

团 体 标 准

T/CAAMTB XX—XXXX

基于城区智慧路口数据的自动驾驶 交通流场景库构建规范

Specifications for constructing traffic flow scenario library for autonomous driving
based on urban smart intersection data

(公开征集意见稿)

在提交反馈意见时，请将您知道的相关专利连同支持性文件一并附上。

XXXX - xx - xx 发布

XXXX - xx - xx 实施

中国汽车工业协会 发布

目 次

前 言	II
1 范围	1
2 规范性引用文件	1
3 术语、定义和缩略语	1
4 要求	2
5 场景库构建	6
6 数据存储与获取	9
附 录 A （资料性） 场景库构建方法	13
参考文献	16

前 言

本文件按照GB/T 1.1—2020《标准化工作导则 第1部分：标准化文件的结构和起草规则》的规定起草。

请注意本文件的某些内容可能涉及专利。本文件的发布机构不承担识别专利的责任。

本文件由中国汽车工业协会智能网联汽车分会提出。

本文件由中国汽车工业协会归口。

本文件起草单位：。

本文件主要起草人：。

基于城区智慧路口数据的自动驾驶交通流场景库构建规范

1 范围

本文件规定了基于城区智慧路口交通流数据构建自动驾驶场景库时应满足的场景采集、设备安装与编号、设备标定与同步、交通参与者识别跟踪、交通参与者轨迹重建、场景库构建和数据存储与获取等要求。

本文件适用于利用城区智慧路口交通流数据构建自动驾驶场景库,满足自动驾驶系统的训练测试需求,路口交通优化、交通流量预测、交通管理等场景应用可参考使用。

2 规范性引用文件

本文件没有规范性引用文件。

3 术语、定义和缩略语

3.1 术语和定义

下列术语和定义适用于本文件。

3.1.1 交通流场景 traffic flow scenario

为了满足基于场景的自动驾驶系统测试需求,结合道路情况,通过对获取的交通参与者轨迹进行处理,交通流场景可划分为制动、变道、切入,转弯、直行、掉头、行人横穿马路等典型场景。

3.1.2 设备标定参数 device parameter calibration

包括外参标定和内参标定。外参标定定义了从不同传感器坐标系到参考坐标系(如图像坐标系)的变换基准。其中,激光雷达到相机的外参标定了从激光雷达坐标系到相机图像坐标系的转换关系。为确保各传感器数据的时空一致性,需进行相机与相机、相机与激光雷达之间的外参标定,所有相机参考坐标系需与对应地图坐标系保持一致;内参标定主要用于校正相机的光学畸变,使采集的图像更加准确。

3.1.3 时间同步 time synchronization

在传感器系统中,确保所有数据采集设备(如摄像头、激光雷达、GPS等)在同一时间基准下记录和传输数据,以实现多源信息的时间一致性。

3.1.4 坐标系变换 coordinate system transformation

将交通参与者的位置信息从图像坐标系转换到世界坐标系的过程。该变换依赖于3.2中定义的激光雷达到图像坐标系的外参,结合相应的几何变换,确保在多传感器环境中,各设备记录的交通参与者位置信息在统一的空间基准下保持一致。

3.1.5 轨迹拼接 trajectory splicing

在多视角传感器环境中,将局部交通参与者轨迹通过轨迹关联方法进行匹配和整合,形成连续的全局轨迹结果。

3.1.6 交通流典型场景 typical scenario of traffic flow

从交通流场景中提取的、具有明确行为模式和共性特征的具体驾驶情景,包括但不限于制动、变道、切入、转弯、直行、掉头、行人横穿马路等。交通流典型场景是场景分类分级的基本单元。

3.1.7 场景分类分级 scenario classification and grading

以交通流典型场景为基本单元,对其按行为模式进行类别划分,并依据交互风险程度(如TTC、PET等指标)对每个典型场景进行危险等级评估,最终形成具有明确类别和危险等级的交通流场景库。

3.2 缩略语

下列缩略语适用于本文件。

API: 应用程序编程接口 (Application Programming Interface)

CARLA: 开源自动驾驶仿真平台 (Car Learning to Act)

CCS: 中国标准文献分类号 (China Classification for Standards)

CSV: 逗号分隔值文件 (Comma-Separated Values)

FOV: 视场角 (Field of View)

GPS: 全球定位系统 (Global Positioning System)

Hz: 赫兹 (Hertz, 频率单位)

ICS: 国际标准分类号 (International Classification for Standards)

ISO: 国际标准化组织 (International Organization for Standardization)

JSON: JavaScript 对象表示法 (JavaScript Object Notation)

MP4: MPEG-4 Part 14 (视频文件格式)

