



关于创建具身智能开源数据集的倡议书

尊敬的全国各高校、科研院所、企业界同仁：

具身智能是目前智能科学前沿的方向，被认为是实现通用人工智能的必经之路。以美国为首的发达国家和 OpenAI、英伟达为代表的众多科技巨头，均将具身智能作为重要战略方向，纷纷投入巨资布局。与此同时，中国相关产业界也涌现出一批具身智能前沿探索科研机构与企业，如为斯坦福 Mobile ALOHA 项目提供硬件实施基础的松灵机器人等，积极参与并推动国内相关领域的科研课题与成果应用。近年来，机器人技术的发展日新月异，从工业制造到家庭生活，从深海探索到太空实验，机器人的身影越来越频繁地出现在各个领域。机器人是具身智能的物理载体，具身智能进一步赋予机器人“大脑、感官及体验能力”，使其具备通过与环境反复交互来持续学习及提升。

在大模型时代，通过增加数据量、扩大模型规模，可以实现模型性能的持续提升。为构建具身智能基础大模型，搭建具有规模化效应的具身智能应用平台，一个开源的大规模高质量的机器人感知操作数据集迫在眉睫。正如斯坦福大学的 ImageNet 推动了计算机视觉的研究一样，作为国家战略科技力量的重要组成部分，我们希望依托“中国算力网”和“启智”开源生态，在具身智能领域牵头创建一个具有同样影响力的开源数据集 ARIO (All Robots In One)。

目前，美国的谷歌公司已率先发布了 Open X-Embodiment 数据集和基于此数据集训练的具身智能控制基础模型 RT-X，表现出了在跨场景、多任务、跨平台等情况下良好的泛化性能并普遍超越此前基于特定场景和数据集下的技能水平。为推动国内具身智能领域的前沿探索与产业应用，鹏城实验室携手松灵机器人等科创企业共同倡议全国各高校、科研院所和企业，积极行动，共同打造国内首个大规模、多模态，并且涵盖多个场景、技能、任务、平台类型的具身智能数据集 ARIO。Open X-Embodiment 尽管先行一步，但仍存在一些不足，比如感知数据源比较单一，只包含图像，部分数据不规范，大部分机器人形式都是单臂的，限制了下游应用等。相比于 Open X-Embodiment，ARIO 将是全球首个包含五种模态（图像、点云、文本、触觉、听觉），并同时涵盖服务和工业等两大垂直领域，支持丰富应用场景的具身智能数据集。由此，我们呼吁各方：

1. 建立统一的机器人感知与操作数据收集和开源标准，确保数据的规范性。
2. 加强数据安全和隐私保护，确保开源数据不侵犯个人隐私和商业机密。
3. 鼓励高校、科研机构和企业积极参与数据采集、共享和合作研究，共建一个高质量的数据集生态，合力推动具身智能技术的创新发展。

