

T/BAX

团 体 标 准

T/BAX 0006.3—2023

视频图像感知智能应用适配技术要求 第3部分：感知前端和边缘解析设备适配

Technical requirements for intelligent application adaptation of video
and image sensing—Part 3: Video and image sensing intelligent terminal
device and edge analysis device adaptation

2023 - 11 - 02 发布

2024 - 01 - 01 实施

北京安全防范行业协会 发布

目 次

前言 III

引言 IV

1 范围 1

2 规范性引用文件 1

3 术语和定义 1

4 基本要求 1

 4.1 感知前端 1

 4.2 边缘解析设备 1

5 功能要求 2

 5.1 感知前端 2

 5.2 边缘解析设备 2

6 接口要求 3

 6.1 设备接入与信息获取接口 3

 6.2 算法封装文件管理接口 5

 6.3 算法包和算法引擎管理接口 8

 6.4 任务管理接口 11

 6.5 视频图像接入接口 15

附录 A（规范性）设备接入与信息获取 16

 A.1 查询感知前端状态与资源 16

 A.2 查询感知前端授权配额使用量 17

 A.3 查询边缘解析设备状态与资源 18

 A.4 查询边缘解析设备授权配额使用量 18

附录 B（规范性）算法封装文件管理接口 20

 B.1 感知前端算法封装文件下发 20

 B.2 感知前端算法封装文件删除 21

 B.3 感知前端算法封装文件列表 21

 B.4 感知前端算法封装文件详情 23

 B.5 感知前端算法封装文件内可执行文件下发 26

 B.6 边缘解析设备算法封装文件下发 27

 B.7 边缘解析设备算法封装文件删除 27

 B.8 边缘解析设备算法封装文件列表 28

 B.9 边缘解析设备算法封装文件详情 28

 B.10 边缘解析设备算法封装文件内可执行文件下发 30

附录 C（规范性）算法包和算法引擎管理 31

C. 1 感知前端算法包与算法引擎加载	31
C. 2 感知前端算法包与算法引擎卸载	31
C. 3 感知前端算法包与算法引擎列表查询	31
C. 4 感知前端算法包与算法引擎详情查询	33
C. 5 边缘解析设备算法包与算法引擎加载	34
C. 6 边缘解析设备算法包与算法引擎卸载	34
C. 7 边缘解析设备算法包与算法引擎列表查询	35
C. 8 边缘解析设备算法包与算法引擎详情查询	35
附录 D（规范性）任务管理	37
D. 1 感知前端任务创建	37
D. 2 感知前端任务删除	41
D. 3 感知前端任务启动	41
D. 4 感知前端任务停止	41
D. 5 感知前端任务查询列表	42
D. 6 感知前端任务查询详情	44
D. 7 边缘解析设备任务创建	46
D. 8 边缘解析设备任务删除	48
D. 9 边缘解析设备任务启动	48
D. 10 边缘解析设备任务停止	49
D. 11 边缘解析设备任务查询列表	49
D. 12 边缘解析设备任务查询详情	50

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。

本文件是T/BAX 0006《视频图像感知智能应用适配技术要求》的第3部分。T/BAX 0006已经发布了以下部分：

- 第1部分：总体要求；
- 第2部分：算法适配；
- 第3部分：感知前端和边缘解析设备适配；
- 第4部分：中心解析设施适配；
- 第5部分：算法算力服务管理平台。

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。

本文件由北京安全防范行业协会提出并归口。

本文件起草单位：北京市公安局、北京市经济和信息化局、视频图像信息智能分析与共享应用技术国家工程实验室、北京蓝色星际科技股份有限公司、北京安信适配技术有限公司、北京市大数据中心、北京旷视科技有限公司、北京百度网讯科技有限公司、富盛科技股份有限公司、北京密码云芯科技有限公司、北京瑞莱智慧科技有限公司。

本文件主要起草人：刘翀、肖笛、崔云红、步飞、宋杉、赵惠芳、王建勇、赵朋、梁玉晨、陈桂红、陈岩峰、马芳、李西田、张伟、母成、欧阳晓智、李标、钟永强、麻思、孟卿卿、张家吏、孙马秋、张浩天、张飞跃、王俊泽。



引 言

视频图像感知智能应用是国家数字化进程和智慧城市建设中的重要内容，在公共安全管理、社会治理、社区管理等应用场景中发挥关键作用。

视频感知体系智能化应用适配中心围绕提升智能应用、安全可信、自主可控的核心目标开展适配工作，制定智能提升技术要求、安全可信技术要求、自主可控技术要求三大类标准，指导企业研制产品和适配应用推广。智能应用提升技术要求围绕“软硬解耦、算法动态加载、算力智能调度”的需求展开。

T/BAX 0006《视频图像感知智能应用适配技术要求》通过建立统一的适配技术规范，指导视频图像解析算法与设备及平台的适配，解决算法与设备及平台的紧耦合问题，随业务需求变化实现算法快速更新迭代和部署应用及云边端算力动态调度。

T/BAX 0006由五部分构成。

- 第1部分：总体要求。目的在于确定视频图像感知智能应用适配的总体结构、总体要求和适配要求。
- 第2部分：算法适配。目的在于规范视频图像感知智能应用适配的算法适配基本要求、功能要求、授权交付物要求和接口要求。
- 第3部分：感知前端和边缘解析设备适配。目的在于规范视频图像感知智能应用适配的视频图像智能感知前端和智能边缘解析设备适配的基本要求、功能要求、接口要求等。
- 第4部分：中心解析设施适配。目的在于规范视频图像感知智能应用适配的中心解析设施适配的功能组成，中心解析设备、中心计算平台、中心解析平台等的功能要求和接口要求，以及输入与输出和接口安全要求等。
- 第5部分：算法算力服务管理平台。目的在于规范视频图像感知智能应用适配的算法算力服务管理平台的功能组成、算法接入、算法管理、算力接入、算力管理、视图接入管理要求等技术要求。

