双目视觉与 IMU 数据格式说明

标准化采集数据包

文档版本 1.4

1 格式概述与目录结构

本格式用于组织一次采集片段的双目图像、逐帧时间索引、六轴惯性测量、标定参数及元数据。数据包以清单作为入口，通过流标识、时钟域和坐标系标识建立各文件之间的关联，供数据接入、离线处理和交付校验使用。

数据包包含一个 head 成员、左目视频、右目视频和 IMU 三路数据流，数据模态为 ColorRGB 和 IMU 两种，共 11 个文件。IMU 采用前左上坐标系，保留传感器原点和加速度中的重力分量，额外时间偏移为 0。图像尺寸、频率、记录数和时间范围分别由清单字段给出。

```
数据包根目录/
 Manifest.json
 MetaData/
  metadata.json
 head/
  ColorRGB/
   front_left/
    video.mp4
    timestamps.csv
   front_right/
    video.mp4
    timestamps.csv
  IMU/
   imu.csv
  Calibration/
   rgb-front-left/intrinsic.json
   rgb-front-right/intrinsic.json
   extrinsics/camera_camera_extrinsics.json
   extrinsics/camera_imu_extrinsics.json
```

1 1 文件职责

Manifest.json 描述交付结构与数据摘要；metadata.json 描述采集身份和设备关联。每路视频对应独立的时间索引；两份内参文件分别描述左右相机，两份外参文件描述双目及相机与 IMU 之间的静态变换。

1 2 通用序列化约定

JSON 使用对象、数组、字符串、数值、布尔值和 null 表达数据；CSV 首行为列名，以逗号分隔，不插入空白数据行。文本采用 UTF-8。数据数量均不含表头。JSON 的纳秒时间戳以十进制整数字符串保存，CSV 中使用十进制整数文本；纳秒时间偏移量则使用整数。读取时应避免整数精度损失。

清单中的 path 相对数据包根目录，使用正斜杠。读取文件时将数据包根目录与 path 拼接。标识符、文件名和字段名区分大小写。null 表示该元数据项为空，读取时保留为空值。

2 数据包清单

Manifest.json 为数据包入口，描述三路数据流、关联文件、时钟和坐标系。表中点号表示嵌套路径，斜杠分隔的名称为独立字段。

字段	类型	含义
format	string	清单格式标识，值为 NormalizedDataManifest.v1
schema_revision	string	清单结构修订号；与 format 共同用于选择解析规则
artifact_mode	string	交付产物模式，production 表示生产交付数据
profile	object	目标数据配置，含 id 和 version 两个字符串
source_identity	object	采集来源身份，含 kind、namespace、value 三个字符串
tenant_id	string	租户标识
unit_id	string	数据单元标识
members	object[]	采集成员列表；每项含 id、path、role 三个字符串
clocks	object[]	时钟定义，见第 7 节
frames	object[]	坐标系定义，见第 6 节
streams	object[]	传感器数据流列表，见第 3 节
calibration_files	object[]	标定文件引用列表
metadata_file	object	采集元数据文件引用
storage	object	交付存储定位信息，见第 8 节

2 1 文件引用对象

字段	类型	约定
path	string	根目录相对路径，指向对应文件
role	string	文件角色：media、index、imu、intrinsics、extrinsics 或 metadata
sha256	string	对应文件完整字节内容的 SHA-256；256 bit，以 64 个十六进制字符表示，不带前缀
serialization_profile	object	CSV 引用中的序列化配置，含 id 和 version 字符串

calibration_files、metadata_file 以及各流 observed.files 使用文件引用。Manifest.json 本身不列入自身的哈希范围。引用文件发生任何字节级变化时，应重新计算哈希并同步更新清单。

2 2 标识关联

成员、时钟、坐标系和数据流的 id 在各自集合中唯一。member_id、clock_domain、frame_id 分别引用已声明的成员、时钟、坐标系；source_identity 标识采集来源。

3 数据流描述

streams 中每个对象对应一条传感器流。stream_role 区分左目、右目和惯性传感器，modality 表示数据类型。左右视频共用 ColorRGB 类型，通过各自的流标识和角色区分。

数据流	**stream_role**	**modality**	**row_schema**
左目视频	rgb-front-left	ColorRGB	rgbRow
右目视频	rgb-front-right	ColorRGB	rgbRow
惯性采样	imu-main	IMU	imuRow