OpenDRIVE: 道路网络高精地图描述格式

OpenSCENARIO: 动态场景描述格式 (对应XOSC文件)

PET: 碰撞后时间差 (Post-Encroachment Time)

TTC: 碰撞时间 (Time To Collision)

TB: 团体标准 (Tuan Biao)

VTD: 虚拟测试驾驶仿真平台 (Virtual Test Drive)

XML: 可扩展标记语言 (Extensible Markup Language)

4 要求

4.1 场景采集要求

为满足自动驾驶系统多样化的测试验证的场景需求,应按以下要求进行场景采集:

- a) 路口类型: 应覆盖十字路口、T型路口、Y型路口及多路不规则路口等;
- b) 交通参与者类型: 应包括轿车、公交车、货车、清扫车、两轮车、三轮车、行人等;
- c) 采集时间: 应覆盖早晨、上午、下午、黄昏、夜晚等不同时段,以对应不同交通流密度;
- d) 采集天气: 应覆盖晴天、阴天、雨天、雾天等不同天气情况;
- e) 除上述要求外,还应提供路口对应的高精地图数据。

4.2 设备安装与编号

4.2.1 设备安装

应对路口设备的安装类型、安装方位进行合理选择与布局:

- a) 设备类型
 - 1) 至少包括相机设备,激光雷达、毫米波雷达可选;

- 2) 相机参数要求帧率至少达到 20Hz，分辨率至少到达 1920x1080 像素；
 - 3) 激光雷达参数要求线束不低于 64，频率不低于 10Hz，探测范围为 200~280m，探测距离精度不高于 3cm；
 - 4) 毫米波雷达参数要求探测距离不低于 200m，角分辨率不高于 1°，相对速度测量误差不高于 0.1m/s。
- b) 安装方位
- 相机的安装方位应确保对目标区域（如路口、路段、人行横道等）实现有效覆盖，满足场景采集与感知需求。具体要求如下：
- 1) 相机的安装方位应确保其视野能够完整覆盖目标区域（如路口、车道、人行横道等）。安装高度、距离、角度及视场角（FOV）参数应根据实际场景需求进行配置，以保证对交通参与者（车辆、行人等）的有效捕获和识别。
 - 2) 激光雷达的安装方位应与相机保持协同，其扫描范围（FOV）应确保能够完整覆盖目标区域，上边缘宜与地面平行，以实现目标区域内障碍物及交通参与者的有效探测。
 - 3) 激光雷达安装方位与相机保持一致，激光雷达 FOV 上边缘与地面平行，保证相机和激光雷达都可以覆盖整个路口；
 - 4) 毫米波雷达的安装方位应与相机和激光雷达保持协同，其探测范围（FOV）应覆盖目标区域的关键部分（如车道、人行横道、交叉口等），确保对运动目标的速度、距离等信息的可靠获取。

4.2.2 设备编号

可利用设备安装路口、方位、号次与类型等信息对设备进行编号：

- a) 路口编号：根据城市区域划分或路口位置规划分配，对每个路口进行唯一编号，例如 A1、B2；
- b) 设备号次：在路口编号的基础上添加每路设备的号码，例如 A1-1、A1-2；
- c) 方位编号：按照设备安装位置（例如：路口的东南西北方向）区分，给每路设备加上 E（东）、S（南）、W（西）、N（北）；
- d) 类型编号：在编号末尾添加设备类型标识，C 表示相机类型，L 表示激光雷达类型，M 表示毫米波雷达类型。

示例：相机编号 A1-1E-C 表示路口 A1 东侧第 1 号相机；激光雷达编号 A1-1E-L 表示路口 A1 东侧第 1 号激光雷达。

4.3 设备标定与同步

4.3.1 设备标定

设备标定涉及相机内参、外参及激光雷达外参的标定：

- a) 相机外参：应采用多相机联合标定方法，实现多相机系统在同一坐标系中的数据对齐，将各个相机的旋转矩阵和位移向量统一到同一坐标系内；
- b) 相机内参：使用标定工具计算内参矩阵，获得相机焦距、主点位置、径向和切向畸变系数；并做去畸变处理，将原始图像进行畸变矫正，以达到更准确的几何信息表达；
- c) 激光雷达外参：应采用与相机外参标定一致的方法，其位姿应保证与相机在同一坐标系内；使用激光雷达-相机联合标定方法确定激光雷达和相机之间的旋转和平移矩阵，实现多传感器数据的空间对齐。