视频图像感知智能应用适配技术要求

第3部分：感知前端和边缘解析设备适配

1 范围

本文件规定了视频图像感知智能应用适配的视频图像智能感知前端（以下简称“感知前端”）和边缘解析设备适配的基本要求、功能要求、接口要求等。

本文件适用于感知前端和边缘解析设备（以下简称“端边设备”）适配的设计、研发、检测和验收。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

- GB/T 28181—2022 公共安全视频监控联网系统信息传输、交换、控制技术要求
- GA/T 1400.1—2017 公安视频图像信息应用系统 第1部分：通用技术要求
- GA/T 1400.4—2017 公安视频图像信息应用系统 第4部分：接口协议要求
- T/BAX 0006.1—2023 视频图像感知智能应用技术要求 第1部分：总体要求
- T/BAX 0006.4—2023 视频图像感知智能应用技术要求 第4部分：中心解析设施适配
- T/BAX 0006.5—2023 视频图像感知智能应用技术要求 第5部分：算法算力服务管理平台

3 术语和定义

GB/T 28181—2022、GA/T 1400.1—2017、T/BAX 0006.1—2023界定的术语和定义适用于本文件。

4 基本要求

4.1 感知前端

感知前端符合以下基本要求：

- a) 应提供加载运行视频图像智能解析算法的基础运行环境；
- b) 计算能力应大于等于 0.5Tops，满足对所采集视频图像进行最小算法模型实时智能解析的要求；
- c) 存储容量应大于等于 512MB，满足对所采集视频图像进行最小算法模型实时智能解析的要求；
- d) 在满足 GB/T 28181—2022 中 5.5 网络传输质量要求的条件下，应能实现视频图像智能解析算法稳定下载；
- e) 应能支持视频图像智能解析算法的长期稳定运行；
- f) 加载视频图像智能解析算法不应影响与算法无关的其他功能（如视频图像的采集等）。

4.2 边缘解析设备

边缘解析设备符合以下基本要求：

- a) 计算能力应满足大于等于 2Tops，满足对所接入视频图像至少两路进行最小算法模型进行实时智能解析的要求；
- b) 存储容量应满足大于等于 2GB，满足对所接入视频图像至少两路进行最小算法模型进行实时智能解析的要求；
- c) 在满足 GB/T 28181—2022 中 5.5 网络传输质量要求的条件下，应能实现视频图像智能解析算法稳定下载；
- d) 应能支持视频图像智能解析算法的长期稳定运行；
- e) 加载视频图像智能解析算法不应影响与算法无关的其他功能；
- f) 宜采用容器化模式加载运行视频图像智能解析算法。

5 功能要求

5.1 感知前端

感知前端符合以下功能要求：

- a) 应支持视频图像信息的采集；
- b) 应支持对采集的视频图像进行智能解析；
- c) 应支持视频图像智能解析算法的管理，包括算法封装文件的导入、删除、查询等；
- d) 应支持视频图像智能解析算法的动态加载、卸载、运行状态查询；
- e) 应支持对解析任务的管理，包括任务的创建、删除、查询、启动、停止等；
- f) 应支持算法算力服务管理平台的统一管理调度；
- g) 应支持通过符合 GA/T 1400.4 规定的接口协议输出视频图像信息及智能解析结果；
- h) 宜支持视频图像信息和智能解析结果的本地存储。

5.2 边缘解析设备

边缘解析设备应符合以下功能要求：

- a) 支持多路视频图像采集设备的接入；
- b) 支持对接入的至少两路视频图像进行智能解析；
- c) 支持视频图像智能解析算法的管理，包括算法封装文件的导入、删除、查询等；
- d) 支持视频图像智能解析算法的动态加载、卸载、运行状态查询；
- e) 支持对解析任务的管理，包括任务的创建、删除、查询、启动、停止等；
- f) 支持对计算资源的查询，包括总计算资源、当前剩余计算资源等；
- g) 支持算法算力服务管理平台的统一管理调度；
- h) 支持通过符合 GA/T 1400.4 规定的接口协议输出视频图像信息及智能解析结果；
- i) 支持视频图像信息和智能解析结果的本地存储。

6 接口要求

6.1 设备接入与信息获取接口

6.1.1 接口调用流程

端边设备与算法算力服务管理平台（以下简称“算法算力平台”）之间设备接入与信息获取接口的调用流程见图1，此部分接口应在向算法算力平台注册后，才可被算法算力平台调用。

