

1. 在配置好环境后就可以进行对Isaacgym现有任务的训练和推断了

举例：

- Train: `python train.py task=Cartpole`
- Inference: `python train.py task=Cartpole`
`checkpoint=runs/Cartpole_08-02-30-59/nn/last_Cartpole_ep_100_rew__356.58_.pth`
`test=True`

train和inference在3080的机器上只需要1-2分钟

在训练的过程中，会将训练到一定程度的policy和最后继续训练的policy都存起来

2. 对**AMP**进行测试

`python train.py task=HumanoidAMP wandb_activate=True wandb_entity=yoyostudy`

`python train.py task=HumanoidAMP experiment=AMP_walk`

- 直接用训练好的CHECKPOINT作INFERENCE

`python train.py task=HumanoidAMP experiment=AMP_walk`
`checkpoint=runs/AMP_walk_08-06-07-17/nn/AMP_walk_08-06-07-19.pth test=True`

`python train.py task=HumanoidAMP experiment=AMP_dance`
`checkpoint=runs/AMP_walk_08-06-07-17/nn/AMP_walk_08-06-07-19.pth test=True`

用训练好的AMP_WALK直接做Amp_DANCE的inference效果很差，如果用ASE或者CALM应该就会有一个通用的LOW LEVERL POLICY

a). 可视化，‘`wandb_activate=True wandb_entity=XXXXX`’ 可以开启wandb可视化并且将训练log, reward, loss, performance, gpu等等呈现在我的wandb账号里面

通过看wandb里面的reward曲线，可以早点kill掉已经收敛好的policy

我的运行LOG, 已经PUBLIC

b). logs:

训练到某些程度的POLICY

=> saving checkpoint

'runs/AMP_walk_08-06-07-17/nn/AMP_walk_08-06-07-19_50.pth'

=> saving checkpoint

'runs/AMP_walk_08-06-07-17/nn/AMP_walk_08-06-07-19_100.pth'

。。。

在训练的时候，会不定期保存一些policy, 可以通过可视化发现一些早期保存的policy推理出的效果并不是很好

c). 用WALK训练的POLICY直接inference dance

```
python train.py task=HumanoidAMP experiment=AMP_dance
```

```
checkpoint=runs/HumanoidAMP_09-03-55-30/nn/HumanoidAMP_09-03-55-34_750.p  
th test=True
```

结果: 得到的REWARD为av reward: 293.110107421875

可视化结果: agent并没有办法产生DANCE的动作, 视频里的动作还是走路

原因分析: 1. 没有FEEDIN DANCE的MOTION DATA

d). 更改motion data to dance, 训练dance task

```
python train.py task=HumanoidAMP
```

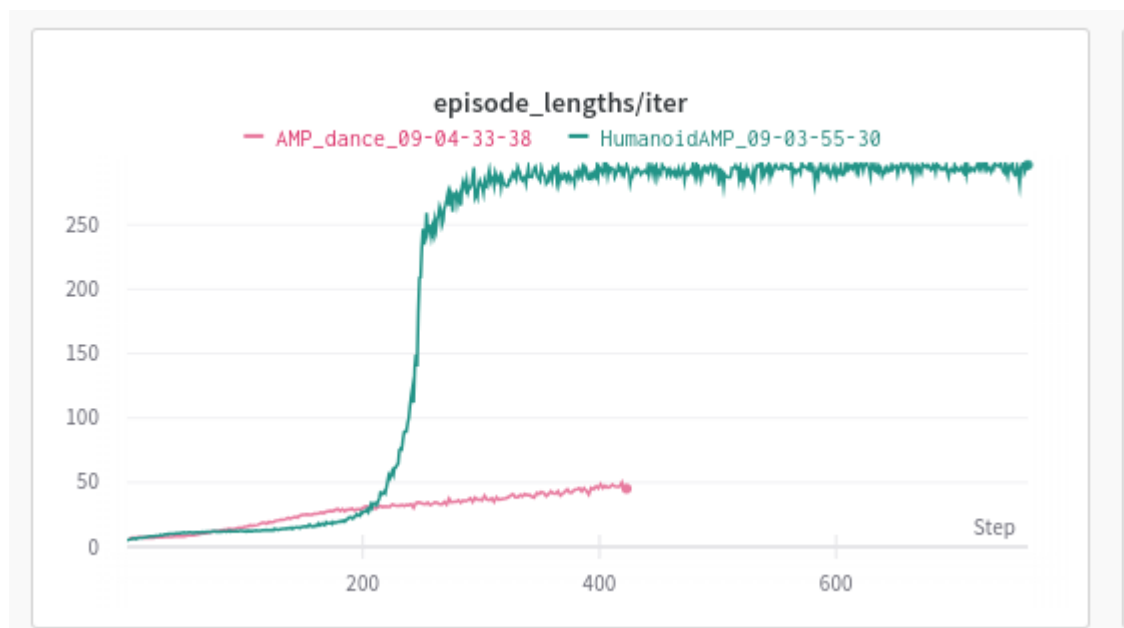
```
++task.env.motion_file=amp_humanoid_dance.npy experiment=AMP_dance
```

```
wandb_entity=yoyostudy wandb_activate=True
```