让我们携手努力，共同推动 ARIO 的创建和开源工作，为具身智能和机器人行业的繁荣发展贡献智慧和力量！

此致

敬礼！

鹏城实验室、松灵机器人、中山大学、南方科技大学

2024 年 3 月 28 日

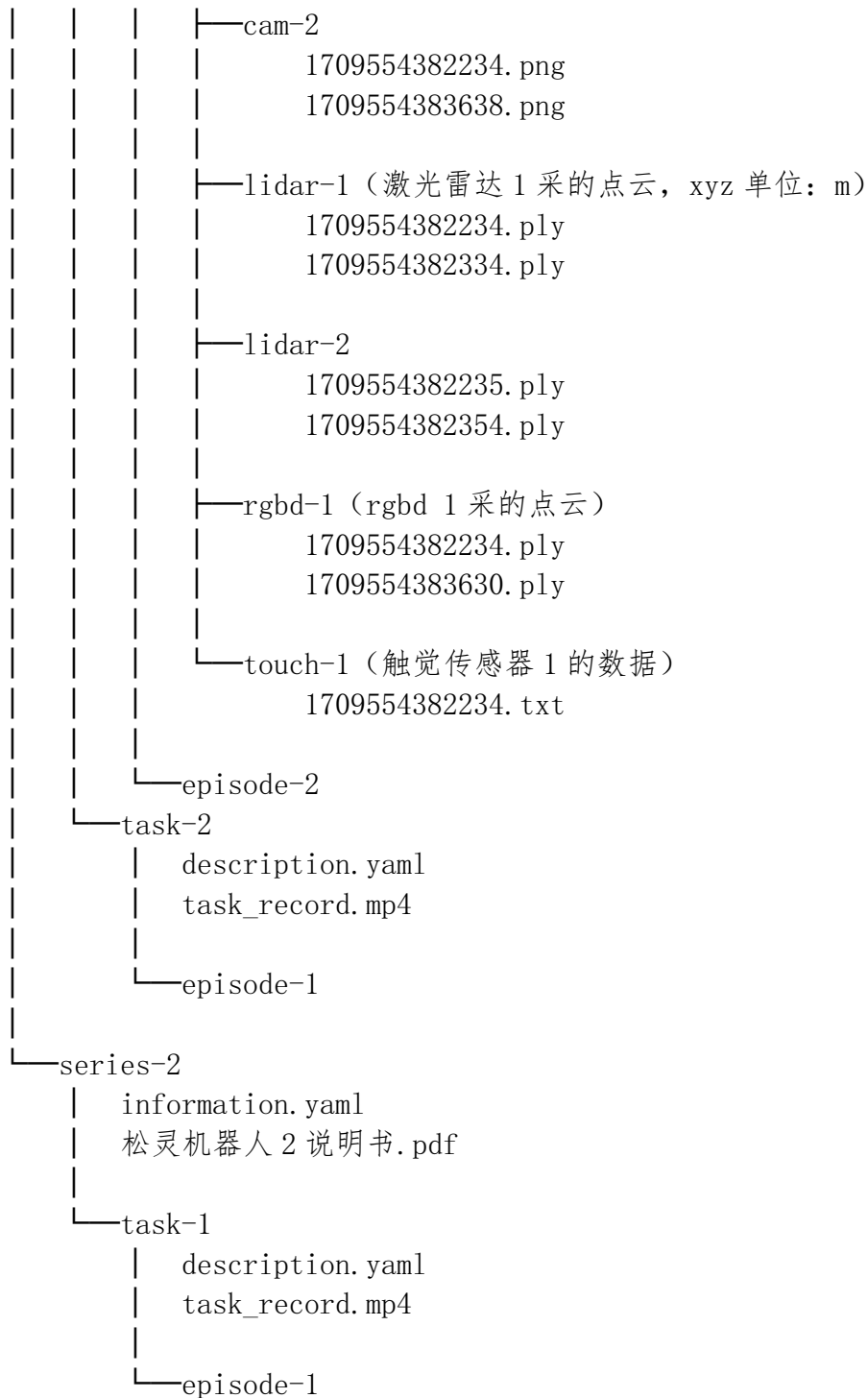
附录 1. 数据格式说明

1.1 数据结构方案

数据文件结构整体分为：collection——series——task——episode，collection 是指一次提交上传的数据集样本，可能包含不同的场景和机器人类型，series 是指同一个场景和同一个机器人采集的系列数据，如双臂机器人在厨房采集的系列数据，可能包含不同的任务，task 是一个具体的任务，比如抓取苹果，同一个任务可以重复采集多次，episode 是针对某一具体任务的一次完整采集过程。episode 下分传感器采集数据，各传感器可根据自己频率自行采集，但要以同一个时间戳为基准。示例文件结构如下：