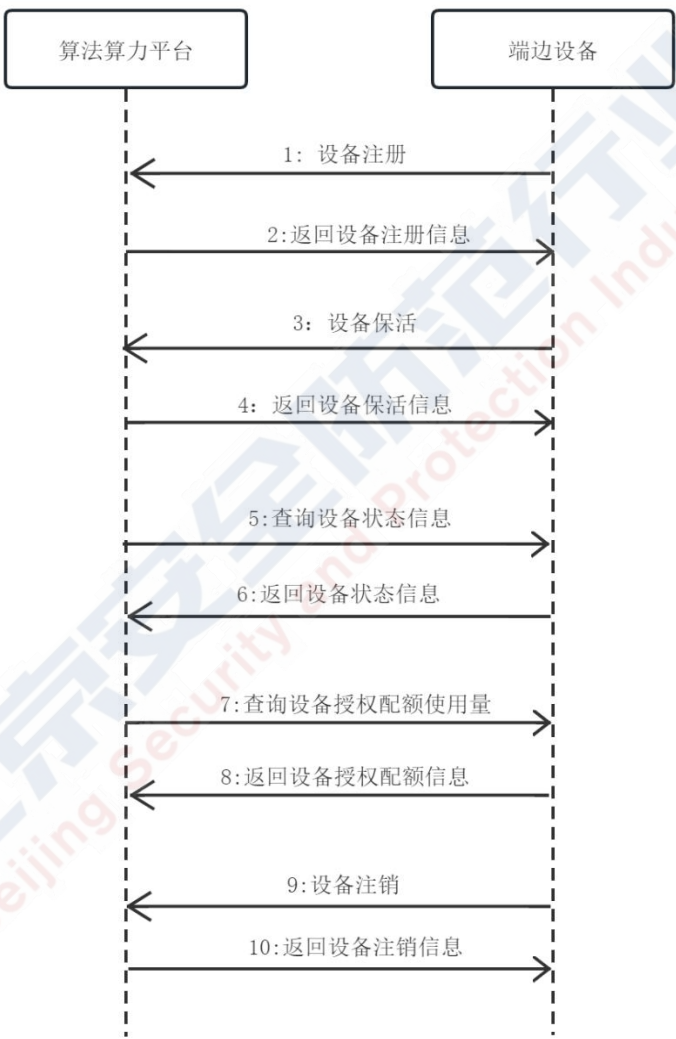


图 1 设备接入与信息获取交互流程

6.1.2 感知前端

感知前端为算法算力平台提供设备接入和信息查询接口，包括设备的注册、注销、保活、状态信息查询、授权配额量查询，其中注册、注销和保活接口应符合T/BAX 0006.5—2023中附录C的规定；状态信息查询和授权配额量查询接口应符合表1～表3的规定，接口消息对象属性应符合A.1和A.2的规定。

表 1 感知前端接入接口

序号	URI	方法	功能说明	调用方	提供方
1	/VITD/Status	GET	状态及资源情况查询	算法算力平台	感知前端
2	/VITD/Quota/<AlgoFilePackageID>	GET	授权配额量查询	算法算力平台	感知前端

表 2 感知前端状态查询

URI	/VITD/Status		
功能	感知前端状态查询		
方法	查询字符串	请求消息体	响应消息体
GET	无	无	DeviceStatusResponse
注释	查询感知前端状态与资源		

表 3 感知前端授权配额使用量查询

URI	/VITD/Quota/<AlgoFilePackageID>		
功能	查询前端感知设备授权配额使用量		
方法	查询字符串	请求消息体	响应消息体
GET	算法封装文件ID	无	DeviceQuotaDetailResponse
注释	查询感知前端授权配额量，供平台查询指定算法包或算法引擎的已使用配额和总配额		

6.1.3 边缘解析设备

边缘解析设备为算法算力平台提供设备接入和信息查询接口，包括设备的注册、注销、保活、状态信息查询、授权配额量查询，其中注册、注销和保活接口应符合T/BAX 0006.5—2023中附录C的规定；状态信息查询和授权配额量查询接口应符合表4～表6的规定，接口对象详细属性应符合附录A.3～A.4的规定。

表 4 边缘解析设备接入接口

序号	URI	方法	功能说明	调用方	提供方
1	/EAD/Status	GET	状态及资源情况查询	算法算力平台	边缘解析设备
2	/EAD/Quota/<AlgoFilePackageID>	GET	授权配额量查询	算法算力平台	边缘解析设备

表 5 边缘解析设备状态查询

URI	/EAD/Status		
功能	边缘解析设备状态查询		
方法	查询字符串	请求消息体	响应消息体
GET	无	无	DeviceStatusResponse
注释	查询边缘解析设备状态与资源		

表 6 边缘解析设备授权配额使用量查询

URI	/EAD/Quota/<AlgoFilePackageID>			
功能	查询边缘解析设备授权配额使用量			
方法	查询字符串	请求消息体	响应消息体	

表 6 边缘解析设备授权配额使用量查询（续）

GET	算法封装文件ID	无	DeviceQuotaDetailResponse
注释	查询边缘解析设备授权配额量，供平台查询指定算法包或算法引擎的已使用配额和总配额		