4.3.2 时间同步

传感器数据之间应保证时间同步，以确保各传感器采集的内容能够在同一时间帧内，应包括GPS授时、时间戳记录、同步校验流程：

- a) GPS 授时：在所有传感器上配备支持 GPS 授时的同步模块，使各设备的采集系统能够同步到全球标准时间；
- b) 时间戳记录：每帧数据采集时生成的时间戳应记录至毫秒或微秒级别，时间戳应精确转换为标准时间格式，并附带至每帧数据中，确保在不同传感器采集数据间的时间偏差可控在特定范围内（ $\pm 5\text{ms}$ 以内）；

示例：在每帧数据采集时自动打上时间戳，例如 2024-11-11 14:48:15.123456，并记录到该帧数据的元数据中。

- c) 同步校验：应计算不同传感器的时间偏差，校验不同传感器的时间戳是否一致，必要时对数据进行插值或时间偏移处理。

4.4 交通参与者识别跟踪

4.4.1 算法输入

算法输入要求以10Hz的采样频率从路口各传感器同步获取数据，并对获取的数据进行数据脱敏、数据校验、数据去噪、数据增强、数据矫正、数据切片等数据预处理操作。

4.4.2 算法输出

在预处理后的数据基础上，对交通参与者及交通信号进行检测和跟踪，输出对象和精度应满足以下要求：

- a) 输出对象：包括交通参与者（轿车、公交车、货车、清扫车、两轮车、三轮车、行人等）和红绿灯状态。
- b) 输出精度：
 - 1) 检测框正确率：应不小于 95%，确保识别的边界框准确无误。
 - 2) 类别分类正确率：应不小于 95%，保障目标分类结果的可靠性。
 - 3) 目标跟踪正确率：应不小于 95%，在连续帧中保持目标 ID 的一致性。
 - 4) 状态/属性识别正确率：应不小于 95%，确保对交通标识（如红绿灯状态、箭头方向、交通标志类型等）的识别结果准确可靠。

4.4.3 输出格式

每帧的识别追踪结果应包括交通参与者检测框、目标类别、置信度、红绿灯状态等信息：

- a) 检测框：2D/3D 检测框的坐标信息，用于表示目标的空间位置；
- b) 目标类别：标识交通参与者的类型（轿车、公交车、货车等）；
- c) 置信度：检测和分类的置信度，数值范围一般为 0~1，代表识别结果的可靠性；
- d) 红绿灯状态：当前红绿灯的状态信息（如红灯、绿灯、黄灯）。

示例：轿车：2D 检测框（ x_1, y_1, x_2, y_2 ），类别 car，置信度 0.98；行人：3D 检测框（ x, y, z, w, h, d ），类别 pedestrian，置信度 0.95；红绿灯：状态 red，置信度 0.99。

4.5 交通参与者轨迹重建

4.5.1 坐标系变换

应对各路传感器交通参与者识别跟踪结果进行坐标系变换,将交通参与者位置信息从图像坐标系变换到世界坐标系,确保不同传感器下交通参与者位置的一致性。

4.5.2 全局轨迹拼接

对多视角传感器下的局部交通参与者轨迹进行关联拼接,形成全局轨迹结果,应包括轨迹关联、轨迹平滑、精度要求等内容。

- a) 轨迹拼接结果应保证时间、空间、目标类别的一致性：
 - 1) 时空一致性：通过时间戳和空间位置匹配,确保关联的轨迹属于同一目标,同一时间基准下相邻片段时间间隔 $\leq 100\text{ms}$,拼接点处位置误差需 $\leq 0.5\text{m}$,航向突变需 ≤ 20 ；
 - 2) 目标特征匹配：利用目标的外观特征(如形状、颜色、大小)以及动态特征(如速度、加速度)进行匹配,减少跨视角误识别。
- b) 轨迹拼接结果应保证空间、时间上的平滑连续性：
 - 1) 空间位置平滑：应使用平滑算法,对拼接后的轨迹数据进行去噪和平滑处理,降低误差；
 - 2) 时间序列连续性：对轨迹进行时间序列处理,确保轨迹在时间维度上的连续性,必要时进行插值补充。
- c) 轨迹拼接结果精度应达到以下要求：
 - 1) 目标匹配正确率：不低于 95%,保证同一目标在多视角中的一致识别；
 - 2) 位置轨迹误差：要求 $< 0.5\text{m}$,确保重建的轨迹在实际位置上的准确性。

4.5.3 轨迹信息扩展

全局交通参与者轨迹重建结果除目标位置、类别信息外,还应包括目标每帧的速度、加速度、偏转角、红绿灯状态等信息：

- a) 速度与加速度估计：通过连续帧的位移变化计算交通参与者的速度和加速度；
- b) 偏转角估计：根据目标的运动方向,使用方向向量(通过连续帧的轨迹位置)计算偏转角,确保轨迹符合实际行驶方向；
- c) 红绿灯状态匹配：应将红绿灯状态与位置匹配,将目标靠近的红绿灯状态关联到目标轨迹中。