collection（一次提交的数据集样本）

```
|   commit.yaml（提交者信息与声明）
|
|   └──series-1（同一个场景，同一个机器人）
|       |   calibration_1.yaml（相机 1 标定参数）
|       |   calibration_cam1_lidar1.yaml（相机 1 与 lidar 1 的标定参数）
|       |   IMU.pdf（IMU 传感器说明书）
|       |   information.yaml（场景描述，机器人信息，各传感器数量和信息）
|       |   touch.pdf（触觉传感器说明书）
|       |   松灵机器人说明书.pdf
|       |
|       └──task-1（一个任务，如：抓取苹果）
|           |   description.yaml（instruction）
|           |   task_record.mp4（每个任务的视频记录）
|           |
|           └──episode-1（一次完整采集过程）
|               |   audio-1-1709554382234.aac（音频数据）
|               |   base.txt（机器人本体运动数据）
|               |   IMU-1.txt（IMU 传感器数据）
|               |   left_master_arm_joint-0.txt（master 左臂关节 0 数据）
|               |   left_master_gripper.txt（master 左夹持器运动数据）
|               |   left_slave_arm_joint-0.txt（slave 左臂关节 0 数据）
|               |   left_slave_gripper.txt（slave 左夹持器运动数据）
|               |   pan_tilt.txt（头部云台数据）
|               |   right_master_arm_joint-5.txt（master 右臂关节 5 数据）
|               |   right_master_gripper.txt（master 右夹持器运动数据）
|               |   right_slave_arm_joint-5.txt（slave 右臂关节 5 数据）
|               |   right_slave_gripper.txt（slave 右臂的夹持器运动数据）
|               |
|               └──cam-1（camera 1 采的图像，相机采样帧率应 ≥30FPS）
|                   |   1709554382234.png
|                   |   1709554383638.png
```



1.2 数据采样方案

- (1) **场景**。数据采样场景宜多样化, 示例场景如, 室内: bedroom, kitchen, shopping mall, dining room, cafe, living room 等, 室外: park, residential district 等。
- (2) **动作**。机器人动作宜多样化, 示例动作如: pick、move、open、close、push、place、put、navigate、separate、point、insert、knock、drag、drop、wipe、assemble、turn on 等。

-
- (3) **必须采集。**各模态的数据中，文本指令、图像是每次采集都必须有的，如果是操作相关的任务还必须采集机器人末端和夹持器状态数据，如果是导航相关的任务，则必须采集机器人本体运动状态数据，其他数据在有条件的情况下应尽量采集。
 - (4) **默认单位。**数据记录的时间戳，是 Unix 时间戳，单位是 ms，**各传感器应使用同一时间戳基准。**角度单位默认是度。单臂机器人的手臂，可以视为右臂。位移单位默认是 m。力矩（扭矩）单位是 Nm。
 - (5) **task。**针对同一个 task，比如抓取物体，可以采集多个 episode，可以将物体以不同姿态，放在桌上的不同位置，甚至可以调整机器人本体位置，分别采集对应的 episode，建议一个 task 采集的 episode 数量不少于 10 次。
 - (6) **视频记录。**建议每个 task 用手机或相机以第三人称视角，针对某个 episode，采集一个 task_record.mp4，记录 task 相关的环境和操作信息，方便理解。
 - (7) **相机数据采集。**以相机 1 为例，在 cam-1 目录下，每帧图像以时间戳命名，格式为 png。information.yaml 文件中提到的相机数据都要采集。采集帧率不低于 30 FPS。
 - (8) **激光雷达点云数据采集。**以 lidar 1 为例，在 lidar-1 目录下，每帧点云以时间戳命名，格式为 ply，xyz 单位是 m。information.yaml 文件中提到的 lidar 数据都要采集。采集频率不低于 5 Hz。
 - (9) **rgbd 相机数据采集。**以 rgbd 1 相机为例，采集带颜色的点云格式数据，每个点的 xy 表示每个像素的坐标，取值范围取决于相机分辨率，以图片的左下角为零点，以 1280*720 相机为例，左下角 xy 为(0,0)，右上角为(1279,719)，每个点的 z 为深度 D，以相机为零点，越靠近相机，z 越小。每个点的 rgb 属性即为相机 rgb 值。在 rgbd-1 目录下，每帧点云以时间戳命名，点云格式为 ply。information.yaml 文件中提到的 rgbd 数据都要采集。采集帧率不低于 30 FPS。
 - (10) **触摸传感器数据采集。**以 touch 1 为例，在 touch-1 目录下，每帧数据以时间戳命名。information.yaml 文件中提到的 touch 数据都要采集。
 - (11) **本体移动数据的采集。**记录在 base.txt 中，格式为：每行记录一次采集数据，每行按顺序依次为时间戳，剩下的严格按照 information.yaml 中 base_control 的设置依次采集，各变量用空格隔开。采集频率最好不低于 30 Hz。
 - (12) **手臂末端夹持器的数据采集。**左右手分别记录在 left_master_gripper.txt 和 right_master_gripper.txt，若额外有 2 条 slave 手臂，记录文件命名可以拓展为 left_slave_gripper.txt，right_slave_gripper.txt。单臂的只记录 right_master_gripper.txt，格式为：每行记录一次采集数据，每行按顺序依次为时间戳，剩下的严格按照 information.yaml 中 endpoint_control 的设置依次采集，最后加上夹持器开合状态，各变量用空格隔开。夹持器打开是 0，闭合是 1，若有中间状态则映射为 0-1。采集频率最好不低于 30 Hz。
 - (13) **音频数据的采集。**在 episode 目录下，有多少个录音设备就采集多少个音频文件，录音是整个 episode 过程持续采集，主要是采集机器人在运动或操作过程中与环境触碰的声音，比如夹持器碰到杯子的声音，文件命名格式为“audio-设备序号-初始时间戳.aac”，如：audio-1-1709554382234.aac。
 - (14) **手臂关节数据的采集。**information.yaml 文件中，若 recorded_left_master_arm_joints 等相关变量不为空，则应采集相应关节的数据，例如 left_master_arm_joint-0.txt，采集的是 master 左臂 0 号关节的数据，格式为：每行记录一次采集数据，每行按顺序依次为时间戳，剩下的严格按照 information.yaml 中 joint_control 的设置依次采集，变量用空格隔开。其他手臂和关节数据可类比采集。采集频率最好不低于 30 Hz。
 - (15) **头部云台数据的采集。**information.yaml 文件中，若 pan_tilt 为 True，则应采集机