6.2 算法封装文件管理接口

6.2.1 接口调用流程

端边设备与算法算力平台之间算法封装文件的调用流程见图2，此部分接口应在向算法算力平台建立注册后，才可被算法算力平台调用。

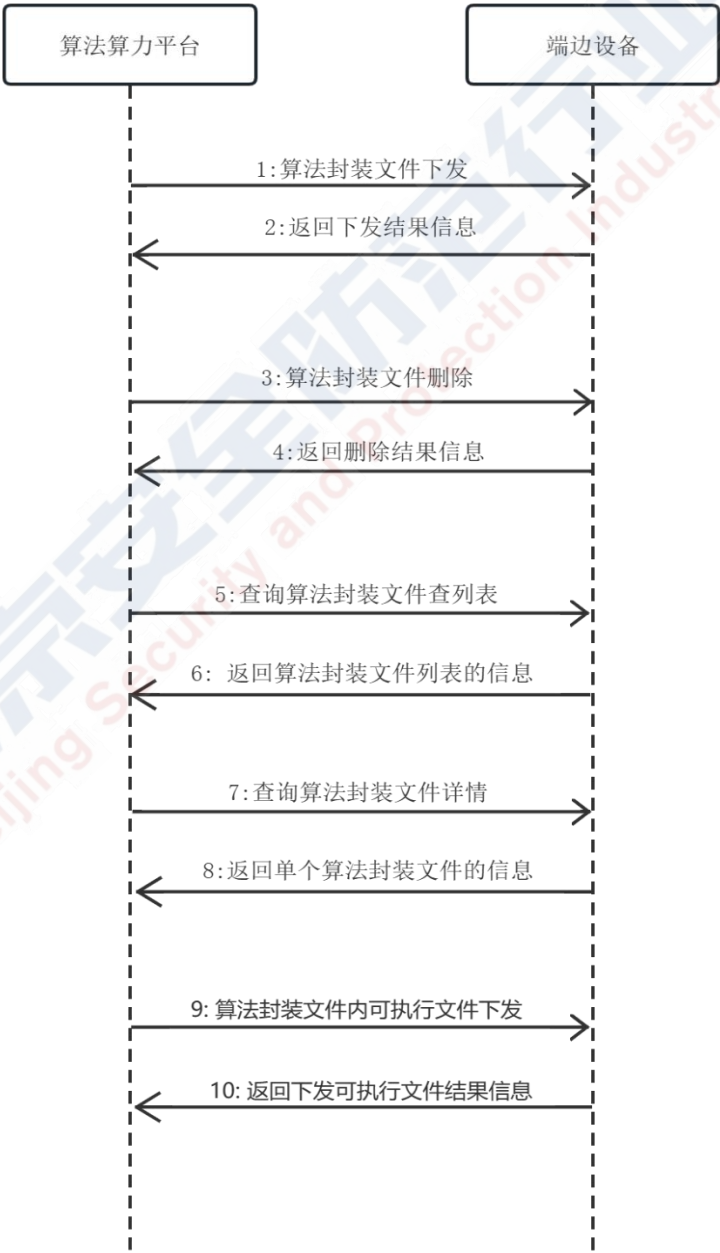


图 2 算法管理交互流程

6.2.2 感知前端

感知前端对外提供的算法封装文件管理接口应符合表7～表12的规定。其中，接口对象属性信息应符合附录B.1～B.5的规定。

表7 感知前端算法管理接口

序号	URI	方法	功能说明	调用方	提供方
1	/VITD/AlgoFilePackages	POST	算法封装文件下发	算法算力平台	感知前端
2	/VITD/AlgoFilePackages/<AlgoFilePackageID>	DELETE	算法封装文件删除	算法算力平台	感知前端
3	/VITD/AlgoFilePackages:search	POST	算法封装文件列表查询	算法算力平台	感知前端
4	/VITD/AlgoFilePackages/<AlgoFilePackageID>	GET	算法封装文件详情查询	算法算力平台	感知前端
5	/VITD/AlgoFilePackages:pushExecFile	POST	算法封装文件内可执行文件 下发	算法算力平台	感知前端

表8 算法封装文件下发

URI	/VITD/AlgoFilePackages		
功能	算法封装文件下发		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoFilePackagePushRequest	AlgoFilePackagePushResponse
注释	下发的封装文件可根据封装文件的大小决定是否包含可执行文件，当可执行文件过大时可通过附录B.5的下发可执行文件接口完成对可执行文件的单独下发，以确保下发成功率。 可执行文件说明详见：T/BAX 0006.2—2023中附录A.1 yaml说明。		

表9 算法封装文件删除

URI	/VITD/AlgoFilePackages/<AlgoFilePackageID>		
功能	算法封装文件删除		
方法	查询字符串	请求消息体	响应消息体
DELETE	算法封装文件ID	无	EmptyResponse
注释	删除算法封装文件		

表10 算法封装文件列表查询

URI	/VITD/AlgoFilePackages:search		
功能	查询算法封装文件列表		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoFilePackageSearchRequest	AlgoFilePackageSearchResponse
注释	查询算法封装文件列表		

表 11 算法封装文件详情查询

URI	/VITD/AlgoFilePackages/<AlgoFilePackageID>		
功能	查询算法封装文件详情		
方法	查询字符串	请求消息体	响应消息体
GET	算法封装文件ID	无	AlgoFilePackageDetailResponse
注释	查询算法封装文件详情		

表 12 算法封装文件内可执行文件下发

URI	/VITD/AlgoFilePackages:pushExecFile		
功能	算法封装文件内可执行文件下发		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoPackagePushRequest	AlgoPackagePushResponse
注释	算法封装文件内可执行文件下发，作为可选步骤，补充算法封装文件里的可能不包含的可执行文件。此处的文件需要和对应的算法封装文件algo.yaml内部定义一致，否则可能无法正常运行。		

6.2.3 边缘解析设备

边缘解析设备对外提供的算法封装文件管理接口应符合表13～表18的规定。其中，接口对象属性信息应符合附录B.6～B.10的规定

表 13 边缘解析设备算法管理接口

序号	URI	方法	功能说明	调用方	提供方
1	/EAD/AlgoFilePackages	POST	算法封装文件下发	算法算力平台	边缘解析设备
2	/EAD/AlgoFilePackages/<AlgoFilePackageID>	DELETE	算法封装文件删除	算法算力平台	边缘解析设备
3	/EAD/AlgoFilePackages:search	POST	算法封装文件列表查询	算法算力平台	边缘解析设备
4	/EAD/AlgoFilePackages/<AlgoFilePackageID>	GET	算法封装文件详情查询	算法算力平台	边缘解析设备
5	/EAD/AlgoFilePackages:pushExecFile	POST	算法封装文件内可执行文件下发	算法算力平台	边缘解析设备

表 14 算法封装文件下发

URI	/EAD/AlgoFilePackages		
功能	算法封装文件下发		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoFilePackagePushRequest	AlgoFilePackagePushResponse
注释	下发的封装文件可根据封装文件的大小决定是否包含可执行文件，当可执行文件过大时可通过通过附录B.10的下发可执行文件接口完成对可执行文件的单独下发，以确保下发成功率。 可执行文件说明详见：T/BAX 0006.2—2023中附录A.1 yaml说明。		

表 15 算法封装文件删除

URI	/EAD/AlgoFilePackages/<AlgoFilePackageID>		
功能	算法封装文件删除		
方法	查询字符串	请求消息体	响应消息体
DELETE	算法封装文件ID	无	EmptyResponse
注释	删除算法封装文件		

表 16 算法封装文件列表查询

URI	/EAD/AlgoFilePackages:search		
功能	查询算法封装文件列表		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoFilePackageSearchRequest	AlgoFilePackageSearchResponse
注释	查询算法封装文件列表		

表 17 算法封装文件详情查询

URI	/EAD/AlgoFilePackages/<AlgoFilePackageID>		
功能	查询算法封装文件详情		
方法	查询字符串	请求消息体	响应消息体
GET	算法封装文件ID	无	AlgoFilePackageDetailResponse
注释	查询算法封装文件详情		

表 18 算法封装文件内可执行文件下发

URI	/EAD/AlgoFilePackages:pushExecFile		
功能	算法封装文件内可执行文件下发		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoPackagePushRequest	AlgoPackagePushResponse
注释	算法封装文件内可执行文件下发，作为可选步骤，补充算法封装文件里的可能不包含的可执行文件。此处的文件需要和对应的算法封装文件algo.yaml内部定义一致，否则可能无法正常运行。		