4.5.4 轨迹数据结构

路口交通流全局轨迹的具体数据结构定义应满足表 1 的要求。

表 1 全局轨迹数据结构定义

元素	是否必填	类型	取值范围	说明
timestamp	是	uint64	-	数据采集时间戳,格式为ISO 8601
frame_id	是	uint64	-	表示数据所属的帧序列,用于精确时间轴上的轨迹还原
track_id	是	uint64	-	唯一的跟踪ID,用于识别同一目标在不同帧中的一致性
object_classes	是	String	{"Truck", "Bus", "Car", "Two-Wheeler", "Pedestrian",	交通参与者的类别(如货车、巴士、小客车、两轮车、行人、清扫车、车头和挂车等)

			"Sweeper","Tractor"和 "Trailer" }	
position	是	Position3D	(x, y, z)	目标的3D空间位置坐标
velocity	是	Vector3D	(vx, vy, vz) ∈ [0, 200] × [0, 200] × [0, 200]	包含速度在各轴方向的分量以及速 度大小的整体值
acceleratio n	是	Vector3D	(ax, ay, az) ∈ [-5, 5] × [-5, 5] × [-5, 5]	包含加速度在各轴方向的分量以及 加速度的整体值
orientation	是	Vector3D	(pitch, yaw, roll) ∈ [-90, 90] × [-180, 180] × [-180, 180]	表示目标运动方向。 Pitch: 描述物体前后倾斜的角度; Yaw :描述物体绕垂直轴旋转的角度; Roll: 描述物体绕其前进方向旋转的 角度
size	是	Vector3D	(l, w, h) ∈ [0, 20] × [0, 5] × [0, 5]	目标物体的三维大小, 包含长度、宽 度和高度, 以米为单位
traffic_ligh t_status	否	string	{ "red", "green", "yellow" }	目标附近的红绿灯状态, 标识其周围 的交通信号信息
lane_id	否	string	-	可选的车道ID, 用于表示目标所属的 车道编号
confidence	否	float16	[0, 1]	整体数据的置信度评分, 用于评估输 出数据的可靠性
vehicle_gr oup_id	否	String	-	用于关联同一辆车的车头和挂车。如 果是普通车, 此字段为空或等于 track_id。

5 场景库构建

5.1 场景内容要求

场景内容要求包括对场景时长、主车信息、场景标签的要求:

- 场景时长: 不低于 10 秒, 频率为 10Hz;
- 主车信息: 应包括起始位置、结束位置及对应的车辆类别;
- 场景标签: 应包括路口类型、采集时段、天气情况、场景类别、危险等级等。

5.2 场景分类分级

应对场景类别进行明确划分, 并定义场景的危险等级:

- 场景类别: 应包括制动、变道、切入, 左转弯、右转弯、直行、掉头、行人横穿马路等典型场景类别;
- 危险等级: 应根据场景交互类型, 利用 TTC、PET 等指标对场景危险等级进行划分。

示例: 低风险: TTC 较大 (例如 >5 秒), PET 较长; 中等风险: TTC 适中 (例如 3-5 秒), PET 稍

短；高风险：TTC 较低（例如<3 秒），PET 较短。

5.3 场景分布统计

应从场景类别和危险等级两个维度对场景相关分布统计,以满足自动驾驶算法训练验证的数据定制采样需求。

- a) 类别分布：应统计每类场景的数量分布，分析各类场景的发生频率；
- b) 危险等级分布：应分别统计每类场景下不同危险等级的数量分布，分析每类场景的风险水平。

5.4 场景数据结构

路口交通流场景的具体数据结构定义应满足表 2 的要求。

表 2 场景数据结构定义

主元素	是否必填	类型	取值范围	说明
scene_id	是	string	-	场景ID, 用于唯一标识每个场景
scene_category	是	string	{"Emergency Braking", "Cut-in", "Lane Change", "Left Turning", "Right Turning", "Straight Driving", "U-turn", "Pedestrian Crossing", "Overtaking", "Lane Departure"}	场景类别,有: 制动、切入、变道、左转、右转、直行、掉头、行人横穿马路、超车、偏离车道等
timestamp	是	uint64	-	场景开始时间
duration	是	uint64	{10, 20, 30}	场景时长, 单位秒
location	是	JSON	{"intersection_type": string, "lane_id": string}	场景所在位置, 包含交叉口类型和车道ID
weather	是	string	{"clear", "overcast", "rain", "snow"}	天气情况,有晴天、阴天、雨天、雪天
time_of_day	是	string	{"AM", "Noon", "PM"}	数据的采集时段
main_vehicle	是	JSON	{"track_id": uint64, "object_class": string, "size": Vector3D, "start_frame_id": uint64, "end_frame_id": uint64, "trajectory": {"position": Vector[Position3D], "velocity": Vector4D, "acceleration": Vector4D, "orientation": Vector4D, "frame_id": vector[uint64],	主车信息, 主车的 track_id、类别、尺寸、起始与终止的 frame_id、轨迹等