字段	类型	含义
id	string	数据流唯一标识，在 streams 中唯一
sensor_id	string	传感器标识，用于与该传感器的标定信息关联
member_id	string	所属成员，关联 members[].id
stream_role	string	流的逻辑角色，取值对应上表
modality	string	数据模态；基础交付包含 ColorRGB 与 IMU 两种
frame_id	string	该流测量值所用坐标系，关联 frames[].id
clock_domain	string	时间戳所属时钟，关联 clocks[].id
row_schema	string	ColorRGB 使用 rgbRow；IMU 使用 imuRow
expected.required	boolean	三路流均为 true，表示交付时必须包含该流
expected.nominal_rate_hz	number	标称帧率或采样率，单位 Hz；不替代实际时间间隔
observed.status	string	present 表示该流的数据已交付
observed.reason	string | null	状态备注，正常交付时为 null
observed.count	integer	非空流的帧数或采样数，大于 0，不含 CSV 表头
observed.first_timestamp_ns	string	第一条记录的硬件时间戳，十进制纳秒整数文本
observed.last_timestamp_ns	string	最后一条记录的硬件时间戳，十进制纳秒整数文本
observed.files	object[]	该流的文件引用，引用结构见第 2 节
media	object	视频媒体描述，见第 4 节；IMU 流不使用该对象

expected 和 observed 均为对象，分别容纳期望配置和实际交付摘要。每路 ColorRGB 的 observed.files 含 media 视频和 index 索引；IMU 含 imu 文件。CSV 序列化配置 ID 分别为 csv-rgb-index 和 csv-imu-imu，并由 version 声明版本。

4 视频索引与惯性采样格式

4 1 视频文件

左右视频独立存储，采用 MP4 与 H.264。按呈现顺序解码视频，第 n 帧关联 frame_index = n 的索引记录。使用同一视频流的 PTS 和 time_base 核对呈现时间，标称帧率读取 expected.nominal_rate_hz。

media 字段	类型	含义
container / codec	string	容器 mp4 与编码 h264
width / height	integer	单路图像宽度与高度，单位 px，均大于 0
pixel_format	string	像素格式；解码时同时核对码流的色彩范围
time_base	integer[2]	时间基 [分子,分母]，两者均大于 0

4 2 逐帧时间索引

timestamps.csv 使用固定列名 frame_index,timestamp_ns,pts。每条记录描述一个视频帧，视频帧与索引记录一一对应。

列名	类型与单位	约束与语义
frame_index	integer	从 0 开始，连续递增；末值等于帧数减 1
timestamp_ns	int64，ns	硬件采集时间，用于同一时钟域内的传感器对齐
pts	integer，时间基刻度	视频呈现时间戳，按该流 media.time_base 换算为秒

索引示例采用 time_base = [1,15360]，256 个刻度对应 1/60 秒。

```
frame_index,timestamp_ns,pts
0,1000000000,0
1,1016666667,256
2,1033333334,512
```

实际数据逐行读取 timestamp_ns 和 pts，按各自时间基换算。

4 3 IMU 六轴采样

imu.csv 使用固定列名 timestamp_ns,ax,ay,az,gx,gy,gz。按列名读取六轴通道，所有测量值均在前左上坐标系中表达。

列名	类型与单位	语义
timestamp_ns	int64，ns	采样硬件时间，与流的 clock_domain 一致
ax / ay / az	float64，m/s^2	IMU 输出坐标系各轴加速度计测量值
gx / gy / gz	float64，rad/s	绕 IMU 输出坐标系各轴的角速度，按右手规则

```
timestamp_ns,ax,ay,az,gx,gy,gz
1000000000,0.0,0.0,9.81,0.0,0.0,0.0
```

加速度保留重力分量，gravity_removed 固定为 false。CSV 存储数值文本；将 timestamp_ns 读为 int64，将六轴测量值读为 float64。六轴值均为有限数。