6.3 算法包和算法引擎管理接口

6.3.1 接口调用流程

端边设备与算法算力平台之间算法包和算法引擎管理接口调用流程见图3，此部分接口应在向算法算力平台建立注册后，才可被算法算力平台调用。

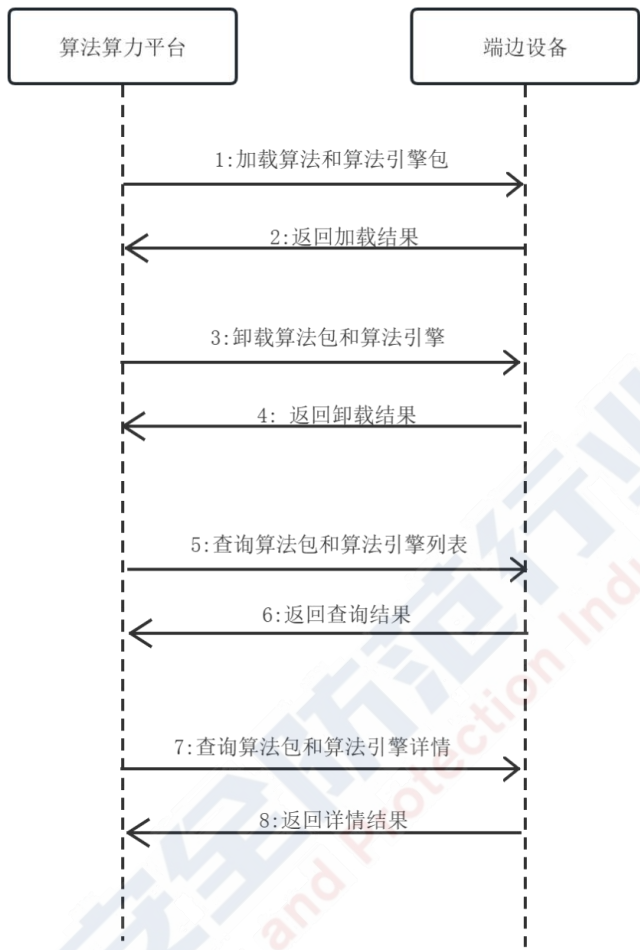


图 3 算法包和算法引擎管理接口调用流程

6.3.2 感知前端

感知前端的算法包和算法引擎管理接口主要包括算法包和算法引擎的加载、卸载、算法包和算法引擎列表查询和算法详情查询接口应符合表19～表23的规定，各个接口属性信息应符合附录C.1～C.4的规定。

表 19 感知前端服务管理接口

序号	URI	方法	功能说明	调用方	提供方
1	/VITD/Machine/AlgoPackages/<AlgoFilePackageID>	GET	加载算法包和算法引擎	算法算力平台	感知前端
2	/VITD/Machine/AlgoPackages/<AlgoFilePackageID>	DELETE	卸载算法包和算法引擎	算法算力平台	感知前端
3	/VITD/Machine/AlgoPackages	POST	查询算法包和算法引擎列表	算法算力平台	感知前端
4	/VITD/Machine/AlgoPackages/<AlgoFilePackageID>	GET	查询算法包和算法引擎详情	算法算力平台	感知前端

表 20 加载算法包和算法引擎

URI	/VITD/Machine/AlgoPackages/<AlgoFilePackageID>		
功能	加载算法包和算法引擎		
方法	查询字符串	请求消息体	响应消息体
GET	算法封装文件ID	无	EmptyResponse
注释	根据AlgoFilePackageID加载算法包和算法引擎		

表 21 卸载算法包和算法引擎

URI	/VITD/Machine/AlgoPackages/<AlgoFilePackageID>		
功能	卸载算法包和算法引擎		
方法	查询字符串	请求消息体	响应消息体
DELETE	算法封装文件ID	无	EmptyResponse
注释	卸载算法包和算法引擎		

表 22 算法包和算法引擎列表查询

URI	/VITD/Machine/AlgoPackages		
功能	查询算法包和算法引擎列表		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoPackageListRequest	AlgoPackageListResponse
注释	查询算法包和算法引擎列表		

表 23 算法包和算法引擎详情查询

URI	/VITD/Machine/AlgoPackages/<AlgoFilePackageID>		
功能	查询算法包和算法引擎详情		
方法	查询字符串	请求消息体	响应消息体
GET	算法封装文件ID	无	AlgoPackageDetailResponse
注释	查询算法包和算法引擎详情		

6.3.3 边缘解析设备

边缘解析设备算法包和算法引擎管理接口主要包括算法包和算法引擎的加载、卸载、算法包和算法引擎列表查询和算法详情查询应符合表24～表28的规定。其中，各个接口属性信息应符合附录C.5～C.8的规定。

表 24 边缘解析设备服务管理接口

序号	URI	方法	功能说明	调用方	提供方
1	/EAD/Machine/AlgoPackages/<AlgoFilePackageID>	POST	加载算法包和算法引擎	算法算力平台	边缘解析设备
2	/EAD/Machine/AlgoPackages/<AlgoFilePackageID>	DELETE	卸载算法包和算法引擎	算法算力平台	边缘解析设备

表 24 边缘解析设备服务管理接口（续）

序号	URI	方法	功能说明	调用方	提供方
3	/EAD/Machine/AlgoPackages	POST	查询算法包和算法引擎列表	算法算力平台	边缘解析设备
4	/EAD/Machine/AlgoPackages/<AlgoFilePackageID>	GET	查询算法包和算法引擎详情	算法算力平台	边缘解析设备

表 25 加载算法包和算法引擎

URI	/EAD/Machine/AlgoPackages/<AlgoFilePackageID>		
功能	加载算法包和算法引擎		
方法	查询字符串	请求消息体	响应消息体
POST	算法封装文件ID	无	EmptyResponse
注释	根据AlgoFilePackageID加载算法包和算法引擎		

