

SVP API 开发参考

文档版本

V2.3

发布日期

2025-12-15

前言

概述

本文档用于指导开发人员进行智能视频处理 (Smart Video Process)，为开发过程中查阅 SVP 支持的各种参考信息，包括 API、头文件。



说明

本文档未经特殊说明，以 XM7206V11A 指代 XM7206 系列所有芯片。

产品版本

与本文档相对应的产品版本如下。

产品名称	产品版本

读者对象

本文档（本指南）主要适用于以下工程师：

- 技术支持工程师
- 软件开发工程师

符号约定

在本文中可能出现下列标志，它们所代表的含义如下。

符号	说明
 危险	表示如不可避免则将会导致死亡或严重伤害的具有高等级风险的危害。
 警告	表示如不可避免则可能导致死亡或严重伤害的具有中等级风险的危害。
 注意	表示如不可避免则可能导致轻微或中度伤害的具有低等级风险的危害。
须知	用于传递设备或环境安全警示信息。如不可避免则可能会导致设备损坏、数据丢失、设备性能降低或其它不可预知的结果。 “须知”不涉及人身伤害。
 说明	对正文中重点信息的补充说明。 “说明”不是安全警示信息，不涉及人身、设备及环境伤害信息。

修订记录

修订记录累积了每次文档更新的说明。最新版本的文档包含以前所有文档版本的更新内容。

修订日期	版本	修订说明
2025-06-05	V2.1	首次发布，DI 版本。
2025-07-28	V2.2	更新 AOV 相关属性，修改 edge_limit 相关描述。
2025-12-15	V2.3	新增二阶段检测、烟火检测、DMS、行车类相关功能及描述。

目 录

1 功能介绍	1
1.1 目标检测类	1
1.1.1 人形检测.....	1
1.1.2 人脸检测.....	1
1.1.3 车形检测.....	1
1.1.4 非机动车检测	1
1.1.5 宠物检测.....	2
1.1.6 人体关键点检测.....	2
1.1.7 头部检测.....	2
1.1.8 二阶段检测	2
1.1.9 二阶段远距离检测	2
1.1.10 烟火检测.....	2
1.1.11 陪伴机器人.....	3
1.2 识别类.....	3
1.2.1 人脸识别.....	3
1.2.2 人脸表情识别	3
1.2.3 ADAS 高级驾驶辅助.....	3
1.2.4 手势识别.....	4
1.2.5 DMS 驾驶员监测系统.....	4
1.2.6 车牌识别.....	4

1.2.7 车辆识别.....	4
1.2.8 车牌车辆识别	4
2 SVP API 参考.....	5
2.1 API 参考.....	5
2.2 数据类型.....	11
2.2.1 数据类型定义	11
2.2.2 宏定义	14
2.2.3 枚举类型.....	19
2.2.4 结构体	31

1 功能介绍

1.1 目标检测类

1.1.1 人形检测

输入：640x360 YUV420SP；

输出：人形的目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (人形)，追踪 ID 等。

1.1.2 人脸检测

输入：640x360 YUV420SP；

输出：人脸的目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (人脸)，追踪 ID 等。

1.1.3 车形检测

输入：640 x 360 YUV420SP；

输出：机动车的目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (机动车)，追踪 ID 等。

1.1.4 非机动车检测

输入：640 x 360 YUV420SP；

输出：非机动车的目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (自行车 or 电动车 or 三轮车 or 骑自行车的人 or 骑电动车的人 or 骑三轮车的人)，追踪 ID 等。

1.1.5 宠物检测

输入：640 x 360 YUV420SP;

输出：宠物的目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 类别标签 (宠物), 追踪 ID 等。

1.1.6 人体关键点检测

输入：640 x 360 YUV420SP;

输出：人形的目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 类别标签 (人形) 人形的 17 个关键点和得分 (x, y, score), 追踪 ID 等。

1.1.7 头部检测

输入：640 x 360 YUV420SP;

输出：头部的目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 类别标签 (人头), 追踪 ID 等。

1.1.8 二阶段检测

输入：640 x 360 YUV420SP;

输出：人/车/非/宠的目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 类别标签 (人形/车形/非机动车/宠物), 追踪 ID 等。

1.1.9 二阶段远距离检测

输入：1280 x 720 YUV420SP;

输出：人/车/非/宠的目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 类别标签 (人形/车形/非机动车/宠物), 追踪 ID 等。

1.1.10 烟火检测

输入：640 x 360 YUV420SP;

输出：火焰/烟雾目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 类别标签 (火焰/烟雾) 等。

1.1.11 陪伴机器人

输入：640 x 360 YUV420SP；

输出：宠物目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (宠物)，目标距离 (需要转换系数转换成真实距离) 等。

1.2 识别类

1.2.1 人脸识别

输入：640 x 360 YUV420SP, 1920 x 1080 YUV420SP；

输出：人脸目标框 (x1, y1, x2, y2)，256 维的浮点特征向量等；

识别：根据 query 人脸特征向量和数据库中的特征向量进行余弦距离匹配，距离小于阈值，识别成功，输出对应人脸 ID，否则识别失败。

1.2.2 人脸表情识别

输入：640 x 360 YUV420SP, 1920 x 1080 YUV420SP；

输出：人脸目标框 (x1, y1, x2, y2)，人脸表情 (微笑、正常、未知)，人脸质量 (0 到 1 浮点)，人脸关键点 (5 个)，追踪 ID 和 AGE 等。

1.2.3 ADAS 高级驾驶辅助

输入：640 x 360 YUV420SP, 1920 x 1080 YUV420SP；

输出：

人形目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (人形) 等；

机动车目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (机动车) 等；

非机动车目标框 (x1, y1, x2, y2)，置信度 (0 到 1 浮点)，类别标签 (自行车/电动车/三轮车/骑自行车的人/骑电动车的人/骑三轮车的人) 等；

车牌目标框 (x1, y1, x2, y2)；车道线原始关键点数量及坐标等；

车道线拟合后关键点数量及坐标，车道线拟合系数等。

1.2.4 手势识别

输入：640 x 360 YUV420SP, 1920 x 1080 YUV420SP;

输出：手的目标框 (x1, y1, x2, y2), 手势的类别, 置信度 (0 到 1 浮点) 等。

1.2.5 DMS 驾驶员监测系统

输入：640 x 360 YUV420SP;

输出：

人脸目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 疲劳状态等;

手机目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 玩手机状态等;

香烟目标框 (x1, y1, x2, y2), 置信度 (0 到 1 浮点), 抽烟状态等;

持续追踪目标 (面积最大的人脸), 人脸 68 个关键点 (外部计算俯仰角、偏航角、翻滚角判断分心状态) 等。

1.2.6 车牌识别

输入：1920 x 1080 YUV420SP;

输出：车牌目标框 (x1, y1, x2, y2), 追踪 ID, 车牌字符, 车牌字符置信度 (0 到 1 浮点), 车牌颜色类别, 车牌颜色置信度 (0 到 1 浮点), 车牌触发方向等。

1.2.7 车辆识别

输入：1920 x 1080 YUV420SP;

输出：车辆目标框 (x1, y1, x2, y2), 追踪 ID, 车辆类型, 车辆类型置信度 (0 到 1 浮点), 车辆颜色类别, 车辆颜色置信度 (0 到 1 浮点), 车辆触发方向等。

1.2.8 车牌车辆识别

输入：1920 x 1080 YUV420SP;

输出：车牌目标框 (x1, y1, x2, y2), 车辆目标框 (x1, y1, x2, y2), 追踪 ID, 车牌字符, 车牌字符置信度 (0 到 1 浮点), 车牌颜色类别, 车牌颜色置信度 (0 到 1 浮点), 车辆类型, 车辆类型置信度 (0 到 1 浮点), 车辆颜色类别, 车辆颜色置信度 (0 到 1 浮点), 车牌触发方向, 车辆触发方向等。

2 SVP API 参考

2.1 API 参考

该功能模块为用户提供以下 API:

- [xmedia_svp_init](#): 初始化 SVP。
- [xmedia_svp_uninit](#): 去初始化 SVP。
- [xmedia_svp_set_config](#): 设置 svp 配置信息。
- [xmedia_svp_get_config](#): 获取 svp 配置信息。
- [xmedia_svp_task_create](#): 创建一路 svp task 处理流程。
- [xmedia_svp_task_destroy](#): 销毁一路 svp task 处理流程。
- [xmedia_svp_task_set_attr](#): 设置 svp task 处理流程的属性。
- [xmedia_svp_task_get_attr](#): 获取 svp task 处理流程的属性。
- [xmedia_svp_task_process](#): 开始 svp task 处理流程。

xmedia_svp_init

【描述】

初始化 SVP。

【语法】

```
xmedia_s32 xmedia_svp_init(xmedia_void);
```

【参数】

无。

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

xmedia_svp_uninit

【描述】

去初始化 SVP。

【语法】

```
xmedia_s32 xmedia_svp_uninit(xmedia_void);
```

【参数】

无。

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

xmedia_svp_set_config

【描述】

设置 svp 配置信息。

【语法】

```
xmedia_s32 xmedia_svp_set_config(xmedia_svp_cfg* svp_cfg);
```

【参数】

参数名称	描述	输入/输出
svp_cfg	svp 配置信息。	输入

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

xmedia_svp_get_config

【描述】

获取 svp 配置信息。

【语法】

```
xmedia_s32 xmedia_svp_get_config(xmedia_svp_cfg* svp_cfg);
```

【参数】

参数名称	描述	输入/输出
svp_cfg	svp 配置信息。	输入

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

xmedia_svp_task_create

【描述】

创建一路 svp task 处理流程。

【语法】

```
xmedia_s32 xmedia_svp_task_create(xmedia_s32 *handle, const xmedia_svp_task_cfg cfg);
```

【参数】

参数名称	描述	输入/输出
handle	创建成功返回 SVP 句柄。	输出
cfg	svp_task 配置信息。	输入

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

【备注】

每个句柄对应一路 svp task 处理流程。

xmedia_svp_task_destroy

【描述】

销毁一路 svp task 处理流程。

【语法】

```
xmedia_s32 xmedia_svp_task_destroy(xmedia_s32 handle);
```

【参数】

参数名称	描述	输入/输出
handle	SVP 处理流程句柄。	输入

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

xmedia_svp_task_set_attr

【描述】

设置 svp task 处理流程的属性。

【语法】

```
xmedia_s32 xmedia_svp_task_set_attr(xmedia_s32 handle, const xmedia_void *task_attr);
```

