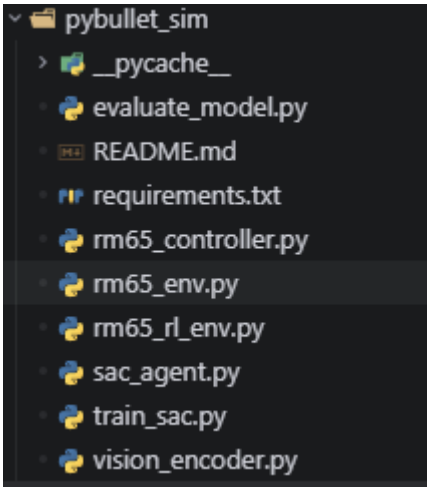


仿真环境

基于 PyBullet 的 RM65 机械臂仿真环境，集成了强化学习（SAC 算法）和传统运动学控制方法。

强化学习方案： 使用 SAC(Soft Actor-Critic)算法。 Actor 动作生成模型使用当前观察(摄像头输入+机械臂状态)作为输入，输出动作采样。Critic 网络也采样类似的架构。



1. 文件夹说明：
- evaluate_model.py**: 评估强化学习模型的效果。 参数说明：
--model 模型文件路径 --config 推理配置文件 --episodes 需要推理的次数 --gui 是否显示 pybullet 环境 --save-video 是否保存推理过程的视频为 mp4 格式 --compare-ik （与传统运动学对比，目前还未实现） --output-dir 相关输出路径（机械臂轨迹曲线等）
示例：python .\evaluate_model.py --model "C:\DiskD\Genesis-main\pybullet_sim\results\models\final_model.pth" --gui --save-video

rm65_controller.py: 传统运动学控制代码，提供运动学和动力学控制（可能存在错误）。

rm65_env.py: 构建机械臂的环境，基于 RM65 机械臂。主要功能包括：机械臂参数设置，摄像头位置设置等。 需要根据机械臂 URDF 文件目录替换代码中相应目录进行机械臂加载：

```
# 默认URDF路径
if urdf_path is None:
    self.urdf_path = os.path.join(
        os.path.dirname(__file__),
        "..", "genesis", "assets", "RM65", "urdf",
        "rm_65_b_description", "urdf", "rm_65_b_description.urdf"
    )
else:
    self.urdf_path = urdf_path
```

rm65_rl_env.py: 基于 rm65_env 类构建的强化学习环境，主要功能包括：观察空间定义，仿真步执行，重置环境，目标位置生成，获取观测值，计算奖励，环境渲染等。

sac_agent.py: SAC 算法构建，包含 Actor、Critic 模型构建，以及 ReplayBuffer 定义。

train_sac.py: 强化学习训练代码，参数说明： --config: 训练的配置文件 --env 机械臂环境的选择(目前只实现验证 RM65Reach) --vision 是否使用视觉观察作为模型输入 --timesteps 训练的步数 --eval-only 采用评估模型禁用模型训练 --model 模型的预训练权重 --output-dir 权重输出路径

vision_encoder.py: 视觉编码器构建，用于强化学习的视觉观察特征提取。目前是只构建了 ResNet-18 模型。

README.md: 项目说明文件，里面列举了代码的使用示例。

requirements.txt: 项目环境配置需求。