表 26 卸载算法包和算法引擎

URI	/EAD/Machine/AlgoPackages/<AlgoFilePackageID>		
功能	卸载算法包和算法引擎		
方法	查询字符串	请求消息体	响应消息体
DELETE	算法封装文件ID	无	EmptyResponse
注释	卸载算法包和算法引擎		

表 27 算法包和算法引擎列表查询

URI	/EAD/Machine/AlgoPackages		
功能	查询算法包和算法引擎列表		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoPackageListRequest	AlgoPackageListResponse
注释	查询算法包和算法引擎列表		

表 28 算法包和算法引擎详情查询

URI	/EAD/Machine/AlgoPackages/<AlgoFilePackageID>		
功能	查询算法包和算法引擎详情		
方法	查询字符串	请求消息体	响应消息体
GET	算法封装文件ID	无	AlgoPackageDetailResponse
注释	查询算法包和算法引擎详情		

6.4 任务管理接口

6.4.1 接口调用流程

端边设备与算法算力平台之间任务管理接口调度流程见图4，此部分接口应在向算法算力平台建立注册后，才可被算法算力平台调用。

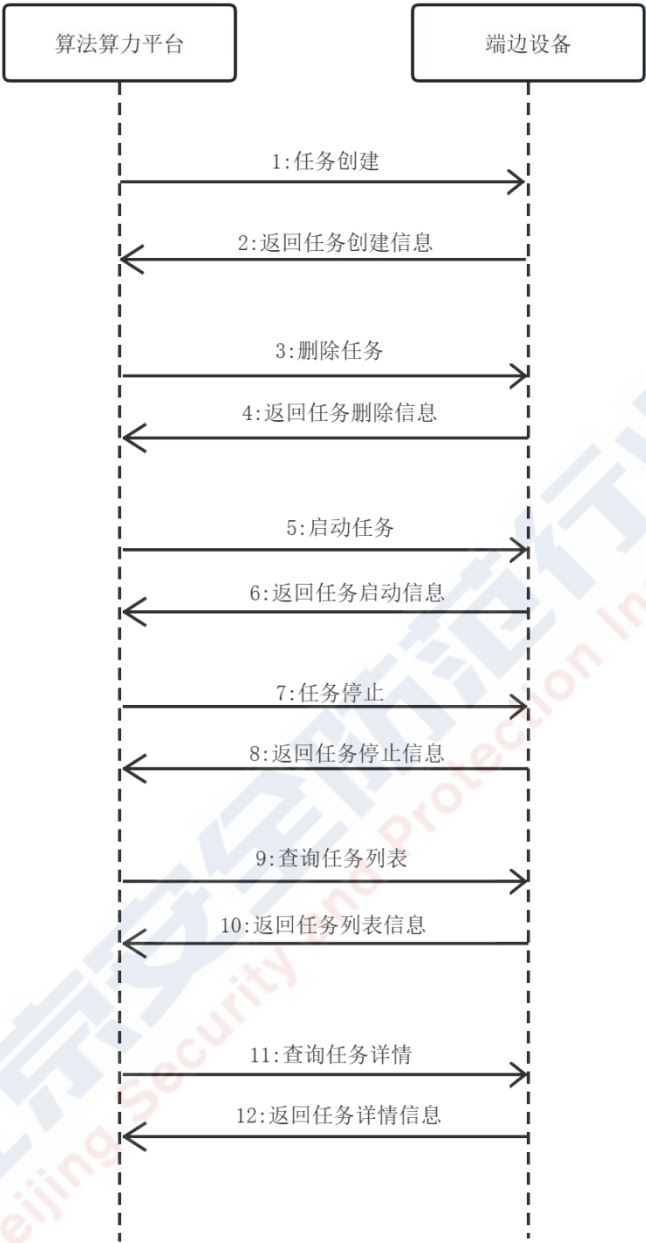


图 4 任务管理接口调用流程

6.4.2 感知前端

感知前端应提供任务的创建、删除、启动、停止、查询等任务管理接口应符合表29～表35的规定。其中，各个接口属性信息应符合附录D.1～D.6的规定。

表 29 感知前端任务管理接口

序号	URI	方法	功能说明	调用方	提供方
1	/VITD/AlgoTasks	POST	任务创建	算法算力平台	感知前端
2	/VITD/AlgoTasks/<ID>	DELETE	任务删除	算法算力平台	感知前端
3	/VITD/AlgoTasks/<ID>:resume	POST	任务启动	算法算力平台	感知前端

表 29 感知前端任务管理接口（续）

序号	URI	方法	功能说明	调用方	提供方
4	/VITD/AlgoTasks/<ID>:stop	POST	任务停止	算法算力平台	感知前端
5	/VITD/AlgoTasks:search	POST	任务查询列表	算法算力平台	感知前端
6	/VITD/AlgoTasks/<ID>	GET	任务查询详情	算法算力平台	感知前端

表 30 任务创建

URI	/VITD/AlgoTasks		
功能	任务创建		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoTaskCreateRequest	AlgoTaskCreateResponse
注释	任务创建，支持在创建时选择启动任务或不启动任务。可与附录A.1、A.2搭配起来使用，达到对任务的灵活控制。同时，如果对应的算法服务没有启动，会自动启动一个匹配的算法服务用来运行此任务。		

表 31 任务删除

URI	/VITD/AlgoTasks/<ID>		
功能	任务删除		
方法	查询字符串	请求消息体	响应消息体
DELETE	任务ID	无	EmptyResponse
注释	任务删除		

表 32 任务启动

URI	/VITD/AlgoTasks/<ID>:resume		
功能	任务启动		
方法	查询字符串	请求消息体	响应消息体
POST	任务ID	无	EmptyResponse
注释	任务启动		

表 33 任务停止

URI	/VITD/AlgoTasks/<ID>:stop		
功能	任务停止		
方法	查询字符串	请求消息体	响应消息体
POST	任务ID	无	EmptyResponse
注释	任务停止		