器人头部云台的数据，例如 pan_tilt.txt，格式为：每行记录一次采集数据，每行按顺序依次为时间戳，剩下的严格按照 information.yaml 中 pan_tilt_control 的设置依次采集，各变量用空格隔开。

- (16) **IMU 数据的采集。** information.yaml 文件中，若 IMU_num 大于 0，则应采集 IMU 的数据，例如 IMU-1.txt，格式为：每行记录一次采集数据，每行按顺序依次为时间戳，加速度 x，加速度 y，加速度 z，角速度 x，角速度 y，角速度 z，各变量用空格隔开，加速度单位为 m/s^2 ，角速度单位为度/s。同时应在“information.yaml”文件同一目录下附上 IMU 数据手册，如：'IMU.pdf'，并在机器人说明书上描述 IMU 的 xyz 正方向对应的机器人方向。
- (17) **关于人形机器人或四足机器狗等形态的数据采集。** 本体中心的运动，可以视为 base，参考 base.txt 的格式采集，其他如腿部/腰部等数据，可以参考末端执行器或手臂关节的格式采集，以“left_foot.txt”、“left_foot_joint-0.txt”等命名数据文件。
- (18) **命名规范。** 各数据文件和目录的命名应严格遵循格式要求，可以参考示例文件，方便后续数据处理。
- (19) **说明书。** 机器人平台相关的说明文档，可附在 collection（各 series 用同一款机器人）或 series（各 series 用不同机器人）目录下。
- (20) **咨询。** 有问题可以发邮件咨询 yeh@pcl.ac.cn。