【参数】

参数名称	描述	输入/输出
handle	svp task 处理流程句柄。	输入
task_attr	设置某一类型 task 的属性。需保证与 handle 对应 task 类型一致。	输入

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

【注意事项】

1. task_attr 参数为 void 指针类型，不同 task 类型对应不同的属性，需要在输入时做对应类型的强转。
2. 创建后，销毁前的任意时间均可调用。

xmedia_svp_task_get_attr

【描述】

获取 svp task 处理流程的属性。

【语法】

```
xmedia_s32 xmedia_svp_task_get_attr(xmedia_s32 handle, xmedia_void *task_attr);
```

【参数】

参数名称	描述	输入/输出
handle	svp_task 处理流程句柄。	输入
task_attr	获取某一类型 task 的属性。需保证与 handle 对应 task 类型一致。	输出

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。

返回值	描述
XMEDIA_FAILURE	函数执行失败。

【注意事项】

1. task_attr 参数为 void 指针类型，不同 task 类型对应不同的属性，需要在输入时做对应类型的强转。
2. 创建后，销毁前的任意时间均可调用。

xmedia_svp_task_process

【描述】

开始 svp task 处理流程。

【语法】

```
xmedia_s32 xmedia_svp_task_process(xmedia_s32 handle, const xmedia_svp_task_input *input, xmedia_void *output);
```

【参数】

参数名称	描述	输入/输出
handle	svp 处理流程句柄。	输入
input	输入图像帧信息。	输入
output	输出某一类型 task 的结果。需保证与 handle 对应 task 类型一致。	输出

【返回值】

返回值	描述
XMEDIA_SUCCESS	函数执行成功。
XMEDIA_FAILURE	函数执行失败。

【注意事项】

1. output 参数为 void 指针类型，不同 task 类型对应不同的输出数据类型，需要做对应类型的强转。
2. 创建后，销毁前的任意时间均可调用。

2.2 数据类型

2.2.1 数据类型定义

SVP 模块相关数据类型定义如下：

- `XMEDIA_SVP_MAX_TASK_NUM`：定义 SVP 最大任务个数。
- `XMEDIA_SVP_MAX_TARGET_NUM`：定义 SVP 单帧最多输出的目标个数。
- `MAX_DETECT_KEYPOINT_NUM`：定义 SVP yolov8 关键点数量。
- `XMEDIA_SVP_DMS_MAX_TARGET_NUM`：定义 DMS 单帧输出目标最大个数。
- `XMEDIA_SVP_DMS_SPECIAL_MAX_NUM`：定义 DMS 手机和香烟最多输出的目标个数。
- `XMEDIA_SVP_LANE_MAX_NUM`：定义 SVP 最大车道线数量。
- `XMEDIA_SVP_LANE_AXIS_NUM`：定义 SVP 车道线方向数量（行、列）。
- `XMEDIA_SVP_LANE_MAX_ROW_POINT`：定义 SVP 车道线行方向关键点个数。
- `XMEDIA_SVP_LANE_MAX_COL_POINT`：定义 SVP 车道线列方向关键点个数。
- `XMEDIA_SVP_LANE_FIT_ORDER`：定义 SVP 车道线拟合因子数量。
- `XMEDIA_SVP_FEATURE_COL`：定义 SVP 人脸特征向量纬度。
- `XMEDIA_SVP_FR_KEYPOINT_NUM`：定义 SVP 人脸关键点数量。
- `XMEDIA_SVP_MAX_VENC_CHN_NUM`：定义 SVP venc 最大通道数。
- `XMEDIA_SVP_MAX_VI_PIPE_NUM`：定义 SVP vi 最大通道数。
- `XMEDIA_SVP_PLATE_CHAR_MAX_NUM`：定义车牌字符最大数量。
- `XMEDIA_SVP_MAX_REGION`：定义最大区域数量。
- `XMEDIA_SVP_MAX_EDGE`：定义区域最大边数。
- `XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM`：车牌车辆单帧输出目标最大个数。

- `xmedia_svp_alg_type`: 定义设置算法类别。
- `xmedia_svp_class_type`: 定义输出某个目标的类别。
- `xmedia_svp_inputdata_format`: 定义输入数据格式。
- `xmedia_svp_model_type`: 定义模型输入类型。
- `xmedia_svp_motion_state`: 定义目标运动状态类型。
- `xmedia_svp_task_type`: 定义 SVP 任务类型。
- `xmedia_svp_class_emotion_type`: 定义表情类型。
- `xmedia_svp_mem_reuse_type`: 定义内存复用类型。
- `xmedia_svp_direction`: 定义车牌车辆方向。
- `xmedia_svp_plate_color`: 定义车牌颜色。
- `xmedia_svp_vehicle_type`: 定义车辆类型。
- `xmedia_svp_vehicle_color`: 定义车辆颜色。
- `xmedia_svp_rect`: 定义矩形坐标结构体。
- `xmedia_svp_detect_result`: 定义目标检测结果信息结构体。
- `xmedia_svp_point`: 定义点结构体。
- `xmedia_svp_area_limit_attr`: 定义面积限制属性结构体。
- `xmedia_svp_edge_limit_attr`: 定义边缘过滤属性结构体。
- `xmedia_svp_task_input`: 定义 SVP 任务输入结构体。
- `xmedia_svp_modules`: 定义模型输入信息结构体。
- `xmedia_svp_task_cfg`: 定义任务配置信息结构体。
- `xmedia_svp_detect_base_attr`: 定义基础检测属性结构体。
- `xmedia_svp_deepsort_attr`: 定义 deepsort 算法属性结构体。
- `xmedia_svp_uav_attr`: 定义无人机任务属性结构体。
- `xmedia_svp_uav_output`: 定义无人机任务检测输出结构体。
- `xmedia_svp_yolov5_attr`: 定义 yolov5 检测属性结构体。
- `xmedia_svp_yolov5_output`: 定义 yolov5 检测输出结构体。

- `xmedia_svp_aov_attr`: 定义 aov 检测属性结构体。
- `xmedia_svp_aov_detect_output`: 定义 aov 检测输出结构体。
- `xmedia_svp_keypoint`: 定义关键点结构体。
- `xmedia_svp_yolov8_attr`: 定义 yolov8 检测属性结构体。
- `xmedia_svp_yolov8_result`: 定义 yolov8 检测结果信息结构体。
- `xmedia_svp_yolov8_output`: 定义 yolov8 检测结果输入结构体。
- `xmedia_svp_adas_distance_attr`: 定义 adas 任务距离属性结构体。
- `xmedia_svp_adas_attr`: 定义 adas 任务属性结构体。
- `xmedia_svp_adas_result`: 定义 adas 任务结果输出结构体。
- `xmedia_svp_fire_attr`: 定义烟火检测 (yolov8) 属性结构体。
- `xmedia_svp_gesture_result`: 定义手势识别结果输出结构体。
- `xmedia_svp_face_pose_attr`: 定义人脸姿态属性结构体。
- `xmedia_svp_fr_attr`: 定义人脸识别属性结构体。
- `xmedia_svp_fr_result`: 定义人脸识别结果信息结构体。
- `xmedia_svp_fr_output`: 定义人脸识别结果输出结构体。
- `xmedia_svp_face_attribute_result`: 定义人脸属性任务结果信息结构体。
- `xmedia_svp_face_attribute_output`: 定义人脸属性任务结果输出结构体。
- `xmedia_svp_companion_robot_output`: 定义陪伴机器人任务结果输出结构体。
- `xmedia_svp_mem_info`: 定义内存信息结构体。
- `xmedia_svp_cfg`: 定义模型内存共用结构体。
- `xmedia_svp_multi_base_attr`: 定义多类别检测任务基本属性结构体。
- `xmedia_svp_multi_detect_attr`: 定义多类别检测任务属性结构体。
- `xmedia_svp_dms_attr`: 定义 DMS 检测任务属性结构体。
- `xmedia_svp_dms_state`: 定义 DMS 状态属性结构体。
- `xmedia_svp_dms_report`: 定义 DMS 状态结果上报结构体。
- `xmedia_svp_dms_result`: 定义 DMS 结果结构体。

- `xmedia_svp_dms_output`: 定义 DMS 任务输出信息。
- `xmedia_svp_vec2i`: 定义万分比坐标点结构体。
- `xmedia_svp_landmarks`: 定义坐标点集合结构体。
- `xmedia_svp_zone`: 定义区域信息结构体。
- `xmedia_svp_plate_reco_attr`: 定义车牌识别任务属性结构体。
- `xmedia_svp_plate_result`: 定义车牌识别任务结果结构体。
- `xmedia_svp_plate_reco_output`: 定义车牌识别任务输出结构体。
- `xmedia_svp_vehicle_reco_attr`: 定义车辆识别属性结构体。
- `xmedia_svp_vehicle_result`: 定义车辆识别任务结果结构体。
- `xmedia_svp_vehicle_reco_output`: 定义车辆识别任务输出结构体。
- `xmedia_svp_plate_vehicle_attr`: 定义车牌车辆识别属性结构体。
- `xmedia_svp_plate_veh_result`: 定义车牌车辆识别任务结果结构体。
- `xmedia_svp_plate_vehicle_output`: 定义车牌车辆识别任务输出结构体。