			"traffic_light_status": Vector[string], "lane_id": Vector[string]}}	
background_objects_trajectory	是	Vector[JSON]	{ "track_id": uint64, "object_class": string, "size": Vector3D, "start_frame_id": uint64 , "end_frame_id": uint64, "trajectory": { "position": Vector[Position3D], "velocity": Vector4D, "acceleration": Vector4D, "orientation": Vector4D, "frame_id": vector[uint64], "traffic_light_status": Vector[string], "lane_id": Vector[string]}}	交通参与者的轨迹信息, 有track_id,类别, 起始与终止帧号, 坐标, 速度, 加速度等信息
risk_level	是	float16	[0, 1]	场景的整体风险等级
map_name	是	string	-	场景所依赖的地图信息
risk_metrics	是	JSON	{ "TTC": float32, "PET": float32}	场景的危险度指标, 包括TTC和PET
frequency_distribution	是	JSON	{ "scene_category": string, "distribution": vector[float]}	场景类别和危险等级的分布统计

5.5 场景库质量要求

场景库应满足以下质量要求：

- 场景库的多样性要求：应包含 5.2 节所述的所有场景类型，每种场景类型至少应包含 10–20 个不同的场景实例，以模拟不同的交通情况；
- 场景库的危险性要求：根据 5.3 节统计结果，应包括不同危险等级的场景，确保高风险场景的比例适当，高风险、中风险、低风险场景数量的比例可设为 2:3:5；
- 场景库的数量要求：应至少包含 100 个场景实例，确保对不同交通场景的全面测试。

5.6 场景库仿真要求

为了在主流仿真器（如 CARLA、VTD 等）执行自动驾驶场景测试，场景库在高精地图导入和场景文件格式转换方面需要满足以下仿真要求：

- 高精地图导入：应确保高精地图数据的格式与所选择的仿真器兼容。主流仿真器通常支持多种格式（如 OpenDrive、OSM 等），需要将地图数据转换为适合仿真器的格式；高精地图应包含详细的道路网络信息，包括车道线、交通标志、信号灯、路边障碍物等，以便准确模拟真实场景；

- b) 场景文件格式转换：应将本标准定义的轨迹文件格式转换为仿真器所需的场景文件格式（如 XOSC、XML、JSON 等），以导入仿真平台进行动态交通参与者的轨迹还原；
- c) 参数配置：在仿真平台中配置相关环境参数，如天气条件、采集时间、路面状况等，以匹配实际场景。

6 数据存储与获取

6.1 数据存储

应以路口为单位，将不同传感器和轨迹场景数据以文件形式进行有序存储，具体存储结构如下：

```

/IntersectionData/
├── Intersection_A1/ # 路口编号 A1
│   ├── 2024-11-11_10_00/ # 时间段：2024年11月11日10:00
│   │   ├── CameraVideos/ # 路口摄像头视频数据
│   │   │   ├── A1-1E-C.mp4 # 东侧第1个相机的视频数据
│   │   │   ├── A1-2S-C.mp4 # 南侧第2个相机的视频数据
│   │   │   └── ... # 其他相机的视频数据
│   │   ├── LidarPointClouds/ # 激光雷达点云数据
│   │   │   ├── A1-1E-L.pcd # 东侧第1个激光雷达点云数据
│   │   │   ├── A1-2S-L.pcd # 南侧第2个激光雷达点云数据
│   │   │   └── ... # 其他激光雷达点云数据
│   │   ├── HDMap/ # 高精地图文件
│   │   │   └── hd_map.xodr # 路口的高精地图文件（OpenDRIVE格式）
│   │   ├── ObjectTracking/ # 目标识别追踪数据
│   │   │   ├── A1-1E-C_tracking.json # 东侧第1个相机的目标追踪数据
│   │   │   ├── A1-2S-C_tracking.json # 南侧第2个相机的目标追踪数据
│   │   │   └── ... # 其他相机的目标追踪数据
│   │   ├── GlobalTrafficTrajectory/ # 全局交通流轨迹
│   │   │   └── traffic_trajectory.json # 当前时间段的全局交通流数据
│   │   ├── SceneFiles/ # 交通流场景文件
│   │   │   ├── braking.xml # 制动场景描述文件
│   │   │   ├── lane_change.xml # 变道场景描述文件
│   │   │   └── ... # 其他场景文件
│   │   └── Metadata/ # 数据元信息
│   │       └── metadata.json # 当前路口及时间段的元信息
│   └── ...
└── Intersection_B2/ # 路口编号 B2
    └── ...
  