1.3 配置文件详解

1.3.1 commit.yaml

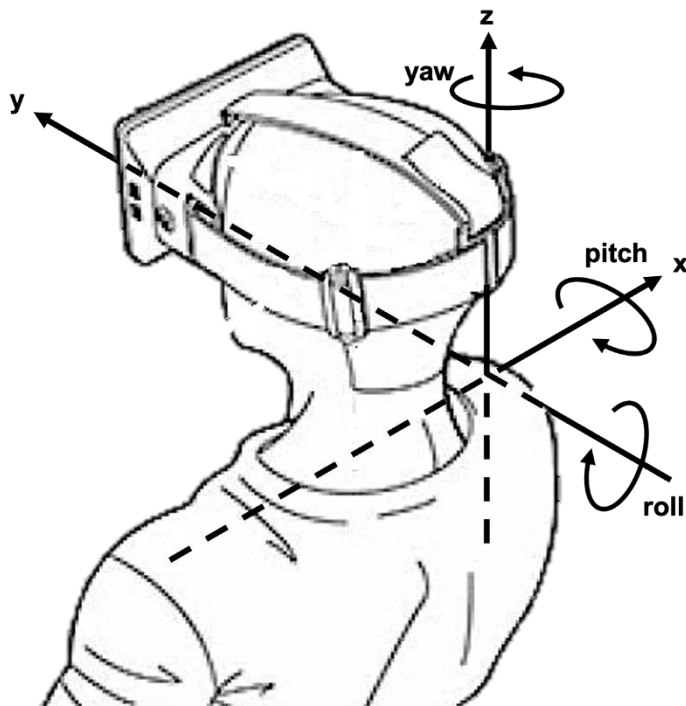
- author_name: 提交人姓名，如：'David'
- work_organization: 提交人单位，如：'PCL'
- author_email: 提交人邮箱，如：'yh1David@pcl.ac.cn'
- role: 提交人职位，如：'engineer'
- dataset_name: 此次提交数据集名称，如：'songling'
- license: 数据集开源 license，应支持在署名情况下可商用，可填：'CC BY 4.0', 'MIT license', 'Unsure'
- PII_exclude: 数据集是否排除了可识别个人身份的信息，必须为 True，相关责任由上传者承担
- thirdparty_consent: 如果数据集包含了第三方数据，请确保得到第三方授权，必须为 True，相关责任由上传者承担
- healthy_content: 数据集内容应确保健康，不能包含涉及暴力、毒品、虐待等不健康主题的内容，必须为 True，相关责任由上传者承担

1.3.2 information.yaml

- series_name: 系列名称，可根据平台、场景、agent 形态命名，如：'Songling_kitchen_bimanual'
- scene: 场景描述，室内/室外，厨房/客厅……，如：'indoor, kitchen'
- is_simulated: 真实数据还是仿真数据，True 表示仿真，False 表示真实
- morphology: 机器人形态，可多选，如：['bimanual', 'wheeled']，候选范围：
-'bimanual': 双臂

- 'single_arm': 单臂
- 'AGV': 自动导引车
- 'quadrupedal': 四足的
- 'wheeled': 轮式的
- 'drone': 无人机
- 'humanoid': 人形机器人

- num_joints_per_arm: 每条手臂的关节数, 不含夹持器, 范围 ≥ 0 , 没有手臂的填 0
- gripper: 夹持器的控制是张开/闭合二元状态还是连续状态, 候选范围:
 - 'binary': 二元的
 - 'continuous': 连续的
- same_coordinate: 对于采集的执行器末端数据或者本体移动数据, 其坐标系方向是否与预设坐标系方向相同, 如相同, 则为 True。预设坐标系如下图所示, 机器人右边为 x 正方向, 前方为 y 正方向, 头顶上方为 z 正方向, 坐标系原点的位置在右臂关节的肩部, 即右手臂与身体的连接部位的中心。机器人坐标系可以与预设坐标系不同, 但必须都是遵循右手定则的坐标系, 并且 x-pitch, y-roll, z-yaw, 这三者的对应关系应一致, 即不能 x 对应 roll。



- endpoint_transform 和 origin_offset: 若 same_coordinate 为 False, 则需指定末端执行器的坐标系到预设坐标系的旋转矩阵和平移坐标, 如: endpoint_transform 为 $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$, origin_offset 为 $[0, 0, 0]$ 。计算公式为: 采集的末端坐标 \cdot endpoint_transform + origin_offset = 预设坐标系下的末端坐标。origin_offset 如不能提供精确值 (误差小于 1cm), 可以不提供, 设置为: $[\]$, 没有手臂的机器人也为: $[\]$ 。机器人说明书里应附上末端执行器坐标系的方向。
- base_transform: 若 same_coordinate 为 False, 则需指定本体移动的坐标系到预设坐标系的转换矩阵, 如: $\begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ 。计算公式为:

采集的 本体移动坐标 \cdot base_transform = 预设坐标系下的 本体移动坐标。
机器人说明书里应附上 本体移动坐标系的方向。

- IMU_transform: 若 same_coordinate 为 False, 则需指定 IMU 的坐标系到预设坐标系的转换矩阵, 如: $[[1,0,0],[0,1,0],[0,0,1]]$ 。计算公式为: 采集的 IMU 坐标 \cdot IMU_transform = 预设坐标系下的 IMU 坐标。机器人说明书里应附上 IMU 坐标系的方向。
- endpoint_control: 执行器末端运动坐标形式, 从如下可选范围中挑选要采集的数据内容, left_gripper.txt、right_gripper.txt 中每行的数据内容与顺序, 除首位的时间戳和末位的开合状态外, 应严格与 endpoint_control 的内容相对应, 如: ['absolute', 'x', 'y', 'z', 'pitch', 'roll', 'yaw'], 表示要以绝对值记录末端的 xyz 和 pitch, roll, yaw 角度。可选范围:
 - 'absolute'/'relative': 绝对值/相对值, 绝对值是指每次采集的运动数据是基于预设坐标系或其他本体坐标系的值, 相对值是指, 每次采集的运动数据都是在上一次运动基础上的变化值。这个概念只针对位置和姿态, 如有位置或姿态数据, 则 'absolute'/'relative' 两者必须选一个, 不针对速度、角速度、力矩。
 - 'x', 'y', 'z': 位置坐标
 - 'pitch', 'roll', 'yaw': 姿态, 'pitch', 'roll', 'yaw' 依次分别对应 xyz 轴的旋转
 - 'vx', 'vy', 'vz': xyz 方向的运动速度
 - 'wx', 'wy', 'wz': xyz 方向的旋转速度
 - 'tx', 'ty', 'tz': xyz 方向的力矩
 - 'none': 没有手臂, 或无法移动, 或不记录相关数据
- base_control: 机器人本体运动坐标形式, 从可选范围中挑选要采集的数据内容, base.txt 中每行的数据内容与顺序, 除首位的时间戳外, 应严格与 base_control 的内容相对应, 如: ['relative', 'x', 'y', 'yaw'], 表示要以相对值记录本体的 xy 位移和 yaw 方向旋转角度。又如: ['vx', 'vy', 'wz'], 表示要记录 xy 方向的速度和绕 z 旋转角速度。可选范围参考 endpoint_control。
- joint_control: 机器人手臂关节运动形式, 从可选范围中挑选要采集的数据内容, left_arm_joint-0.txt 等关节数据文件中, 每行的数据内容与顺序, 除首位的时间戳外, 应严格与 joint_control 的内容相对应, 如: ['absolute', 'pitch', 'wx', 'tx'], 表示要以绝对值记录关节的角度、角速度、力矩, 由于关节只有 1 个自由度, 默认都是 x 方向。可选范围参考 endpoint_control。
- pan_tilt: 机器人头部是可动云台则为 True, 否则为 False。
- pan_tilt_control: 若 pan_tilt 为 True, 则需设置头部云台运动坐标形式, 从可选范围中挑选要采集的数据内容, pan_tilt.txt 中每行的数据内容与顺序, 除首位的时间戳外, 应严格与 pan_tilt_control 的内容相对应, 如: ['absolute', 'pitch', 'yaw'], 表示要以绝对值记录云台的俯仰角和偏航角。坐标系应按照预设坐标系的规定。可选范围参考 endpoint_control。
- endpoint_origin: 若末端执行器用的是绝对坐标, 这里指明坐标原点在机器人上的位置, 可选范围:
 - 'shoulder': 手臂根部
 - 'middle': 机器人本体中部
 - 'head': 机器人头部