2.2.2 宏定义

XMEDIA_SVP_MAX_TASK_NUM

【说明】

最大 task 任务数量。

【定义】

```
#define XMEDIA_SVP_MAX_TASK_NUM (16u)
```

【相关数据类型及接口】

无

XMEDIA_SVP_MAX_TARGET_NUM

【说明】

单帧最多输出的目标个数。

【定义】

```
#define XMEDIA_SVP_MAX_TARGET_NUM (50u)
```

【相关数据类型及接口】

无

MAX_DETECT_KEYPOINT_NUM

【说明】

yolov8 关键点数量。

【定义】

```
#define MAX_DETECT_KEYPOINT_NUM (68u)
```

【相关数据类型及接口】

无

XMEDIA_SVP_DMS_MAX_TARGET_NUM

【说明】

DMS 单帧输出目标最大个数。

【定义】

```
#define XMEDIA_SVP_DMS_MAX_TARGET_NUM (5u)
```

【相关数据类型及接口】

无

XMEDIA_SVP_DMS_SPECIAL_MAX_NUM

【说明】

DMS 手机和香烟最多输出的目标个数。

【定义】

```
#define XMEDIA_SVP_DMS_SPECIAL_MAX_NUM (3u)
```

【相关数据类型及接口】

无

XMEDIA_SVP_LANE_MAX_NUM

【说明】

最大车道线数量。

【定义】

```
#define XMEDIA_SVP_LANE_MAX_NUM 4
```

【相关数据类型及接口】

无

XMEDIA_SVP_LANE_AXIS_NUM

【说明】

车道线方向数量（行、列）。

【定义】

```
#define XMEDIA_SVP_LANE_AXIS_NUM 2
```

【相关数据类型及接口】

无

XMEDIA_SVP_LANE_MAX_ROW_POINT

【说明】

车道线行方向关键点数量。

【定义】

```
#define XMEDIA_SVP_LANE_MAX_ROW_POINT 18
```

【相关数据类型及接口】

无

XMEDIA_SVP_LANE_MAX_COL_POINT

【说明】

车道线列方向关键点数量。

【定义】

```
#define XMEDIA_SVP_LANE_MAX_COL_POINT 40
```

【相关数据类型及接口】

无

XMEDIA_SVP_LANE_FIT_ORDER

【说明】

车道线拟合因子数量。

【定义】

```
#define XMEDIA_SVP_LANE_FIT_ORDER 3
```

【相关数据类型及接口】

无

XMEDIA_SVP_FEATURE_COL

【说明】

人脸特征向量纬度。

【定义】

```
#define XMEDIA_SVP_FEATURE_COL 512
```

【相关数据类型及接口】

无

XMEDIA_SVP_FR_KEYPOINT_NUM

【说明】

人脸关键点数量。

【定义】

```
#define XMEDIA_SVP_FR_KEYPOINT_NUM 5
```

【相关数据类型及接口】

无

XMEDIA_SVP_MAX_VENC_CHN_NUM

【说明】

SVP venc 最大通道数。

【定义】

```
#define XMEDIA_SVP_MAX_VENC_CHN_NUM 16
```

【相关数据类型及接口】

无

XMEDIA_SVP_MAX_VI_PIPE_NUM

【说明】

SVP vi 最大通道数。

【定义】

```
#define XMEDIA_SVP_MAX_VI_PIPE_NUM 8
```

【相关数据类型及接口】

无

XMEDIA_SVP_PLATE_CHAR_MAX_NUM

【说明】

车牌字符最大数量。

【定义】

```
#define XMEDIA_SVP_PLATE_CHAR_MAX_NUM 21
```

【相关数据类型及接口】

无

XMEDIA_SVP_MAX_REGION

【说明】

最大区域数量。

【定义】

```
#define XMEDIA_SVP_MAX_REGION 5
```

【相关数据类型及接口】

无

XMEDIA_SVP_MAX_EDGE

【说明】

区域最大边数。

【定义】

```
#define XMEDIA_SVP_MAX_EDGE 10
```

【相关数据类型及接口】

无

XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM

【说明】

车牌车辆单帧输出目标最大数量。

【定义】

```
#define XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM (10u)
```

【相关数据类型及接口】

无

2.2.3 枚举类型

xmedia_svp_alg_type

【说明】

定义设置算法类别。

【定义】

```
typedef enum {  
    XMEDIA_SVP_ALG_TYPE_PERSON,  
    XMEDIA_SVP_ALG_TYPE_FACE,  
    XMEDIA_SVP_ALG_TYPE_CAR,  
    XMEDIA_SVP_ALG_TYPE_PET,  
    XMEDIA_SVP_ALG_TYPE_HEAD,  
    XMEDIA_SVP_ALG_TYPE_ELECTRIC_BICYCLE,  
    XMEDIA_SVP_ALG_TYPE_NON_MOTORIZED_VEHICLE,  
    XMEDIA_SVP_ALG_TYPE_FACE_RECOGNITION,  
    XMEDIA_SVP_ALG_TYPE_PERSON_KEYPOINT,  
}
```



```

XMEDIA_SVP_ALG_TYPE_FACE_ATTRIBUTE,
XMEDIA_SVP_ALG_TYPE_GESTURE,
XMEDIA_SVP_ALG_TYPE_BOUNDARY_PERSON,
XMEDIA_SVP_ALG_TYPE_BOUNDARY_CAR,
XMEDIA_SVP_ALG_TYPE_OFF_DUTY,
XMEDIA_SVP_ALG_TYPE_PFS,
XMEDIA_SVP_ALG_TYPE_FIREWORKS,
XMEDIA_SVP_ALG_TYPE_PACKAGE,
XMEDIA_SVP_ALG_TYPE_FIRESMOKE,
XMEDIA_SVP_ALG_TYPE_SECOND_STAGE,
XMEDIA_SVP_ALG_TYPE_PERSON_CAR,
XMEDIA_SVP_ALG_TYPE_DMS,
XMEDIA_SVP_ALG_TYPE_LICENSE_DETECT,
XMEDIA_SVP_ALG_TYPE_LICENSE_RECO,
XMEDIA_SVP_ALG_TYPE_LICENSE_COLOR,
XMEDIA_SVP_ALG_TYPE_VEHICLE_DETECT,
XMEDIA_SVP_ALG_TYPE_VEHICLE_RECO,
XMEDIA_SVP_ALG_TYPE_MAX,
} xmedia_svp_alg_type;

```

【成员】

成员名称	描述
XMEDIA_SVP_ALG_TYPE_PERSON	人形检测。
XMEDIA_SVP_ALG_TYPE_FACE	人脸检测。
XMEDIA_SVP_ALG_TYPE_CAR	车形检测。
XMEDIA_SVP_ALG_TYPE_PET	宠物检测。
XMEDIA_SVP_ALG_TYPE_HEAD	人头检测。
XMEDIA_SVP_ALG_TYPE_ELECTRIC_BICYCLE	电梯电动车检测。
XMEDIA_SVP_ALG_TYPE_NON_MOTORIZED_VEHICLE	非机动车检测。
XMEDIA_SVP_ALG_TYPE_FACE_RECOGNITION	人脸识别。
XMEDIA_SVP_ALG_TYPE_PERSON_KEYPOINT	人体关键点检测。
XMEDIA_SVP_ALG_TYPE_FACE_ATTRIBUTE	人脸属性识别。
XMEDIA_SVP_ALG_TYPE_GESTURE	手势识别。

成员名称	描述
XMEDIA_SVP_ALG_TYPE_BOUNDARY_PERSON	周界检测（人）。
XMEDIA_SVP_ALG_TYPE_BOUNDARY_CAR	周界检测（车）。
XMEDIA_SVP_ALG_TYPE_OFF_DUTY	离岗检测。
XMEDIA_SVP_ALG_TYPE_PFS	客流量检测。
XMEDIA_SVP_ALG_TYPE_FIREWORKS	烟火检测。
XMEDIA_SVP_ALG_TYPE_PACKAGE	包裹检测。
XMEDIA_SVP_ALG_TYPE_LICENSE	车牌检测。
XMEDIA_SVP_ALG_TYPE_FIRESMOKE	烟火检测 yolov8。
XMEDIA_SVP_ALG_TYPE_SECOND_STAGE	二阶段检测。
XMEDIA_SVP_ALG_TYPE_PERSON_CAR	人车检测。
XMEDIA_SVP_ALG_TYPE_DMS	DMS 检测。
XMEDIA_SVP_ALG_TYPE_LICENSE_DETECT	车牌检测。
XMEDIA_SVP_ALG_TYPE_LICENSE_RECO	车牌字符识别。
XMEDIA_SVP_ALG_TYPE_LICENSE_COLOR	车牌颜色识别。
XMEDIA_SVP_ALG_TYPE_VEHICLE_DETECT	车辆检测。
XMEDIA_SVP_ALG_TYPE_VEHICLE_RECO	车辆识别。
XMEDIA_SVP_ALG_TYPE_MAX	保留。

xmedia_svp_class_type

【说明】

定义输出某个目标的类别。

【定义】

```
typedef enum {
    XMEDIA_SVP_CLASS_TYPE_PERSON,
    XMEDIA_SVP_CLASS_TYPE_FACE,
    XMEDIA_SVP_CLASS_TYPE_CAR,
```

```

XMEDIA_SVP_CLASS_TYPE_PET,
XMEDIA_SVP_CLASS_TYPE_HEAD,
XMEDIA_SVP_CLASS_TYPE_ELECTRIC_BICYCLE,
XMEDIA_SVP_CLASS_TYPE_MASK,
XMEDIA_SVP_CLASS_TYPE_PERSON_KEYPOINT,
XMEDIA_SVP_CLASS_TYPE_BIKE,
XMEDIA_SVP_CLASS_TYPE_BIKER,
XMEDIA_SVP_CLASS_TYPE_MOTOR,
XMEDIA_SVP_CLASS_TYPE_MOTORER,
XMEDIA_SVP_CLASS_TYPE_TRICYCLE,
XMEDIA_SVP_CLASS_TYPE_TRICYCLER,
XMEDIA_SVP_CLASS_TYPE_FIREWORKS_FIRE,
XMEDIA_SVP_CLASS_TYPE_FIREWORKS_SMOKE,
XMEDIA_SVP_CLASS_TYPE_PACKAGE,
XMEDIA_SVP_CLASS_TYPE_FIRE,
XMEDIA_SVP_CLASS_TYPE_SMOKE,
XMEDIA_SVP_CLASS_TYPE_GESTURE_V,
XMEDIA_SVP_CLASS_TYPE_GESTURE_THUMB,
XMEDIA_SVP_CLASS_TYPE_GESTURE_OK,
XMEDIA_SVP_CLASS_TYPE_GESTURE_PALM,
XMEDIA_SVP_CLASS_TYPE_NOGESTURE,
XMEDIA_SVP_CLASS_TYPE_LICENSE,
XMEDIA_SVP_CLASS_TYPE_MAX,
} xmedia_svp_class_type;