```

6.2 数据分页机制

6.2.1 分页机制设计原则

为了支持离线数据集的灵活获取，应设计API调用接口，支持按需获取不同路口和时间段的传感器数据、轨迹数据、场景数据、场景元数据的功能。

所有返回数据列表的查询接口均应支持分页功能，以避免单次请求数据量过大导致系统资源耗尽或请求超时。分页参数通过请求的查询字符串传递，响应中除数据列表外应携带分页元数据。

6.2.2 请求参数

表 3 请求参数说明

参数名称	类型	必选	说明
pageNo	整型	否	当前页码，从 1 开始计。默认值为 1。
pageSize	整型	否	每页返回的最大记录数。默认值为 20，最大不应超过 200。若请求值超过最大值，服务端应按最大值处理或返回参数错误。
sortBy	字符串	否	排序字段名称。接口文档应明确定义可用的排序字段，默认按数据产生时间降序。
order	字符串	否	排序方向，取值为 asc（升序）或 desc（降序），默认值为 desc。

6.2.3 响应元数据

服务端返回的分页响应应包含以下元数据字段，统一放置于响应体的 data 或 pagination 对象中（具体位置由各接口约定，但所有接口应保持一致）：

- a) total：整型，符合查询条件的总记录数；
- b) pageNo：整型，当前页码；
- c) pageSize：整型，实际生效的每页记录数；
- d) pages：整型，总页数，计算方式为 $\text{ceil}(\text{total} / \text{pageSize})$ 。

6.2.4 游标分页

对于数据动态增长、排序不稳定（如实时视频流、事件流）的查询场景，推荐采用游标分页替代页码分页。游标分页应遵循以下要求：

- a) 请求参数使用 cursor（字符串，服务端上一次响应中返回）和 limit（整型，类似 pageSize，默认值及最大值与 pageSize 一致）；
- b) 首次请求时不携带 cursor，服务端应从第一页开始返回；
- c) 响应中不应返回 total 与 pages，而应返回；
- d) nextCursor：字符串，用于获取下一页数据的游标。若已无更多数据，该字段值为 null；
- e) hasMore：布尔值，表示是否还有更多数据；
- f) 游标的具体内容由服务端定义（例如可基于时间戳与唯一标识的组合生成），客户端不应解析、篡改或自行构造游标值。

6.3 数据获取

6.3.1 获取路口与时间段元信息接口

- a) 接口名称：GET /api/intersection/{intersection_id}/{timestamp}/metadata;
- b) 描述：获取指定路口和时间段的元信息（时间、设备信息等）；
- c) 请求参数：
 - 1) intersection_id（必填）：路口编号；
 - 2) timestamp（必填）：时间段起止时间；
- d) 返回数据：JSON 格式的元信息。

6.3.2 获取视频数据接口

- a) 接口名称：GET/api/intersection/{intersection_id}/{timestamp}/camera/{camera_id}/video;
- b) 描述：获取指定时间段的特定相机视频数据；
- c) 请求参数：
 - 1) intersection_id（必填）：路口编号；
 - 2) timestamp（必填）：时间段起止时间；
 - 3) camera_id（必填）：相机编号（如 A1-1E-C）；
- d) 返回数据：视频文件流（MP4）。

6.3.3 获取目标追踪数据接口

- a) 接口名称：GET/api/intersection/{intersection_id}/{timestamp}/object_tracking/{camera_id};
- b) 描述：获取指定时间段的相机目标追踪数据；
- c) 请求参数：
 - 1) intersection_id（必填）：路口编号；
 - 2) timestamp（必填）：时间段起止时间；
 - 3) camera_id（必填）：相机编号（如 A1-1E-C）；
- d) 返回数据：JSON 格式的追踪数据。

6.3.4 获取交通流轨迹数据接口

- a) 接口名称：GET /api/intersection/{intersection_id}/{timestamp}/traffic_trajectory;
- b) 描述：获取指定时间段内的全局交通流轨迹数据；
- c) 请求参数：
 - 1) intersection_id（必填）：路口编号；
 - 2) timestamp（必填）：时间段起止时间；
- d) 返回数据：JSON 或 CSV 格式的交通流轨迹。