-
- 'bottom': 机器人底部
 - 'none': 非绝对坐标或不涉及此信息
 - action_frequency: 机器人控制动作频率, 单位: Hz, 如: 30
 - blocking_control: 是否阻塞式控制, 即必须先完成当前指令的动作才能执行下一个指令, True 表明是阻塞式
 - arm_num: 要采集的机器人手臂数量, 如: 4, 手臂数量超过 2, 应在机器人说明书上描述各手臂与人之间的操控关系。
 - arm_operation_mode: 机器人手臂操作模式, 列表长度应等于手臂数量, 元素顺序为: [master 左, master 右, slave 左, slave 右, ...], 如: ['kinesthetic', 'kinesthetic', 'teleoperation', 'teleoperation'], 元素可选范围:
 - 'kinesthetic': 人直接把机械手移到指定位置或把机器人推到指定位置
 - 'manipulation': 人通过遥控器操作
 - 'teleoperation': 人手上或身上有运动传感器采集人动作数据, 机器人跟随采集数据运动
 - 'imitation': 机器人通过激光雷达/相机等传感器学习/模仿人动作, 机器人和人没有直接接触
 - camera_num: 相机数量, 有多少个相机, 下面就要根据示例填多少个相机的信息, 如: 2
 - cam_view: 相机安装视角, 列表长度应等于相机数量, 如: ['ego-centric', 'third-person'], 可选范围:
 - 'ego-centric': 第一人称, 一般指安装在头顶的相机
 - 'third-person': 第三人称, 一般指安装在桌子或周围环境的相机
 - 'left_wrist': 安装在机器人左臂腕部或执行器末端的相机
 - 'right_wrist': 安装在机器人右臂腕部或执行器末端的相机
 - cam_calibration_file: 相机的标定参数文件, 列表长度应等于相机数量, 如: ['calibration_1.yaml', ''], 参数文件放在与 information.yaml 相同的路径, 没有标定文件可以为: ''
 - lidar_num: 激光雷达数量, 有多少个激光雷达, 下面就要根据示例填多少个激光雷达的信息, 如: 2
 - lidar_position: 激光雷达安装位置, 列表长度应等于雷达数量, 如: ['head', 'bottom'], 可选范围:
 - 'head': 机器人头部
 - 'middle': 机器人中间
 - 'bottom': 机器人底部
 - cam1_lidar1_calibration_file: 相机 1 和 lidar1 的相对位置标定参数文件, 如: 'calibration_cam1_lidar1.yaml', 文件放在与 information.yaml 相同的路径, 如无, 可填: ''。如有其他相机或 lidar 的相对位置标定文件也可参考此格式填上, 如: cam2_lidar2_calibration_file: 'calibration_cam2_lidar2.yaml'。
 - rgb_d_num: rgb_d 相机数量, 有多少个 rgb_d 相机, 下面就要根据示例填写多少个 rgb_d 相机信息, 如: 1
 - rgb_d_view: rgb_d 相机安装视角, 列表长度应等于相机数量, 如: ['third-person'], 可选范围:

-
- 'ego-centric': 第一人称, 一般指安装在头顶的相机
 - 'third-person': 第三人称, 一般指安装在桌子或周围环境的相机
 - 'left_wrist': 安装在机器人左臂腕部或执行器末端的相机
 - 'right_wrist': 安装在机器人右臂腕部或执行器末端的相机
 - touch_num: 触摸传感器数量, 有多少个传感器, 下面就要根据示例填多少个信息, 如: 2
 - touch_position: 触摸传感器安装位置, 列表长度应等于传感器数量, 如: ['left, gripper', 'right, gripper']
 - touch_manual: 触摸传感器说明书, 如用了触摸传感器, 应在“information.yaml”文件同一目录下附上说明书, 如: 'touch.pdf', 如无, 可填: ''
 - recorded_left_master_arm_joints: 要采集运动数据的 master 左臂关节点序号, 这里不包括夹持器的运动数据, 这里的序号要跟实际采集的数据一致, 数值越大, 表示该关节越靠近末端, 如: [0, 1, 2, 3, 4, 5], 如不采集关节信息, 可为: []。
 - recorded_left_slave_arm_joints: 要采集运动数据的 slave 左臂关节点序号, 详情同上, 如果只有 1 条左臂, 则此项为 []。
 - recorded_right_master_arm_joints: 要采集运动数据的 master 右臂关节点序号, 说明同上, 如: [0, 1, 2, 3, 4, 5]。
 - recorded_right_slave_arm_joints: 要采集运动数据的 slave 右臂关节点序号, 详情同上, 如果只有 1 条右臂, 则此项为 []。
 - audio_num: 音频传感器数量, 如: 1。
 - audio_frequency: 音频采样频率, 列表长度应等于传感器数量, 单位 Hz, 如: [48000], 如没有音频, 可以填: [0]。
 - IMU_num: IMU 数量, 有多少个 IMU, 下面就要根据示例填多少个 IMU 的信息, 如: 1。
 - IMU_position: IMU 安装位置, 列表长度应等于 IMU 数量, 如: ['bottom'], 可选范围:
 - 'head': 机器人头部
 - 'middle': 机器人中间
 - 'bottom': 机器人底部