```

【成员】

成员名称	描述
XMEDIA_SVP_ALG_TYPE_PERSON	人形类别。
XMEDIA_SVP_ALG_TYPE_FACE	人脸类别。
XMEDIA_SVP_ALG_TYPE_CAR	车形类别。
XMEDIA_SVP_ALG_TYPE_NONCAR	非机动车类别。
XMEDIA_SVP_ALG_TYPE_PET	宠物类别。
XMEDIA_SVP_ALG_TYPE_HEAD	人头类别。
XMEDIA_SVP_ALG_TYPE_ELECTRIC_BICYCLE	电梯电动车类别。
XMEDIA_SVP_CLASS_TYPE_MASK	口罩类别。

成员名称	描述
XMEDIA_SVP_CLASS_TYPE_PERSON_KEYPOINT	人体关键点类别。
XMEDIA_SVP_CLASS_TYPE_BIKE	自行车类别。
XMEDIA_SVP_CLASS_TYPE_BIKER	骑自行车的人类别。
XMEDIA_SVP_CLASS_TYPE_MOTOR	电动车类别。
XMEDIA_SVP_CLASS_TYPE_MOTORER	骑电动车的人类别。
XMEDIA_SVP_CLASS_TYPE_TRICYCLE	三轮车类别。
XMEDIA_SVP_CLASS_TYPE_TRICYCLER	骑三轮车的人类别。
XMEDIA_SVP_CLASS_TYPE_FIREWORKS_FIRE	火。
XMEDIA_SVP_CLASS_TYPE_FIREWORKS_SMOKE	烟。
XMEDIA_SVP_CLASS_TYPE_PACKAGE	包裹。
XMEDIA_SVP_CLASS_TYPE_FIRE	火 yolov8。
XMEDIA_SVP_CLASS_TYPE_SMOKE	烟 yolov8。
XMEDIA_SVP_CLASS_TYPE_GESTURE_V	手势剪刀手。
XMEDIA_SVP_CLASS_TYPE_GESTURE_THUMB	手势大拇指。
XMEDIA_SVP_CLASS_TYPE_GESTURE_OK	手势 OK。
XMEDIA_SVP_CLASS_TYPE_GESTURE_PALM	手势手掌。
XMEDIA_SVP_CLASS_TYPE_NOGESTURE	无手势。
XMEDIA_SVP_CLASS_TYPE_LICENSE	车牌类别。
XMEDIA_SVP_CLASS_TYPE_MAX	保留。

xmedia_svp_inputdata_format

【说明】

定义输入数据格式。

【定义】

```
typedef enum {
```

```

XMEDIA_SVP_INPUTDATA_FORMAT_YUV420SP,
XMEDIA_SVP_INPUTDATA_FORMAT_RGB888,
XMEDIA_SVP_INPUTDATA_FORMAT_MAX,
} xmedia_svp_inputdata_format;

```

【成员】

成员名称	描述
XMEDIA_SVP_INPUTDATA_FORMAT_YUV420SP	输入图像格式 YUV420SP。
XMEDIA_SVP_INPUTDATA_FORMAT_RGB888	输入图像格式 RGB888。
XMEDIA_SVP_INPUTDATA_FORMAT_MAX	保留。

xmedia_svp_model_type

【说明】

定义输入模型模式。

【定义】

```

typedef enum {
    XMEDIA_SVP_MODEL_FILE,
    XMEDIA_SVP_MODEL_MEM
} xmedia_svp_model_type;

```

【成员】

成员名称	描述
XMEDIA_SVP_MODEL_FILE	模型文件输入。
XMEDIA_SVP_MODEL_MEM	模型内存输入。

【相关数据类型及接口】

无

xmedia_svp_motion_state

【说明】

定义目标运动状态类型。

【定义】

```
typedef enum {
    XMEDIA_SVP_MOTION_STATE_UNKNOWN = 0,
    XMEDIA_SVP_MOTION_STATE_STATIC,
    XMEDIA_SVP_MOTION_STATE_MOVING,
} xmedia_svp_motion_state;
```

【成员】

成员名称	描述
XMEDIA_SVP_MOTION_STATE_UNKNOWN	目标运动状态确认中。
XMEDIA_SVP_MOTION_STATE_STATIC	目标运动状态静止。
XMEDIA_SVP_MOTION_STATE_MOVING	目标运动状态运动。

xmedia_svp_task_type

【说明】

定义 SVP 任务类型。

【定义】

```
typedef enum {
    XMEDIA_SVP_TASK_UAV,
    XMEDIA_SVP_TASK_DETECT,
    XMEDIA_SVP_TASK_AOV,
    XMEDIA_SVP_TASK_DETECT_AND_KEYPOINT,
    XMEDIA_SVP_TASK_ADAS,
    XMEDIA_SVP_TASK_FIRE,
    XMEDIA_SVP_TASK_GESTURE,
    XMEDIA_SVP_TASK_FACE_RECOGNITION,
    XMEDIA_SVP_TASK_EMOTION_CLASSIFICATION,
    XMEDIA_SVP_TASK_RCNN,
    XMEDIA_SVP_TASK_RCNN_720P,
    XMEDIA_SVP_TASK_COMPANION_ROBOT,
    XMEDIA_SVP_TASK_MULTI_DETECT,
    XMEDIA_SVP_TASK_DMS,
    XMEDIA_SVP_TASK_PLATE,
    XMEDIA_SVP_TASK_VEHICLE,
    XMEDIA_SVP_TASK_PLATE_VEHICLE,
    XMEDIA_SVP_TASK_MAX,
} xmedia_svp_task_type;
```

【成员】

成员名称	描述
XMEDIA_SVP_TASK_UAV	无人机任务。
XMEDIA_SVP_TASK_DETECT	基础检测任务。
XMEDIA_SVP_TASK_AOV	AOV 检测任务。
XMEDIA_SVP_TASK_DETECT_AND_KEYPOINT	Yolov8 关键点检测任务。
XMEDIA_SVP_TASK_ADAS	ADAS 任务。
XMEDIA_SVP_TASK_FIRE	烟火检测 (yolov8) 任务。
XMEDIA_SVP_TASK_GESTURE	手势识别任务。
XMEDIA_SVP_TASK_FACE_RECOGNITION	人脸识别任务。
XMEDIA_SVP_TASK_EMOTION_CLASSIFICATION	表情分类任务。
XMEDIA_SVP_TASK_RCNN	RCNN 二阶段任务。
XMEDIA_SVP_TASK_RCNN_720P	RCNN720P 二阶段任务。
XMEDIA_SVP_TASK_COMPANION_ROBOT	陪伴机器人任务。
XMEDIA_SVP_TASK_MULTI_DETECT	多类别基础检测任务。
XMEDIA_SVP_TASK_DMS	DMS 任务。
XMEDIA_SVP_TASK_PLATE	车牌识别任务。
XMEDIA_SVP_TASK_VEHICLE	车辆识别任务。
XMEDIA_SVP_TASK_PLATE_VEHICLE	车牌车辆识别任务。
XMEDIA_SVP_TASK_MAX	保留。

xmedia_svp_class_emotion_type

【说明】

定义表情类型。

【定义】

```
typedef enum {
```

```

XMEDIA_SVP_CLASS_EMOTION_SMILE,
XMEDIA_SVP_CLASS_EMOTION_NORMAL,
XMEDIA_SVP_CLASS_EMOTION_UNKNOWN,
} xmedia_svp_class_emotion_type;

```

【成员】

成员名称	描述
XMEDIA_SVP_CLASS_EMOTION_SMILE	微笑表情。
XMEDIA_SVP_CLASS_EMOTION_NORMAL	正常表情。
XMEDIA_SVP_CLASS_EMOTION_UNKNOWN	未知表情。

【相关数据类型及接口】

[xmedia_svp_face_attribute_result](#)

[xmedia_svp_mem_reuse_type](#)

【说明】

定义内存复用类。

【定义】

```

typedef enum {
    XMEDIA_SVP_MEM_TYPE_BLOCK,
    XMEDIA_SVP_MEM_TYPE_AINR_SHARE,
    XMEDIA_SVP_MEM_TYPE_COMPLETE,
    XMEDIA_SVP_MEM_TYPE_MAX,
} xmedia_svp_mem_reuse_type;

```

【成员】

成员名称	描述
XMEDIA_SVP_MEM_TYPE_BLOCK	SVP 多模型任务 workspace/input/output 内存分别复用。
XMEDIA_SVP_MEM_TYPE_AINR_SHARE	SVP 多模型任务 workspace/input/output 内存分别复用，同时与 ISP 复用 workspace。

成员名称	描述
XMEDIA_SVP_MEM_TYPE_COMPLETE	暂不支持。
XMEDIA_SVP_MEM_TYPE_MAX	内存不复用。。

【相关数据类型及接口】

[xmedia_svp_cfg](#)

xmedia_svp_direction

【说明】

定义车牌车辆方向。

【定义】

```
typedef enum {
    XMEDIA_SVP_DIRECTION_FORWARD,
    XMEDIA_SVP_DIRECTION_BACKWARD,
    XMEDIA_SVP_DIRECTION_UNKOWN = 99,
} xmedia_svp_direction;
```

【成员】

成员名称	描述
XMEDIA_SVP_DIRECTION_FORWARD	向前。
XMEDIA_SVP_DIRECTION_BACKWARD	向后。
XMEDIA_SVP_DIRECTION_UNKOWN	未知。

【相关数据类型及接口】

- [xmedia_svp_plate_result](#)
- [xmedia_svp_vehicle_result](#)
- [xmedia_svp_plate_vch_result](#)

xmedia_svp_plate_color

【说明】

定义车牌颜色。

【定义】

```
typedef enum {
    XMEDIA_SVP_PLATE_COLOR_BLACK    = 0,
    XMEDIA_SVP_PLATE_COLOR_BULE     = 1,
    XMEDIA_SVP_PLATE_COLOR_GREEN    = 2,
    XMEDIA_SVP_PLATE_COLOR_RESEVER  = 3,
    XMEDIA_SVP_PLATE_COLOR_WHITE    = 4,
    XMEDIA_SVP_PLATE_COLOR_PURPLE   = 5,
    XMEDIA_SVP_PLATE_COLOR_YELLOW   = 6,
    XMEDIA_SVP_PLATE_COLOR_UNKNOWN  = 7,
} xmedia_svp_plate_color;
```