6.3.5 获取交通场景描述文件接口

- a) 接口名称：GET/api/intersection/{intersection_id}/{timestamp}/scene_file/{scene_type};
- b) 描述：获取指定时间段的特定类型交通场景文件；
- c) 请求参数：
 - 1) intersection_id（必填）：路口编号；
 - 2) timestamp（必填）：时间段起止时间；

- 3) scene_type (可选)：场景文件类型 (如 braking、lane_change 等)；
- d) 返回数据：XML 格式的场景描述文件。

附录 A (资料性) 场景库构建方法

本附录给出了交通流场景库构建的方法示例，供使用本文件时参考。

A.1 场景库构建流程

整体流程如图A.1所示：

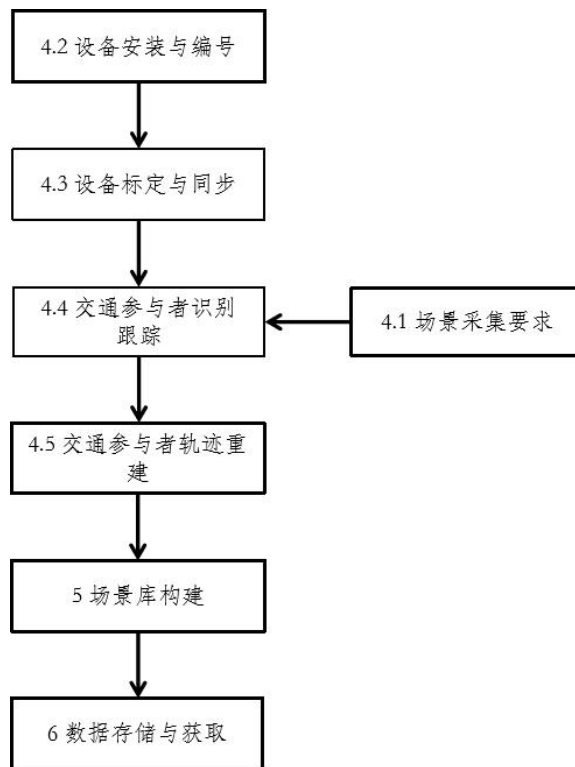


图 A.1 场景库构建流程

A.2 方法

A.2.1 预处理

通过软件界面或API将路口各传感器设备采集的视频数据传输至路口交通流场景数据合成平台，完成数据脱敏、数据校验、数据去噪、数据增强、数据矫正、数据切片等数据预处理操作。

A.2.2 交通参与者识别跟踪

在预处理后的数据基础上，对交通参与者及交通信号灯进行检测识别和跟踪。以十字路口为例，各个相机的识别跟踪结果如图A.2所示。



图 A.2 路口各相机交通参与者识别跟踪结果

A.2.3 交通参与者轨迹重建

交通参与者轨迹重建首先对各路相机的交通参与者识别跟踪结果进行局部轨迹重建,如图A.3所示;然后对各路相机的局部重建轨迹进行融合,形成全局化的、时空一致的交通参与者轨迹,路口交通参与者全局轨迹重建结果如图A.4所示。