在不tune parameter: disc_grad_penalty的训练下比较不稳定, 并且增长缓慢

我的训练LOG: 已经PUBLIC

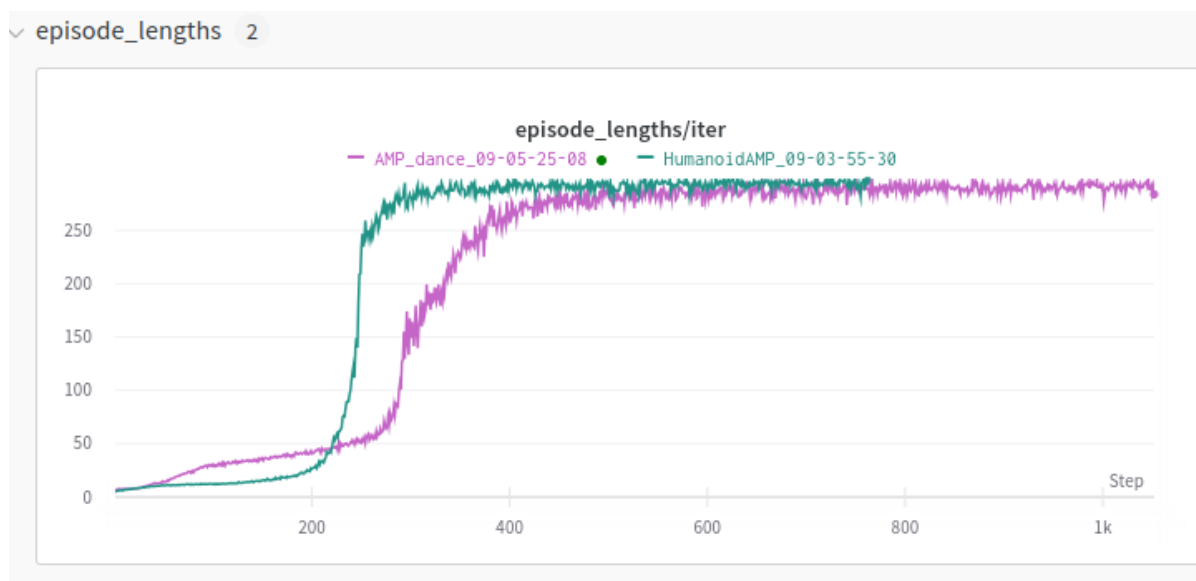
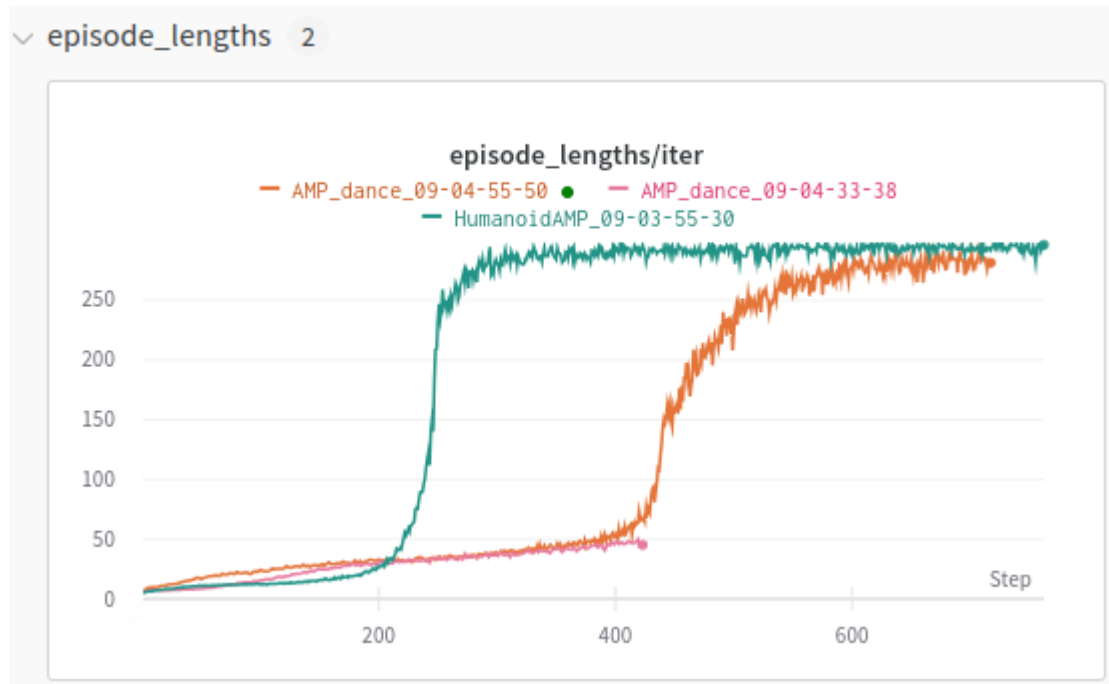
https://wandb.ai/yoyostudy/isaacgymenvs/runs/uid_AMP_dance_09-04-33-38/overview?workspace=user-yoyostudy