【成员】

成员名称	描述
XMEDIA_SVP_PLATE_COLOR_BLACK	黑牌。
XMEDIA_SVP_PLATE_COLOR_BULE	蓝牌。
XMEDIA_SVP_PLATE_COLOR_GREEN	绿牌。
XMEDIA_SVP_PLATE_COLOR_RESEVER	保留。
XMEDIA_SVP_PLATE_COLOR_WHITE	白牌。
XMEDIA_SVP_PLATE_COLOR_PURPLE	紫牌。
XMEDIA_SVP_PLATE_COLOR_YELLOW	黄牌。
XMEDIA_SVP_PLATE_COLOR_UNKNOWN	未知。

【相关数据类型及接口】

- [xmedia_svp_plate_result](#)
- [xmedia_svp_plate_veh_result](#)

xmedia_svp_vehicle_type

【说明】

定义车辆类型。

【定义】

```
typedef enum {
    XMEDIA_SVP_VEHICLE_TYPE_SEDAN,
    XMEDIA_SVP_VEHICLE_TYPE_MINIVAN,
```

```

XMEDIA_SVP_VEHICLE_TYPE_TRUCK,
XMEDIA_SVP_VEHICLE_TYPE_TANKER
XMEDIA_SVP_VEHICLE_TYPE_MINIBUS,
XMEDIA_SVP_VEHICLE_TYPE_UNKNOWN = 99,
} xmedia_svp_vehicle_type;

```

【成员】

成员名称	描述
XMEDIA_SVP_VEHICLE_TYPE_SEDAN	小车。
XMEDIA_SVP_VEHICLE_TYPE_MINIVAN	小型货车。
XMEDIA_SVP_VEHICLE_TYPE_TRUCK	卡车。
XMEDIA_SVP_VEHICLE_TYPE_TANKER	罐车。
XMEDIA_SVP_VEHICLE_TYPE_MINIBUS	巴士。
XMEDIA_SVP_VEHICLE_TYPE_UNKNOWN	未知。

【相关数据类型及接口】

- [xmedia_svp_vehicle_result](#)
- [xmedia_svp_plate_veh_result](#)

xmedia_svp_vehicle_color

【说明】

定义车辆颜色。

【定义】

```

typedef enum {
    XMEDIA_SVP_VEHICLE_COLOR_WHITE,
    XMEDIA_SVP_VEHICLE_COLOR_RED,
    XMEDIA_SVP_VEHICLE_COLOR_YELLOW,
    XMEDIA_SVP_VEHICLE_COLOR_BLACK,
    XMEDIA_SVP_VEHICLE_COLOR_BLUE,
    XMEDIA_SVP_VEHICLE_COLOR_GREEN,
    XMEDIA_SVP_VEHICLE_COLOR_UNKNOWN = 99,
} xmedia_svp_vehicle_color;

```

【成员】

成员名称	描述
XMEDIA_SVP_VEHICLE_COLOR_WHITE	白色。
XMEDIA_SVP_VEHICLE_COLOR_RED	红色。
XMEDIA_SVP_VEHICLE_COLOR_YELLOW	黄色。
XMEDIA_SVP_VEHICLE_COLOR_BLACK	黑色。
XMEDIA_SVP_VEHICLE_COLOR_BLUE	蓝色。
XMEDIA_SVP_VEHICLE_COLOR_GREEN	绿色。
XMEDIA_SVP_VEHICLE_COLOR_UNKNOWN	未知。

【相关数据类型及接口】

- [xmedia_svp_vehicle_result](#)
- [xmedia_svp_plate_veh_result](#)

2.2.4 结构体

xmedia_svp_rect

【说明】

定义矩形坐标结构体。

【定义】

```
typedef struct {
    xmedia_float x1;
    xmedia_float y1;
    xmedia_float x2;
    xmedia_float y2;
} xmedia_svp_rect;
```

【成员】

成员名称	描述
x1	矩形左上角 x 坐标。
y1	矩形左上角 y 坐标。
x2	矩形右下角 x 坐标。

成员名称	描述
y2	矩形右下角 y 坐标。

【相关数据类型及接口】

[xmedia_svp_detect_result](#)

xmedia_svp_detect_result

【说明】

定义目标检测结果信息结构体。

【定义】

```
typedef struct {
    xmedia\_svp\_alg\_type alg_type;
    xmedia\_svp\_class\_type class_type;
    xmedia_float detect_score;
    xmedia_float classfier_score;
    xmedia_s32 tracker_id;
    xmedia_u32 tracker_age;
    xmedia\_svp\_rect rect;
    xmedia_bool special_target;
    xmedia_float distance;
    xmedia\_svp\_motion\_state motion_state;
} xmedia_svp_detect_result;
```

【成员】

成员名称	描述
alg_type	检测类别。
class_type	检测结果类别。
detect_score	检测模型结果置信度。
classfier_score	分类器结果置信度。
tracker_id	检测结果追踪 id。
tracker_age	检测结果追踪 age。
rect	检测结果目标框。

成员名称	描述
special_target	特殊目标。
distance	检测目标距离。
motion_state	运动状态。

【相关数据类型及接口】

[xmedia_svp_yolov5_output](#)

xmedia_svp_point

【说明】

定义单点坐标结构体。

【定义】

```
typedef struct {
    xmedia_float x;
    xmedia_float y;
} xmedia_svp_point;
```

【成员】

成员名称	描述
x	x 坐标。
y	y 坐标。

xmedia_svp_area_limit_attr

【说明】

定义目标框面积限制结构体。

【定义】

```
typedef struct {
    xmedia_bool enable;
    xmedia_float min_area;
    xmedia_float max_area;
} xmedia_svp_area_limit_attr;
```

【成员】

成员名称	描述
enable	框面积大小限制是否使能。
min_area	最小框面积。
max_area	最大框面积。

【相关数据类型及接口】

无

xmedia_svp_edge_limit_attr

【说明】

定义目标框边缘限制结构体。

【定义】

```
typedef struct {
    xmedia_bool enable;
    xmedia_float w_edge;
    xmedia_float h_edge;
} xmedia_svp_edge_limit_attr;
```

【成员】

成员名称	描述
enable	边缘过滤是否使能。
w_edge	宽度过滤距离(目标框左右边缘与图像左右边界的距离)。
h_edge	高度过滤距离(目标框上下边缘与图像上下边界的距离)。

【相关数据类型及接口】

无

xmedia_svp_task_input

【说明】

定义任务输入帧存信息。

【定义】

```
typedef struct {  
    xmedia_video_frame_info *frame;  
    xmedia_u8 frame_num;  
} xmedia_svp_task_input;
```

【成员】

成员名称	描述
frame	视频帧存结构体。
frame_num	输入帧存数量。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_modules

【说明】

定义模型信息。

【定义】

```
typedef struct {  
    xmedia\_svp\_model\_type load_mode;  
    xmedia\_svp\_inputdata\_format format;  
    xmedia_char *pathname;  
    xmedia_u8 *buf;  
    xmedia_u32 len;  
    xmedia\_svp\_alg\_type alg_type;  
    xmedia_void *priv;  
} xmedia_svp_modules;
```

【成员】

成员名称	描述
load_mode	模型加载模式。
format	模型输入图像格式。
pathname	模型文件。

成员名称	描述
buf	模型存放的内存。
len	模型的长度。
alg_type	模型检测类别。
priv	模型私有数据。

【相关数据类型及接口】

[xmedia_svp_task_cfg](#)

xmedia_svp_task_cfg

【说明】

定义任务信息。

【定义】

```
typedef struct {
```

```
    xmedia_svp_task_type task_type;
```

```
    xmedia_svp_modules *modules;
```

```
    xmedia_u8 module_num;
```

```
    xmedia_void *priv;
```

```
} xmedia_svp_task_cfg;
```

【成员】

成员名称	描述
task_type	任务类型。
modules	模型数据。
module_num	模型数量。
*priv	私有数据。

【相关数据类型及接口】

[xmedia_svp_task_create](#)

xmedia_svp_detect_base_attr

【说明】

定义检测基础属性信息。

【定义】

```
typedef struct {  
    xmedia_float detect_threshold;  
    xmedia_float classifier_threshold;  
    xmedia_float iou_threshold;  
    xmedia_u32 max_target_num;  
} xmedia_svp_detect_base_attr;
```

【成员】

成员名称	描述
detect_threshold	检测阈值。
classifier_threshold	分类器阈值。
iou_threshold	iou 相交比阈值。
max_target_num	单帧检测最大目标数。

xmedia_svp_deepsort_attr

【说明】

定义 deepsort 算法属性结构体。

【定义】

```
typedef struct {  
    xmedia_bool enable;  
    xmedia_float max_feature_distance;  
    xmedia_float min_feature_distance;  
    xmedia_float max_iou_distance;  
    xmedia_u32 iou2feature;  
    xmedia_u32 max_age;  
} xmedia_svp_deepsort_attr;
```

【成员】

成员名称	描述
enable	追踪使能。
max_feature_distance	追踪特征匹配距离最大阈值。

成员名称	描述
min_feature_distance	追踪特征匹配距离最小阈值。
max_iou_distance	追踪 iou 匹配距离阈值。
iou2feature	追踪 iou 匹配变成确认态。
max_age	追踪 age 阈值。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

[xmedia_svp_uav_attr](#)

【说明】

定义无人机任务属性结构体。

【定义】

```
typedef struct {
    xmedia\_svp\_detect\_base\_attr person_attr;
    xmedia\_svp\_detect\_base\_attr head_attr;
    xmedia\_svp\_detect\_base\_attr gesture_attr;
    xmedia\_svp\_deepsort\_attr deepsort_attr;
    xmedia_u32 focal_length_px;
    xmedia_u32 head_width_cm;
    xmedia_u32 palm_width_cm;
} xmedia_svp_uav_attr;
```

【成员】

成员名称	描述
person_attr	人形检测属性。
head_attr	人头检测属性。
gesture_attr	手势检测属性。
deepsort_attr	deepsort 算法属性结构体。
focal_length_px	像素焦距。
head_width_cm	人头宽度。

成员名称	描述
palm_width_cm	手掌宽度。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_uav_output

【说明】

定义无人机任务检测输出结构体。

【定义】

```
typedef struct {
    xmedia_svp_detect_result track_result;
    xmedia_svp_detect_result head_result;
    xmedia_svp_detect_result gesture_result;
    xmedia_float head_distance;
    xmedia_bool is_hover;
    xmedia_float gesture_distance;
} xmedia_svp_uav_output;
```