1.3.3 description.yaml

- instruction_EN: 给机器人的英文动作指令, 指令应与机器人动作一致, 必填, 如: 'pick up the apple from the table and put it into the bowl'
- instruction_CH: 给机器人的中文动作指令, 意思应与英文指令一致, 可以不填, 如: '将苹果从桌子上拿起, 放入碗中', 或不填留空: ''
- skills: instruction 中涉及的机器人技能, 就是其中的动词, 如: ['pick', 'put']

1.4 数据集上传与提交

- (1) 数据开源平台为启智社区, 主页为: <https://openi.pcl.ac.cn/>。可参考帮助文档: <https://openi.pcl.ac.cn/docs/index.html#/>, 创建项目, 上传数据集。数据集打包文件的

格式必须为 .zip 或 .tar.gz, 平台限制单个数据集文件大小不超过 200G。

- (2) 提交者应先在启智社区注册, 注册成功后, 将用户名发送到 yeh@pcl.ac.cn, 备注: 加入 ARIO 组织。加入成功后会收到回复邮件。
- (3) 加入 ARIO 组织后, 提交者在启智社区创建项目, 项目路径选在 ARIO 下, 如下图所示。
- (4) 项目创建成功后, 即可在项目下上传数据集文件, 此时应将前述 collection 目录及其包含文件统一打包上传。
- (5) 应当注意: 上传者应确保上传的数据不含个人信息, 不含涉及黄赌毒的内容, 如涉及第三方应获得第三方授权。因数据公开造成的侵权、违法责任由上传者负责。参与 ARIO 组织内项目的数据集上传即视为同意此规定。

1.5 我们的数据格式与 Open X-Embodiment 区别

我们的格式保留了原始数据内容, 并对每项数据都保存了时间戳, 方便用户基于不同的时间间隔进行采样和处理, 用户也可以基于简单的转换程序将我们的数据转换成 Open X-Embodiment 或其他格式来使用, 具有可扩展性和通用性, 能兼容更多的平台。我们的数据也增加了点云、触觉等更多的模态, 并且可以自定义执行器末端、本体、手臂各关节等多个运动对象的数据采集, 也能自定义位置、姿态、速度、角速度、力矩等多种变量的采集, 能支持实现更复杂的任务和更为灵活的机器人控制。

创建项目

项目名称 *

请输入中文、字母、数字和_., 最多100个字符。

项目路径  ARIO

路径只允许字母、数字和_., 最多100个字符。

项目地址: https://openi.pcl.ac.cn/ARIO/pcl_robot_datasets.git

可见性 将项目设为私有

只有组织所有人或拥有权利的组织成员才能看到。

项目描述

模板

任务标签

.gitignore

授权许可

自述

初始化存储库 (添加 .gitignore、许可证和自述文件)

默认分支

我承诺此仓内容不违反任何国家法律法规, 仓库使用过程中遵守OpenI启智社区管理规定和平台使用规则, 不进行恶意攻击、挖矿等任何违法或扰乱平台秩序的信息发布和相关行为。更多信息请参考 [OpenI启智社区平台使用协议](#)

创建项目

取消