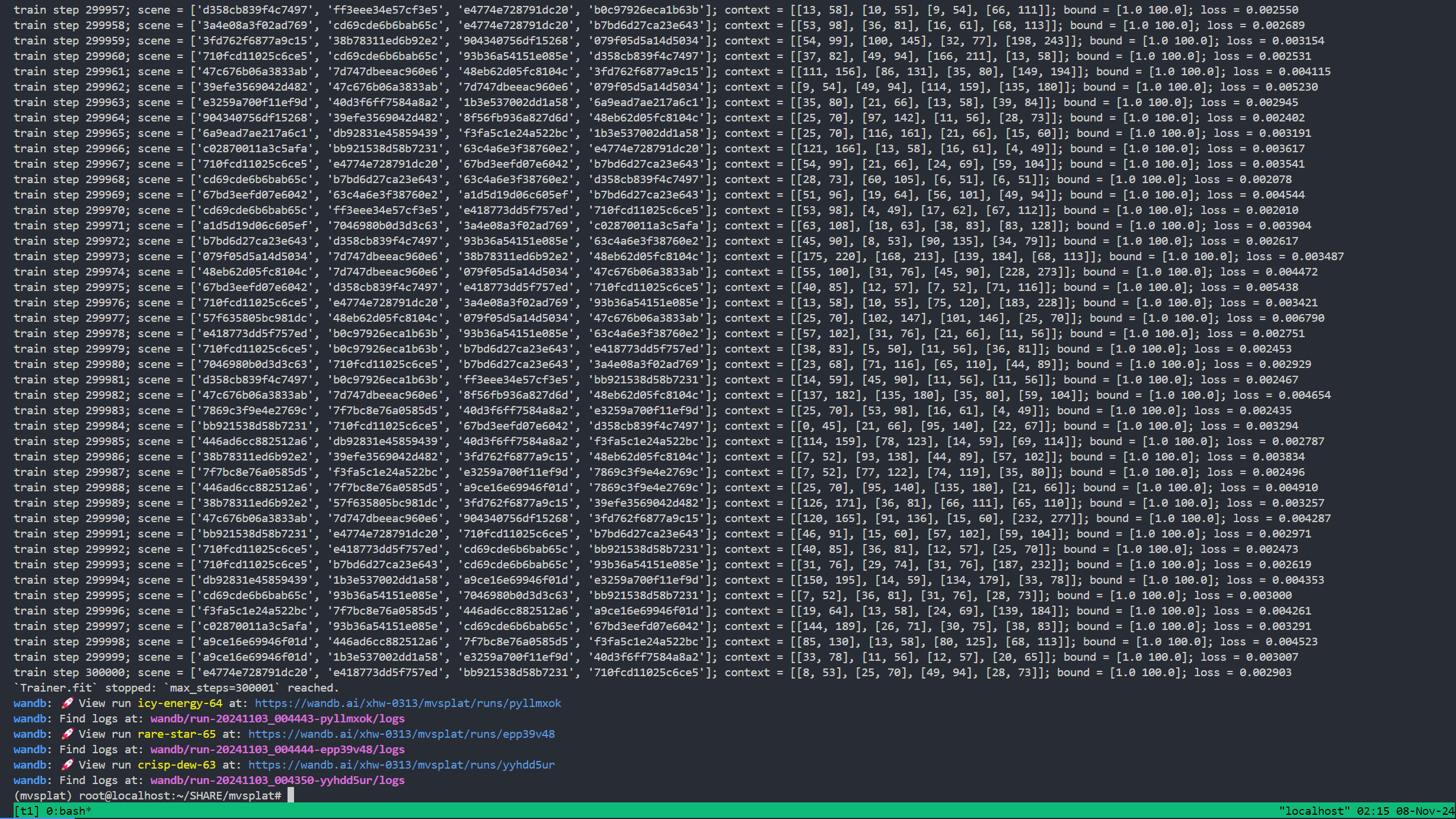
真棒诶！真的kill掉啦！

## 创建tmux t1进入---11.3大概8：50开始训练



大概1min 41steps

11.8号训练结束



推理结果

在re10k子数据集上：

psnr 21.47158213665611

ssim 0.7745077359048944

lpips 0.22853690248570943

encoder: 33 calls, avg. 0.12605281309647995 seconds per call

decoder: 99 calls, avg. 0.004771227788443517 seconds per call

psnr 21.47158213665611

ssim 0.7745077359048944

lpips 0.22853690248570943

encoder: 33 calls, avg. 0.1267075321891091 seconds per call

decoder: 99 calls, avg. 0.004877061554879852 seconds per call

psnr 21.47158213665611

ssim 0.7745077359048944

lpips 0.22853690248570943

encoder: 33 calls, avg. 0.1326753804177949 seconds per call

decoder: 99 calls, avg. 0.005132501775568182 seconds per call