【成员】

成员名称	描述
track_result	追踪结果。
head_result	人头结果。
gesture_result	手势结果。
head_distance	人头距离。
is_hover	是否悬停状态。
gesture_distance	手势距离。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_yolov5_attr

【说明】

定义 yolov5 系检测任务属性。

【定义】

```
typedef struct {  
    xmedia_float detect_threshold;  
    xmedia_float classifier_threshold;  
    xmedia_float iou_threshold;  
    xmedia_u32 max_target_num;  
    xmedia_bool bytetrack_enable;  
    xmedia_bool motionless_filter_enable;  
    xmedia_float stillness_thres;  
    xmedia_u8 movement_fps_thres;  
    xmedia_bool smart_venc_enable;  
    xmedia_bool smart_ae_enable;  
    xmedia_bool smart_venc_array[XMEDIA_SVP_MAX_VENC_CHN_NUM];  
    xmedia_bool smart_ae_array[XMEDIA_SVP_MAX_VI_PIPE_NUM];  
} xmedia_svp_yolov5_attr;
```

【成员】

成员名称	描述
detect_threshold	置信度阈值，建议值 0.5f。
classifier_threshold	分类器阈值，建议值 0.01f。
iou_threshold	iou 相交比阈值，建议值 0.5f。
max_target_num	单帧检测最大目标数。
bytetrack_enable	追踪开关。
motionless_filter_enable	运动检测开关。
stillness_thres	由运动转为静止状态的灵敏度阈值。
movement_fps_thres	由运动转为静止状态的连续帧阈值。
smart_venc_enable	智能编码使能开关。 (仅人形、人脸支持)
smart_ae_enable	智能曝光使能开关。 (仅人形、人脸支持)

成员名称	描述
smart_venc_array [XMEDIA_SVP_MAX_VENC_CHN_NUM]	智能编码支持的 VENC 通道数组。
smart_ae_array [XMEDIA_SVP_MAX_VI_PIPE_NUM]	智能曝光支持的 VI 通道数组。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_yolov5_output

【说明】

定义 yolov5 系任务输出信息。

【定义】

```
typedef struct {
    xmedia_u32 target_num;
    xmedia_svp_detect_result targets[XMEDIA_SVP_MAX_TARGET_NUM];
} xmedia_svp_yolov5_output;
```

【成员】

成员名称	描述
target_num	检测目标数量。
targets[XMEDIA_SVP_MAX_TARGET_NUM]	检测目标。

【适用模型】

人形检测、人脸检测、车形检测、非机动车检测、宠物检测等自研 yolov5 模型。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_aov_attr

【说明】

定义 aov 检测任务属性。

【定义】

```
typedef struct {
    xmedia_float detect_threshold;
    xmedia_float classifier_threshold;
    xmedia_float iou_threshold;
    xmedia_u32 max_target_num;
    xmedia_bool bytetrack_enable;
    xmedia_bool motionless_filter_enable;
    xmedia_bool aov_only_target;
} xmedia_svp_aov_attr;
```

【成员】

成员名称	描述
detect_threshold	置信度阈值，建议值 0.75f。
classifier_threshold	分类器阈值，建议值 0.01f。
iou_threshold	iou 相交比阈值，建议值 0.5f。
max_target_num	单帧检测最大目标数。
bytetrack_enable	追踪开关。
motionless_filter_enable	静止过滤开关。
aov_only_target	仅获取目标有无（开启后仅输出目标有无，不输出坐标等信息）。

【适用模型】

自研 aov 模型。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

[xmedia_svp_aov_detect_output](#)

【说明】

定义 aov 任务输出信息

【定义】

```
typedef struct {
    xmedia_bool is_exist;
    xmedia\_svp\_yolov5\_output detect_output;
```

```
} xmedia_svp_aov_detect_output;
```

【成员】

成员名称	描述
is_exist	输出有无目标。
detect_output	目标检测结果。

【适用模型】

自研 aov 模型。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_keypoint

【说明】

定义关键点结构体。

【定义】

```
typedef struct {  
    xmedia_float x;  
    xmedia_float y;  
    xmedia_float score  
} xmedia_svp_keypoint;
```

【成员】

成员名称	描述
x	关键点的 x 坐标。
y	关键点的 y 坐标。
score	关键点的得分。

【相关数据类型及接口】

无

xmedia_svp_yolov8_attr

【说明】

定义 yolov8 系检测任务属性。

【定义】

```
typedef struct {  
    xmedia_float detect_threshold;  
    xmedia_float classifier_threshold;  
    xmedia_float iou_threshold;  
    xmedia_u32 max_target_num;  
    xmedia_bool bytetrack_enable;  
    xmedia_bool motionless_filter_enable;  
    xmedia_float stillness_thres;  
    xmedia_u8 movement_fps_thres;  
} xmedia_svp_yolov8_attr;
```

【成员】

成员名称	描述
detect_threshold	置信度阈值，建议值 0.65f。
classifier_threshold	分类器阈值，建议值 0.01f。
iou_threshold	iou 相交比阈值，建议值 0.5f。
max_target_num	单帧检测最大目标数。
bytetrack_enable	追踪开关。
motionless_filter_enable	运动状态检测开关。
stillness_thres	由运动转为静止状态的灵敏度阈值。
movement_fps_thres	由运动转为静止状态的连续帧阈值。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_yolov8_result

【说明】

定义 yolov8 系单目标检测结果。

【定义】

```
typedef struct {
```

```

xmedia_svp_alg_type alg_type;
xmedia_svp_class_type class_type;
xmedia_float detect_score;
xmedia_float classifier_score;
xmedia_s32 tracker_id;
xmedia_u32 tracker_age;
xmedia_svp_rect rect;
xmedia_svp_keypoint kpt[MAX_DETECT_KEYPOINT_NUM];
xmedia_svp_motion_state motion_state;
} xmedia_svp_yolov8_result;

```

【成员】

成员名称	描述
alg_type	检测类别。
class_type	检测目标类别。
detect_score	检测模型结果置信度。
classifier_score	分类器结果置信度。
tracker_id	检测目标追踪 id。
tracker_age	检测结果追踪 age
rect	检测目标框。
kpt[MAX_DETECT_KEYPOINT_NUM]	关键点坐标。
motion_state	运动状态。

【相关数据类型及接口】

xmedia_svp_yolov8_output

xmedia_svp_yolov8_output

【说明】

定义 yolov8 系任务输出信息。

【定义】

```

typedef struct {
    xmedia_u32 target_num;
    xmedia_svp_yolov8_result targets[XMEDIA_SVP_MAX_TARGET_NUM];
}

```

```
} xmedia_svp_yolov8_output;
```

【成员】

成员名称	描述
target_num	检测目标数量。
targets[XMEDIA_SVP_MAX_TARGET_NUM]	检测目标。

【适用模型】

人体关键点等自研 yolov8 模型。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_adas_distance_attr

【说明】

定义 ADAS 测距属性结构体。

【定义】

```
typedef struct {  
    xmedia_u32 focal_length_px;  
    xmedia_u32 car_limit_angle;  
    xmedia_float camera_height;  
    xmedia_u32 plate_width;  
} xmedia_svp_adas_distance_attr;
```

【成员】

成员名称	描述
focal_length_px	焦距像素。
car_limit_angle	车辆相机角度。
camera_height	相机高度。
plate_width	车牌宽度。

【相关数据类型及接口】

[xmedia_svp_adas_attr](#)

xmedia_svp_adas_attr

【说明】

定义 ADAS 属性结构体。

【定义】

```
typedef struct {  
    xmedia_svp_yolov5_attr car_attr;  
    xmedia_svp_yolov5_attr plate_attr;  
    xmedia_svp_yolov5_attr person_attr;  
    xmedia_svp_yolov5_attr nmv_attr;  
    xmedia_svp_adas_distance_attr distance_attr;  
} xmedia_svp_adas_attr;
```

【成员】

成员名称	描述
car_attr	车辆检测相关属性。
plate_attr	车牌检测相关属性。
person_attr	人形检测相关属性。
nmv_attr	非机动车相关属性。
distance_attr	测距相关属性。

【相关数据类型及接口】

- [xmedia_svp_task_set_attr](#)
- [xmedia_svp_yolov5_attr](#)

xmedia_svp_adas_result

【说明】

定义 ADAS 输出结果。

【定义】

```
typedef struct {  
    xmedia_s32 lane_point_num[XMEDIA_SVP_LANE_MAX_NUM];  
    xmedia_s32 lane_fitted_point_num[XMEDIA_SVP_LANE_MAX_NUM];  
    xmedia_svp_keypoint  
    row_result[XMEDIA_SVP_LANE_AXIS_NUM][XMEDIA_SVP_LANE_MAX_ROW_POINT];  
}
```

```

    xmedia_svp_keypoint
col_result[XMEDIA_SVP_LANE_AXIS_NUM][XMEDIA_SVP_LANE_MAX_COL_POINT];
    xmedia_svp_keypoint
row_fitted_result[XMEDIA_SVP_LANE_AXIS_NUM][XMEDIA_SVP_LANE_MAX_ROW_POINT];
    xmedia_svp_keypoint
col_fitted_result[XMEDIA_SVP_LANE_AXIS_NUM][XMEDIA_SVP_LANE_MAX_COL_POINT];
    xmedia_svp_yolov5_output car_result;
    xmedia_svp_yolov5_output plate_result;
    xmedia_svp_yolov5_output person_result;
    xmedia_svp_yolov5_output nmv_result;
    xmedia_float coeffs[XMEDIA_SVP_LANE_MAX_NUM][XMEDIA_SVP_LANE_FIT_ORDER];
} xmedia_svp_adas_result;

```

【成员】

成员名称	描述
lane_point_num	每条车道检测到的原始关键点数量。
lane_fitted_point_num	每条车道经过曲线拟合后的关键点数量。
row_result	原始行方向关键点。
col_result	原始列方向关键点。
row_fitted_result	拟合后的行方向关键点。
col_fitted_result	拟合后的列方向关键点。
car_result	车辆检测结果。
plate_result	车牌检测结果。
person_result	人形检测结果。
nmv_result	非机动车检测结果。
coeffs	多项式拟合系数。