可以看到这里红色的新的MOTION DATA明显训练地比之前要慢, 从动作可视化中也可以看到AGENT几乎没有学习到跳舞, 一直在随意运动。

可能的原因: 1. 开了RENDER使得训练慢 2. 需要调参

关闭render重新训练



可以看相比之前的任务，加入新的MOTION DATA确实会使得训练变慢，同时开RENDER可能也会一定程度上影响到训练速度，之后为了比较应该都把RENDER关掉

```
python train.py task=HumanoidAMP experiment=AMP_dance
checkpoint=runs/AMP_dance_09-05-25-08/nn/AMP_dance_09-05-25-12_1500.pth
test=True
```

```
python train.py task=HumanoidAMP experiment=AMP_dance
checkpoint=runs/AMP_dance_09-05-25-08/nn/AMP_dance_09-05-25-12_1500.pth
test=True
```

e) .对新的MOTION DATA 需要TUNING disc_grad_penalty

对比两个设置文件HumanoidAMPPPO 和 HumanoidAMPPPOLowGP，可以看到唯一的区别是disc_grad_penalty参数不同，该参数的取值是在【0.1, 10】，文档里面建议从较大值开始往小改。

f). config file:

这里面提供了两个内置TASK CONFIG:

HumanoidAMP 和HumanoidAMPHands.yaml

hands 增加了left_hand 和 right_hand
这也是一个可以改动的点

```
python train.py task=HumanoidAMPHands train=HumanoidAMPPPOLowGP
experiment=AMP_cartwheel test=True
checkpoint=runs/AMP_cartwheel_12-00-36-08/nn/AMP_cartwheel_12-00-36-12_950.pt
h
```

```
python train.py task=HumanoidAMPHands train=HumanoidSAC
experiment=AMP_cartwheel wandb_activate=True wandb_entity=xxxxx headless=True
```