表 34 任务列表查询

URI	/VITD/AlgoTasks:search				
功能	任务查询列表				
方法	查询字符串	请求消息体		响应消息体	

表 34 任务列表查询（续）

POST	无	AlgoTaskSearchRequest	AlgoTaskSearchResponse
注释	任务查询列表		

表 35 任务详情查询

URI	/VITD/AlgoTasks/<ID>		
功能	任务查询详情		
方法	查询字符串	请求消息体	响应消息体
GET	任务ID	无	AlgoTaskDetailResponse
注释	任务查询详情		

6.4.3 边缘解析设备

边缘解析设备应提供任务的创建、删除、启动、停止、查询等任务管理接口应符合表36～表42的规定。其中，各个接口属性信息应符合附录D.7～D.12的规定。

表 36 边缘解析设备任务管理接口

序号	URI	方法	功能说明	调用方	提供方
1	/EAD/AlgoTasks	POST	任务创建	算法算力平台	边缘解析设备
2	/EAD/AlgoTasks/<ID>	DELETE	任务删除	算法算力平台	边缘解析设备
3	/EAD/AlgoTasks/<ID>:resume	POST	任务启动	算法算力平台	边缘解析设备
4	/EAD/AlgoTasks/<ID>:stop	POST	任务停止	算法算力平台	边缘解析设备
5	/EAD/AlgoTasks:search	POST	任务查询列表	算法算力平台	边缘解析设备
6	/EAD/AlgoTasks/<ID>	GET	任务查询详情	算法算力平台	边缘解析设备

表 37 任务创建

URI	/EAD/AlgoTasks		
功能	任务创建		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoTaskCreateRequest	AlgoTaskCreateResponse
注释	任务创建，支持在创建时选择启动任务或不启动任务。可与附录A.3、A.4搭配起来使用，达到对任务的灵活控制。同时，如果对应的算法服务没有启动，会自动启动一个匹配的算法服务用来运行此任务。		

表 38 任务删除

URI	/EAD/AlgoTasks/<ID>		
功能	任务删除		
方法	查询字符串	请求消息体	响应消息体
DELETE	任务ID	无	EmptyResponse
注释	任务删除		

表 39 任务启动

URI	/EAD/AlgoTasks/<ID>:resume		
功能	任务启动		
方法	查询字符串	请求消息体	响应消息体
POST	任务ID	无	EmptyResponse
注释	任务启动		

表 40 任务停止

URI	/EAD/AlgoTasks/<ID>:stop		
功能	任务停止		
方法	查询字符串	请求消息体	响应消息体
POST	任务ID	无	EmptyResponse
注释	任务停止		

表 41 任务列表查询

URI	/EAD/AlgoTasks:search		
功能	任务查询列表		
方法	查询字符串	请求消息体	响应消息体
POST	无	AlgoTaskSearchRequest	AlgoTaskSearchResponse
注释	任务查询列表		

表 42 任务详情查询

URI	/EAD/AlgoTasks/<ID>		
功能	任务查询详情		
方法	查询字符串	请求消息体	响应消息体
GET	任务ID	无	AlgoTaskDetailResponse
注释	任务查询详情		

6.5 视频图像接入接口

感知前端和边缘解析设备视频流的接入接口应符合GB/T 28181—2022的规定，视频图像的接入接口应符合GA/T 1400.4—2017的规定。

附录 A
(规范性)
设备接入与信息获取

A.1 查询感知前端状态与资源

- a) URI: /VITD/Status
- b) 响应参数

表 A.1 DeviceStatusResponse 属性

序号	名称	标识符	必选/可选	类型	长度	备注
1	结果状态码	Code	R	Integer		0: 成功, 其他: 失败
2	结果描述	Message	R	String	1024	请求结果的描述信息
3	状态信息	Data	0	DeviceInfo		设备状态信息

表 A.2 DeviceInfo 属性

序号	名称	标识符	必选/可选	类型	长度	备注
1	设备ID	DeviceID	R	String	128	设备唯一标识ID
2	在线状态	IsOnline	R	Boolean		True:在线; False:离线
3	系统类型	System	R	String	20	Linux、Android、harmonyos
4	Cpu类型	CpuType	R	String	20	CPU架构, arm、x86
5	Cpu利用率	CpuUtilization	0	Double		cpu 利用率, 0.11 表示 11%
6	Xpu类型	XpuType	R	String	20	Xpu类型, amd、NVIDIA
7	Xpu利用率	XpuUtilization	0	Double		gX
8	内存占用	MemUsage	0	Double		内存占用, 单位 GB
9	内存剩余	MemFree	0	Double		内存剩余, 单位 GB
10	磁盘占用	DiskUsage	0	Double		磁盘占用, 单位 GB
11	磁盘剩余	DiskFree	0	Double		磁盘剩余, 单位 GB

- c) 请求示例: GET /VITD/Status
- d) 响应示例:

```
{
  "Data": {
    "DeviceID": "4028ee368b07b9f0018b139d71d6001e",
    "IsOnline": true,
    "CpuType": "arm",
    "System": "Linux",
    "CpuUtilization": 0.11,
    "XpuType": "amd",
    "XpuUtilization": 0.11,
```

```
    "MemUsage": 24.395,  
    "MemFree": 39.605  
    "DiskUsage": 128.1,  
    "DiskFree": 127.9  
  },  
  "Code": "0",  
  "Message": "success"  
}
```

A.2 查询感知前端授权配额使用量

- a) URI: /VITD/Quota/<AlgoFilePackageID>
- b) 请求参数: AlgoFilePackageID
- c) 响应参数:

表 A.3 DeviceQuotaDetailResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	结果状态码	Code	Integer		R	结果状态码 (0: 成功, 其他: 失败)
2	结果描述	Message	String	1024	R	结果描述
3	指纹数据	Data	DeviceQuota		R	配额信息
4	请求 ID	RequestID	String	128	0	请求 ID

表 A.4 DeviceQuota 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法封装文件 ID	AlgoFilePackageID	String	40	0	算法封装文件ID
2	已使用配额	QuotaUsed	Integer		R	已使用配额, 路数
3	总配额	QuotaTotal	Integer		R	总配额, 路数