【相关数据类型及接口】

xmedia_svp_task_process

xmedia_svp_fire_attr

【说明】

定义火焰检测（yolov8）属性结构体。

【定义】

```
typedef struct {  
    xmedia_u8 smoke_appear_count;  
    xmedia_u8 smoke_loss_count;  
    xmedia_float smoke_change_ratio;  
    xmedia_u8 fire_loss_count;  
    xmedia_float fire_change_ratio;  
    xmedia_float fire_ratio_sum;  
    xmedia_svp_yolov8_attr detect_attr;  
} xmedia_svp_fire_attr;
```

【成员】

成员名称	描述
smoke_appear_count	烟出现的帧数。
smoke_loss_count	烟消失的帧数计数。
smoke_change_ratio	烟的变化率。
fire_loss_count	火消失的帧数计数。
fire_change_ratio	火的变化率。
fire_ratio_sum	火的变化率总合。
detect_attr	yolov8 系检测相关属性。

【相关数据类型及接口】

- [xmedia_svp_task_set_attr](#)
- [xmedia_svp_yolov8_attr](#)

xmedia_svp_gesture_result

【说明】

定义手势识别结果结构体。

【定义】

```
typedef struct {  
    xmedia_svp_rect gesture_rect;  
    xmedia_s32 gesture_tracker_id;  
    xmedia_float select_score[XMEDIA_SVP_MAX_TARGET_NUM];  
}
```

```
xmedia_svp_yolov8_output result;
} xmedia_svp_gesture_result;
```

【成员】

成员名称	描述
gesture_rect	手势矩形框。
gesture_tracker_id	手势追踪 id。
select_score[XMEDIA_SVP_MAX_TARGET_NUM]	手势得分。
result	手势结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

[xmedia_svp_face_pose_attr](#)

【说明】

定义人脸姿态结构体。

【定义】

```
typedef struct {
    xmedia_float clockwise_thres;
    xmedia_float anticlockwise_thres;
    xmedia_float left_thres;
    xmedia_float right_thres;
    xmedia_float upward_thres;
    xmedia_float downward_thres;
} xmedia_svp_face_pose_attr;
```

【成员】

成员名称	描述
clockwise_thres	顺时针阈值。
anticlockwise_thres	逆时针阈值。
left_thres	左转阈值。
right_thres	右转阈值。

成员名称	描述
upward_thres	抬头阈值。
downward_thres	低头阈值。

【适用模型】

自研人脸识别模型。

【相关数据类型及接口】

[xmedia_svp_fr_attr](#)

xmedia_svp_fr_attr

【说明】

定义人脸识别属性结构体。

【定义】

```
typedef struct {
    xmedia\_svp\_yolov5\_attr detect_attr;
    xmedia_u64 luma_score;
    xmedia_s32 ive_sobel_score;
    xmedia\_svp\_face\_pose\_attr fr_pose_attr;
    xmedia_float keypoint_score ;
} xmedia_svp_fr_attr;
```

【成员】

成员名称	描述
tracker_attr	人脸识别追踪属性。
luma_score	人脸亮度分数。
ive_sobel_score	sobel 边缘评分。
fr_pose_attr	人脸姿态属性。
keypoint_score	人脸关键点得分。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_fr_result

【说明】

定义人脸识别结果结构体。

【定义】

```
typedef struct {  
    xmedia_float vector[XMEDIA_SVP_FEATURE_COL];  
    xmedia_svp_rect rect;  
    xmedia_s32 fr_tracker_id ;  
    xmedia_float score;  
    xmedia_svp_point keypoint_arr[XMEDIA_SVP_FR_KEYPOINT_NUM];  
} xmedia_svp_fr_result;
```

【成员】

成员名称	描述
vector[XMEDIA_SVP_FEATURE_COL]	人脸特征向量。
rect	目标的矩阵坐标点。
fr_tracker_id	人脸跟踪 id。
score	人脸分数。
keypoint_arr	人脸的关键点。

【相关数据类型及接口】

[xmedia_svp_fr_output](#)

xmedia_svp_fr_output

【说明】

定义人脸识别任务输出信息。

【定义】

```
typedef struct {  
    xmedia_s32 face_num;  
    xmedia_svp_fr_result fr_result[XMEDIA_SVP_MAX_TARGET_NUM];  
} xmedia_svp_fr_output;
```

【成员】

成员名称	描述
face_num	人脸识别结果数量。
fr_result	人脸识别结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

[xmedia_svp_face_attribute_result](#)

【说明】

定义人脸属性结构体。

【定义】

```
typedef struct {
    xmedia\_svp\_rect rect;
    xmedia_s32 face_tracker_id;
    xmedia_float score ;
    xmedia\_svp\_point keypoint_array[XMEDIA_SVP_FR_KEYPOINT_NUM];
    xmedia\_svp\_class\_emotion\_type emotion_class;
} xmedia_svp_face_attribute_result;
```

【成员】

成员名称	描述
rect	人脸目标框坐标。
face_tracker_id	人脸追踪 ID。
score	人脸质量分数。
keypoint_array	人脸关键点坐标。
emotion_class	表情类别。

【相关数据类型及接口】

- [xmedia_svp_class_emotion_type](#)
- [xmedia_svp_face_attribute_output](#)

[xmedia_svp_face_attribute_output](#)

【说明】

定义人脸属性识别任务结果。

【定义】

```
typedef struct {  
    xmedia_s32 face_num;  
    xmedia_svp_face_attribute_result attribute_result[XMEDIA_SVP_MAX_TARGET_NUM];  
} xmedia_svp_face_attribute_output;
```

【成员】

成员名称	描述
face_num	人脸属性识别结果数量。
attribute_result	人脸属性识别结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_companion_robot_output

【说明】

定义陪伴机器人任务结果输出结构体。

【定义】

```
typedef struct {  
    xmedia_u32 target_num;  
    xmedia_svp_detect_result targets[XMEDIA_SVP_MAX_TARGET_NUM];  
} xmedia_svp_companion_robot_output;
```

【成员】

成员名称	描述
target_num	陪伴机器人识别结果数量。
targets	陪伴机器人识别结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_mem_info

【说明】

定义内存信息结构体。

【定义】

```
typedef struct {  
    xmedia_u32 size;  
    xmedia_u64 phyaddr;  
    xmedia_void *viraddr;  
} xmedia_svp_mem_info;
```

【成员】

成员名称	描述
size	内存大小。
phyaddr	物理地址。
*viradd	虚拟地址。

【相关数据类型及接口】

[xmedia_svp_cfg](#)

[xmedia_svp_cfg](#)

【说明】

定义模型内存共用结构体。

【定义】

```
typedef struct {  
    xmedia\_svp\_mem\_reuse\_type reuse_type;  
    xmedia\_svp\_mem\_info workbuf_reuse_mem;  
    xmedia\_svp\_mem\_info input_reuse_mem;  
    xmedia\_svp\_mem\_info output_reuse_mem;  
} xmedia_svp_cfg;
```

【成员】

成员名称	描述
reuse_type	内存共用类型。
workbuf_reuse_mem	workbuff 内存共用信息。
input_reuse_mem	input 内存共用信息。

成员名称	描述
output_reuse_mem	output 内存共用信息。

【相关数据类型及接口】

- [xmedia_svp_get_config](#)
- [xmedia_svp_set_config](#)

xmedia_svp_multi_base_attr

【说明】

定义多类别检测任务基本属性结构体。

【定义】

```
typedef struct {
    xmedia_float detect_threshold;
    xmedia_float classifier_threshold;
    xmedia_float iou_threshold;
    xmedia_u8 max_target_num;
} xmedia_svp_multi_base_attr;
```

【成员】

成员名称	描述
detect_threshold	置信度阈值。
classifier_threshold	分类器阈值。
iou_threshold	iou 相交比阈值。
max_target_num	单个检测任务的最大目标数，所有检测任务总和最大值 50。

【相关数据类型及接口】

- [xmedia_svp_multi_detect_attr](#)

xmedia_svp_multi_detect_attr

【说明】

定义多类别检测任务属性结构体。

【定义】

```
typedef struct {  
    xmedia_svp_multi_base_attr car_attr;  
    xmedia_svp_multi_base_attr person_attr;  
    xmedia_svp_multi_base_attr nmv_attr;  
    xmedia_svp_multi_base_attr face_attr;  
    xmedia_svp_multi_base_attr pet_attr;  
    xmedia_svp_multi_base_attr pkg_attr;  
    xmedia_bool bytetrack_enable;  
    xmedia_bool motionless_filter_enable;  
    xmedia_float stillness_thres;  
    xmedia_u8 movement_fps_thres;  
} xmedia_svp_multi_detect_attr;
```

【成员】

成员名称	描述
car_attr	车辆检测相关属性。
person_attr	人形检测相关属性。
nmv_attr	非机动车检测相关属性。
face_attr	人脸检测相关属性。
pet_attr	宠物检测相关属性。
pkg_attr	包裹检测相关属性。
bytetrack_enable	追踪使能开关。
motionless_filter_enable	运动状态检测开关。
stillness_thres	由运动转为静止状态的灵敏度阈值。
movement_fps_thres	由运动转为静止状态的连续帧阈。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_dms_attr

【说明】

定义 DMS 检测任务属性结构体。

【定义】

```
typedef struct {  
    xmedia_svp_yolov8_attr detect_attr;  
    xmedia_u8 warn_frame;  
    xmedia_float iou_thres;  
} xmedia_svp_dms_attr;
```

【成员】

成员名称	描述
detect_attr	yolov8 系检测相关属性。
warn_frame	警告帧数。
iou_thres	相交阈值。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_dms_state

【说明】

定义 DMS 状态属性结构体。

【定义】

```
typedef struct {  
    xmedia_bool flag;  
    xmedia_float detect_score;  
    xmedia_svp_rect rect;  
} xmedia_svp_dms_state;
```

【成员】

成员名称	描述
flag	状态标志位。

成员名称	描述
detect_score	检测模型结果置信度。
rect	检测结果目标框。

【相关数据类型及接口】

[xmedia_svp_dms_report](#)

xmedia_svp_dms_report

【说明】

定义 DMS 状态结果上报结构体。

【定义】

```
typedef struct {
    xmedia_bool fatigue_flag;
    xmedia_u8 phone_num;
    xmedia\_svp\_dms\_state phone[XMEDIA_SVP_DMS_SPECIAL_MAX_NUM];
    xmedia_u8 cigar_num;
    xmedia\_svp\_dms\_state cigar[XMEDIA_SVP_DMS_SPECIAL_MAX_NUM];
} xmedia_svp_dms_report;
```