5 相机内参格式

左右相机各有一份 intrinsic.json。每份文件明确关联传感器和图像坐标系；参数对应的是该流记录的图像尺寸与像素约定。

字段	类型	含义
sensor_id	string	对应相机传感器标识
frame_id	string	对应相机光学坐标系标识
image_size	integer[2]	图像尺寸 [width,height]，单位 px
projection_model	string	投影模型，基础格式使用 pinhole
projection_model_version	string	投影模型版本
model_parameters	object	投影参数 fx_px、fy_px、cx_px、cy_px
distortion_model	string	畸变模型，基础格式使用 equidistant
distortion_model_version	string	畸变模型版本
distortion_parameters	object	畸变系数 k1、k2、k3、k4
pixel_convention	string	center_of_top_left_pixel_is_0_0 表示左上角像素中心为 (0,0)
calibration_version	string	该标定内容的版本标识
valid_for_source_identity	string	原样保留

5 1 参数与模型

model_parameters 中的 fx_px、fy_px、cx_px、cy_px 及 distortion_parameters 中的 k1 至 k4 均为有限 number。fx_px、fy_px 为像素单位焦距且大于 0；cx_px、cy_px 为主点坐标；k1 至 k4 为无量纲畸变系数。相机矩阵如下，畸变计算按 equidistant 模型版本解释。

```
K = [[fx_px,     0, cx_px],
   [    0, fy_px, cy_px],
   [    0,     0,     1]]
```

按照 k1、k2、k3、k4 的顺序组成畸变参数 D，并将 K、D 传入 equidistant 鱼眼模型的投影或去畸变接口。图像缩放时，分别用水平和垂直缩放比例更新 fx、fy、cx、cy；从左上角 (u0,v0) 裁剪时，将主点更新为 (cx-u0,cy-v0)。

5 2 标定关联

先按 stream_role 选择左目或右目流，再将流的 sensor_id、frame_id 与对应 intrinsic.json 中的同名字段匹配；image_size 与该流的 media.width、media.height 对应。按此关系加载左右内参。

calibration_version 标识该标定文件的内容版本。按 Manifest.calibration_files 中的 path 定位标定文件，结合 sensor_id、frame_id 及对应采集数据核对标定是否适用。几何计算读取 model_parameters 与 distortion_parameters 中的完整精度数值。

6 坐标系与外参格式

6 1 坐标系声明

Manifest.frames 的每项包含 id、frame_type、handedness、axes 四个字符串。基础配置采用右手坐标系，handedness 为 right。

坐标系 **ID**	**frame_type**	**axes** 与方向
head_imu_main_flu	sensor	x_forward_y_left_z_up；前、左、上
head_rgb_front_left_optical	optical	x_right_y_down_z_forward；右、下、前
head_rgb_front_right_optical	optical	x_right_y_down_z_forward；右、下、前

6 2 外参文件结构

camera_camera_extrinsics.json 保存左右相机之间的变换；camera_imu_extrinsics.json 保存 IMU 与左右相机之间的变换。顶层包含 calibration_version、valid_for_source_identity、convention 和 transforms，其中 transforms 是变换对象数组，其余为字符串。

变换对象字段	类型	含义
id	string	变换唯一标识，用于定位与追踪
source_frame	string	输入点所在坐标系
target_frame	string	变换后点所在坐标系
convention	string	T_target_source，方向为源坐标系到目标坐标系
rotation_xyzw	number[4]	旋转四元数 [qx,qy,qz,qw]，应为单位四元数
translation_m	number[3]	平移向量 [tx,ty,tz]，单位 m，表达在目标坐标系
static	boolean	true 表示该标定变换不随帧时间变化

6 3 变换运算

采用列向量。将 rotation_xyzw 按 [qx,qy,qz,qw] 转为 3×3 旋转矩阵 R，将 translation_m 作为列向量 t，组成以下 4×4 齐次变换矩阵。

```
p_target = R(rotation_xyzw) @ p_source + translation_m
T_target_source = [[R11, R12, R13, tx],
                   [R21, R22, R23, ty],
                   [R31, R32, R33, tz],
                   [ 0,   0,   0,  1]]
T_source_target = inverse(T_target_source)
T_A_C = T_A_B @ T_B_C
```

外参共保存 6 条变换：右目到左目、左目到右目、IMU 到左目、左目到 IMU、IMU 到右目、右目到 IMU。按 source_frame 与 target_frame 选择方向；反向变换取逆矩阵，多段变换按上式从右向左连乘。

IMU 原点固定在惯性传感器位置，归一化保留该原点。IMU 与左右相机原点之间的位移由外参 translation_m 表达。三维点转换使用 R 和 t；同一物理向量在不同坐标轴中的分量转换使用 R。