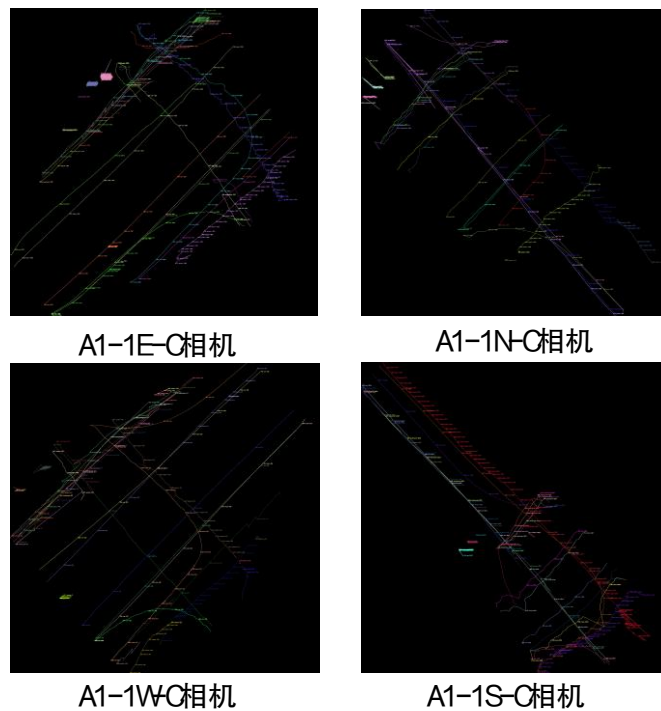


图 A.3 路口交通参与者局部轨迹重建结果

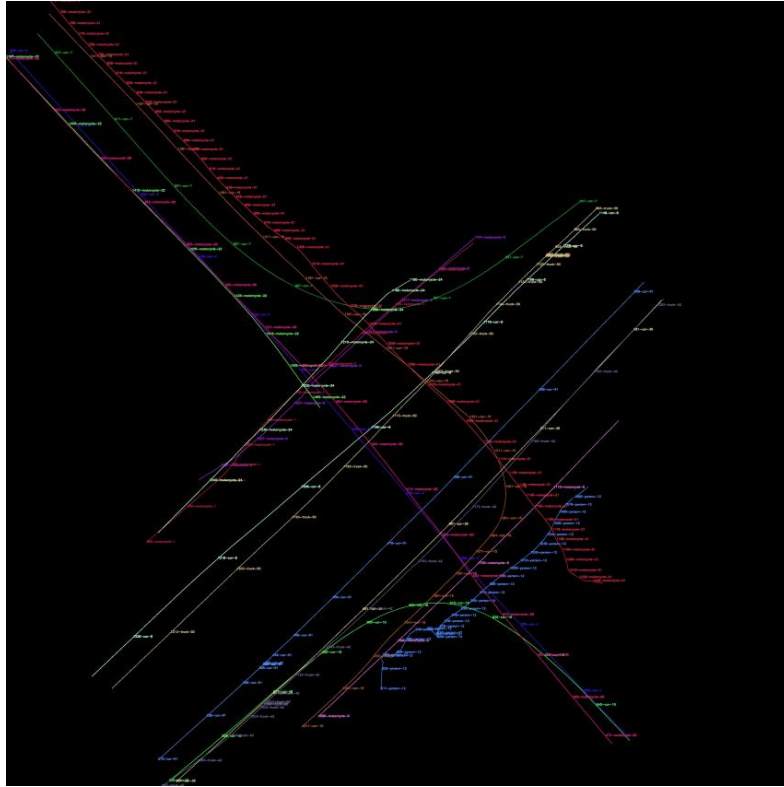


图 A.4 路口交通参与者全局轨迹重建结果

A.2.4 场景库构建

基于重建的全局交通参与者轨迹，对变道、左转弯、直行、行人横穿马路等典型场景进行提取，构建自动驾驶测试场景库，场景库典型场景示例及对应的危险等级如图A.5所示。

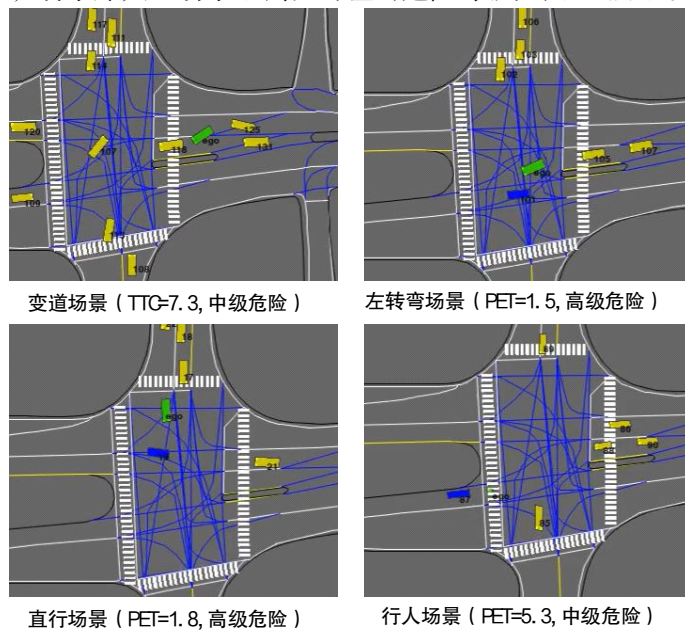


图 A.5 场景库典型场景示例及对应的危险等级

参考文献

- [1] GB/T 44373-2024 智能网联汽车 术语和定义
- [2] GB/T 40429-2021 汽车驾驶自动化分级
- [3] T/CITSA 15-2021 智能交通摄像机安全技术要求
- [4] T/CAA 002-2022 交通参与者行为理解与轨迹预测的评测方法及数据集构建标准
- [5] T/ZSA 40-2020 自动驾驶仿真测试场景集要求
- [6] ISO 34501-2022 道路车辆 自动驾驶系统测试场景 词汇
- [7] ISO 34503-2023 道路车辆 自动驾驶系统试验场景 运行设计域规范
- [8] ISO 21448-2022 道路车辆 预期功能安全