【成员】

成员名称	描述
fatigue_flag	疲劳状态标志位。
phone_num	手机检测数量，最大 3。
phone [XMEDIA_SVP_DMS_SPECIAL_MAX_NUM]	手机状态。
cigar_num	香烟检测数量，最大 3。
cigar [XMEDIA_SVP_DMS_SPECIAL_MAX_NUM]	香烟状态。

【相关数据类型及接口】

[xmedia_svp_dms_result](#)

xmedia_svp_dms_result

【说明】

定义 DMS 结果结构体。

【定义】

```
typedef struct {  
    xmedia_svp_alg_type alg_type;  
    xmedia_svp_class_type class_type;  
    xmedia_float detect_score;  
    xmedia_s32 tracker_id;  
    xmedia_u32 tracker_age;  
    xmedia_svp_rect rect;  
    xmedia_bool special_target;  
    xmedia_svp_keypoint kpt[MAX_DETECT_KEYPOINT_NUM]; // 关键点坐标  
    xmedia_svp_dms_report dms_report;  
} xmedia_svp_dms_result;
```

【成员】

成员名称	描述
alg_type	算法类别。
class_type	检测结果类别。
detect_score	检测模型结果置信度。
tracker_id	检测结果追踪 id。
tracker_age	检测结果追踪 age。
rect	检测结果目标框。
special_target	特殊目标。
kpt[MAX_DETECT_KEYPOINT_NUM]	关键点坐标。
dms_report	DMS 状态结果。

【相关数据类型及接口】

xmedia_svp_dms_output

xmedia_svp_dms_output

【说明】

定义 DMS 任务输出信息。

【定义】

```
typedef struct {  
    xmedia_u32 target_num;  
    xmedia_svp_dms_result targets[XMEDIA_SVP_DMS_MAX_TARGET_NUM];  
} xmedia_svp_dms_output;
```

【成员】

成员名称	描述
target_num	检测结果数量。
targets[XMEDIA_SVP_DMS_MAX_TARGET_NUM]	DMS 检测目标结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

[xmedia_svp_vec2i](#)

【说明】

定义单点万分比坐标结构体。

【定义】

```
typedef struct {  
    xmedia_s32 x;  
    xmedia_s32 y;  
} xmedia_svp_vec2i;
```

【成员】

成员名称	描述
x	X 坐标，采用万分比坐标。
y	Y 坐标，采用万分比坐标。

[xmedia_svp_landmarks](#)

【说明】

定义坐标点参数信息。

【定义】

```
typedef struct {  
    xmedia_s32 points_count;  
    xmedia_svp_vec2i points_array[XMEDIA_SVP_MAX_EDGE];  
} xmedia_svp_landmarks;
```

【成员】

成员名称	描述
points_count	坐标点数量。
points_array[XMEDIA_SVP_MAX_EDGE]	坐标点数组。

【相关数据类型及接口】

[xmedia_svp_zone](#)

xmedia_svp_zone

【说明】

定义区域参数信息。

【定义】

```
typedef struct {  
    xmedia_s32 zone_num;  
    xmedia_svp_landmarks zones[XMEDIA_SVP_MAX_REGION];  
} xmedia_svp_zone;
```

【成员】

成员名称	描述
zone_num	区域个数。
zones[XMEDIA_SVP_MAX_REGION]	区域数组。

【相关数据类型及接口】

- [xmedia_svp_plate_reco_attr](#)
- [xmedia_svp_vehicle_reco_attr](#)
- [xmedia_svp_plate_vehicle_attr](#)

xmedia_svp_plate_reco_attr

【说明】

定义车牌识别任务属性结构体。

【定义】

```
typedef struct {  
    xmedia_svp_yolov5_attr plate_detect_attr;  
    xmedia_u32              plate_min_width;  
    xmedia_u32              report_count;  
    xmedia_float            report_thres;  
    xmedia_svp_zone         detect_zone;  
} xmedia_svp_plate_reco_attr;
```

【成员】

成员名称	描述
plate_detect_attr	Yolov5 系检测相关属性。
plate_min_width	最小车牌宽度。
report_count	上报计数阈值。
report_thres	上报分数阈值。
detect_zone	检测区域。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_plate_result

【说明】

定义车牌识别任务结果结构体。

【定义】

```
typedef struct {  
    xmedia_s32 tracker_id;  
    xmedia_char plate_char[XMEDIA_SVP_PLATE_CHAR_MAX_NUM];  
    xmedia_float char_score;  
    xmedia_svp_plate_color color;  
    xmedia_float color_score;
```

```

xmedia_svp_rect rect;
xmedia_svp_direction lpr_direction;
} xmedia_svp_plate_result;

```

【成员】

成员名称	描述
tracker_id	检测结果追踪 id。
plate_char[XMEDIA_SVP_PLATE_CHAR_MAX_NUM]	车牌字符数组。
char_score	车牌字符置信度。
color	车牌颜色。
color_score	车牌颜色置信度。
rect	检测结果目标框。
lpr_direction	车牌触发方向。

【相关数据类型及接口】

xmedia_svp_plate_reco_output

xmedia_svp_plate_reco_output

【说明】

定义车牌识别任务输出信息。

【定义】

```

typedef struct {
    xmedia_s32 target_num;
    xmedia_svp_plate_result targets[XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM];
} xmedia_svp_plate_reco_output;

```

【成员】

成员名称	描述
target_num	车牌结果数量。
targets[XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM]	车牌目标结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

[xmedia_svp_vehicle_reco_attr](#)

【说明】

定义车辆识别属性结构体。

【定义】

```
typedef struct {  
    xmedia\_svp\_yolov5\_attr plate_detect_attr;  
    xmedia_u32 report_count;  
    xmedia_float report_thres;  
    xmedia\_svp\_zone detect_zone;  
} xmedia_svp_vehicle_reco_attr;
```

【成员】

成员名称	描述
plate_detect_attr	Yolov5 系检测相关属性。
report_count	上报计数阈值。
report_thres	上报分数阈值。
detect_zone	检测区域。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

[xmedia_svp_vehicle_result](#)

【说明】

定义车辆识别任务结果结构体。

【定义】

```
typedef struct {  
    xmedia_s32 tracker_id;  
    xmedia\_svp\_vehicle\_color color;  
    xmedia\_svp\_vehicle\_type type;  
    xmedia_float color_scr;  
    xmedia_float type_scr;  
    xmedia\_svp\_rect rect;
```

```
xmedia_svp_direction vehicle_direction;
} xmedia_svp_vehicle_result;
```

【成员】

成员名称	描述
tracker_id	检测结果追踪 id。
color	车辆颜色。
type	车辆类型。
color_scr	车辆颜色置信度。
type_scr	车辆类型置信度。
rect	检测结果目标框。
vehicle_direction	车辆触发方向。

【相关数据类型及接口】

[xmedia_svp_vehicle_reco_output](#)

xmedia_svp_vehicle_reco_output

【说明】

定义车辆识别任务输出信息。

【定义】

```
typedef struct {
    xmedia_s32 target_num;
    xmedia_svp_vehicle_result targets[XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM];
} xmedia_svp_vehicle_reco_output;
```

【成员】

成员名称	描述
target_num	车辆结果数量。
targets[XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM]	车辆目标结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)

xmedia_svp_plate_vehicle_attr

【说明】

定义车牌车辆识别属性结构体。

【定义】

```
typedef struct {  
    xmedia_svp_yolov5_attr plate_attr;  
    xmedia_svp_yolov5_attr veh_attr;  
    xmedia_u32                plate_min_width;  
    xmedia_u32                report_count;  
    xmedia_float              report_thres;  
    xmedia_svp_zone           detect_zone;  
} xmedia_svp_plate_vehicle_attr;
```

【成员】

成员名称	描述
plate_attr	Yolov5 系检测相关属性。
veh_attr	Yolov5 系检测相关属性。
plate_min_width	最小车牌宽度。
report_count	上报计数阈值。
report_thres	上报分数阈值。
detect_zone	检测区域。

【相关数据类型及接口】

[xmedia_svp_task_set_attr](#)

xmedia_svp_plate_veh_result

【说明】

定义车牌车辆识别任务结果结构体。

【定义】

```
typedef struct {  
    xmedia_s32 tracker_id;  
    xmedia_svp_vehicle_color veh_color;  
    xmedia_svp_vehicle_type veh_type;
```



```

xmedia_float veh_color_scr;
xmedia_float veh_type_scr;
xmedia_svp_rect veh_rect;
xmedia_svp_direction veh_direction;
xmedia_char plate_char[XMEDIA_SVP_PLATE_CHAR_MAX_NUM];
xmedia_float plate_score;
xmedia_svp_plate_color plate_color;
xmedia_float plate_color_scr;
xmedia_svp_rect plate_rect;
xmedia_svp_direction plate_direction;
} xmedia_svp_plate_veh_result;

```

【成员】

成员名称	描述
tracker_id	检测结果追踪 id。
veh_color	车辆颜色。
veh_type	车辆类型。
veh_color_scr	车辆颜色置信度。
veh_type_scr	车辆类型置信度。
veh_rect	车辆坐标。
veh_direction	车牌触发方向。
plate_char[XMEDIA_SVP_PLATE_CHAR_MAX_NUM]	车牌字符数组。
plate_score	车牌字符置信度。
plate_color	车牌颜色。
plate_color_scr	车牌颜色置信度。
plate_rect	车牌坐标。
plate_direction	车牌触发方向。

【相关数据类型及接口】

xmedia_svp_plate_vehicle_output

xmedia_svp_plate_vehicle_output

【说明】

定义车牌车辆识别任务输出信息。

【定义】

```
typedef struct {  
    xmedia_s32 target_num;  
    xmedia_svp_plate_veh_result targets[XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM];  
} xmedia_svp_plate_vehicle_output;
```

【成员】

成员名称	描述
target_num	车牌车辆结果数量。
targets[XMEDIA_SVP_PLATE_VEH_MAX_TARGET_NUM]	车牌车辆目标识别结果。

【相关数据类型及接口】

[xmedia_svp_task_process](#)