7 时钟定义与时间同步

7 1 时钟对象

clocks 字段	类型	含义
id	string	时钟唯一标识，通过 streams[].clock_domain 引用
unit	string	ns，表示纳秒
epoch	string	device_runtime 表示设备运行期时间基准
synchronization	string	时钟同步状态记录，用于采集配置追溯

左右相机和 IMU 共同使用 head_hardware_clock，三路 timestamp_ns 可直接比较并用于包内时间对齐。epoch 为 device_runtime，表示设备运行时间。计算片段内的相对秒数时，以 metadata.first_timestamp_ns 作为共同起点。

7 2 硬件时间和 PTS

timestamp_ns 用于相机与 IMU 对齐，pts 用于定位视频呈现帧。设 media.time_base 为 [a,b]，origin_ns 为片段共同起点，时间换算如下。

```
t_media_seconds = pts * a / b
t_relative_seconds = (timestamp_ns - origin_ns) / 1e9
rate_hz = (N - 1) * 1e9 / (last_ns - first_ns)
```

N 为记录数。对至少两条时间递增的记录，按上述公式计算平均采样率；expected.nominal_rate_hz 保存标称频率。计算相对时间时先执行整数减法，再将结果转换为浮点秒。

7 3 左右相机配对

左右视频按相同 frame_index 组成双目图像对。两份 timestamps.csv 的记录数一致，对应 frame_index 的 timestamp_ns 一致。读取第 n 对图像时，同时获取左目第 n 帧、右目第 n 帧和索引中的 timestamp_ns。

7 4 IMU 时间对齐

读取图像硬件时间 t0、t1，在 imu.csv 中选取 t0 < timestamp_ns <= t1 的 IMU 采样。用每对相邻 IMU 时间戳之差作为积分步长；图像时间位于两条采样之间时，使用两侧采样做线性插值，得到区间边界值。

三路时间戳按时间递增排列，IMU 的首末时间覆盖视频首末时间。每个积分步长均大于 0。加速度包含重力，计算运动加速度时，先将加速度转换到目标坐标系，再按所用惯性模型完成重力补偿。

7 5 时间偏移解释

additional_time_offset_ns 固定为 0，单位 ns。读取端直接使用 CSV 中的 timestamp_ns 完成左右相机与 IMU 的时间对齐，无需加减额外时间偏移。

8 采集元数据与存储信息

8 1 采集元数据

MetaData/metadata.json 保存数据包的采集上下文。该文件与清单使用一致的租户、数据单元和来源身份，时间范围描述所交付有效流的总体覆盖区间。

字段	类型	含义
source_identity	object	来源身份：kind、namespace、value，均为字符串
tenant_id / unit_id	string	租户与数据单元标识，应与 Manifest 一致
clock_domain	string	首末时间戳使用的时钟域
first_timestamp_ns	string	总体起始时间，十进制纳秒整数字符串
last_timestamp_ns	string	总体结束时间，十进制纳秒整数字符串
datacollector_version	string	采集器版本
firmware_version	string | null	固件版本；null 表示该版本未记录
sdk_version	string | null	SDK 版本；null 表示该版本未记录
device_info	object	设备关联对象
device_info.external_device_id	string	外部设备标识
device_info.platform_device_ref	string | null	平台设备关联引用，尚未关联时为 null

三路流共用硬件时钟。总体起始时间取三路 first_timestamp_ns 的最小值，总体结束时间取三路 last_timestamp_ns 的最大值。片段时长按 (last_timestamp_ns-first_timestamp_ns)/1e9 计算，单位为秒。

8 2 存储定位

Manifest.storage 记录交付对象的存储定位和分配信息；它不改变数据包内部文件的相对路径。

字段	类型	含义
configured_prefix	string	交付存储的配置前缀
normalization_version	string	存储记录中的归一化版本标识，与 profile.version 分开保存
output_allocation_id	string	本次输出的分配标识
storage_ref	string	交付对象对应的存储引用

8 3 身份字段的使用

以 source_identity.value 标识采集会话，以 device_info.external_device_id 关联采集设备，以 tenant_id 和 unit_id 归档数据包。平台设备关联使用 platform_device_ref；该字段为 null 时保留空值，使用 external_device_id 记录设备身份。