- d) 请求示例:
GET /VITD/Quota?AlgoFilePackageID=1101150192120600000120231012203630000101
- e) 响应示例:
{
 "Code": "0",
 "Message": "success",
 "Data": {

```
    "AlgoFilePackageID": "1101150192132600000120231012203630000101",  
    "QuotaUsed": 3,
```

```

    "QuotaTotal": 30
  }
}

```

A.3 查询边缘解析设备状态与资源

- a) URI: /EAD/Status
- b) 响应参数: 见 表 A.1 DeviceStatusResponse 属性信息
- c) 请求体示例: GET /EAD/Status
- d) 响应体示例

```

{
  "Data": {
    "DeviceID": "4028ee368b07b9f0018b139d71d6001e",
    "IsOnline": true,
    "CpuType": "X86",
    "System": "Linux",
    "CpuUtilization": 0.11,
    "XpuType": "amd",
    "XpuUtilization": 0.11,
    "MemUsage": 24.395,
    "MemFree": 39.605,
    "DiskUsage": 128.1,
    "DiskFree": 127.9
  },
  "Code": "0",
  "Message": "success"
}

```

A.4 查询边缘解析设备授权配额使用量

- a) URI: /EAD/Quota/<AlgoFilePackageID>
- b) 请求参数: AlgoFilePackageID
- c) 响应参数: 见 表 A.3 DeviceQuotaDetailResponse 属性信息
- d) 请求示例:
GET /EAD/Quota?AlgoFilePackageID=1101150192132600000120231012203630000101
- e) 响应示例:

```

{
  "Code": "0",
  "Message": "success",
  "Data": {

```



```
        "AlgoFilePackageID":  
        "1101150192132600000120231012203630000101",  
        "QuotaUsed": 3,  
        "QuotaTotal": 30  
    }  
}
```



北京安全防范行业协会
Beijing Security and Protection Industry Association

附 录 B

(规范性)

算法封装文件管理接口

B.1 感知前端算法封装文件下发

a) URI: /VITD/AlgoFilePackages

b) 请求参数

表 B.1 AlgoFilePackagePushRequest 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法封装文件 ID	AlgoFilePackageID	String	40	R	算法封装文件ID
2	文件MD5	MD5	String	32	0	算法封装文件md5校验值
3	下载地址	Url	String	1024	R	算法封装文件下载地址
4	文件名	Name	String	128	0	算法封装文件名
5	鉴权地址	AuthAddress	String	1024	0	鉴权服务器 Ip:Port, 可英文逗号分割多个
6	鉴权文件地址	LicenseFile	String	1024	0	鉴权文件地址

c) 响应参数

表 B.2 AlgoFilePackagePushResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	结果状态码	Code	Integer	-	R	0: 成功, 其他: 失败
2	结果描述	Message	String	1024	R	文字描述
3	算法封装文件 ID	AlgoFilePackageID	String	40	R	算法封装文件 ID
4	请求ID	RequestID	String	128	0	请求ID

d) 请求体示例

{

```
"AlgoFilePackageID":"1101150192508600000120231012203630000101",
"MD5 ":"d577273ff885c3f84dadb8578bb41399",
"Url":"https://xx/xx.tar",
"AuthAddress":"127.0.0.1:8080",
"LicenseFile":"/auth_license.txt"
}
```

e) 响应体示例:

```
{
"AlgoFilePackageID":"1101150192508600000120231012203630000101",
"Code":0,
"Message":"success"
}
```

B.2 知前端算法封装文件删除

- a) URI: /VITD/AlgoFilePackages/<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数

EmptyResponse

- d) 请求体示例:
DELETE /VITD/AlgoFilePackages/1101150192120600000120231012203630000101
- e) 响应体示例:

```
{
"Code":0,
"Message":"success"
}
```

B.3 感知前端算法封装文件列表

- a) URI: /VITD/AlgoFilePackages:search
- b) 请求参数

表 B.3 AlgoFilePackageSearchRequest 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	页码	PageNumber	Integer	-	0	-
2	条数	PageSize	Integer	-	0	-
3	算法封装文件名称	Name	String	128	0	-

c) 响应参数

表 B.4 AlgoFilePackageSearchResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	请求结果状态码	Code	Integer	-	R	0: 成功, 其他: 失败
2	请求结果描述	Message	String	1024	R	-
3	数据	Data	AlgoFilePackage SearchRecord[]	-	R	-
4	请求ID	RequestID	String	128	0	请求ID

表 B.5 AlgoFilePackageSearchRecord 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法封装文件ID	AlgoFilePackageID	String	40	R	算法封装文件ID
2	算法封装文件名称	Name	String	128	R	-
3	版本	Version	String	20	R	-
4	作者	Author	String	20	R	-
5	描述	Description	String	1024	0	-
6	状态	Status	Integer	-	R	0: 待下载 1: 已下载-算法封装 文件, 缺可执行文件 2: 已下载-完整文 件, 含可执行文件 3: 已成功运行 4: 文件异常

d) 请求体示例:

```
{
  "PageNumber":1,"
  PageSize":1
}
```

e) 响应体示例:

```
{
  "Data":{
    "PageNumber":1,
    "PageSize":1,
    "TotalSize":101,"
    Records":[
```

```
{
  "AlgoFilePackageID": "1101150192508600000120231012203630
000101",
  "Name": "xx-headcount_alarm-2.0.0",
  "Version": "2.0.0",
  "Author": "zhangsan",
  "Description": "head count alert algorithm",
  "Status": 1
}
],
"Code": 0,
"Message": "success"
}
```

B.4 感知前端算法封装文件详情

- a) URI: /VITD/AlgoFilePackages/<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数

表 B.6 AlgoFilePackageDetailResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	请求结果状态码	Code	Integer	-	R	0: 成功, 其他: 失败
2	请求结果描述	Message	String	1024	R	-
3	数据	Data	AlgoFilePack age	-	R	-
4	请求 ID	RequestID	String	128	0	请求 ID

表 B.7 AlgoFilePackage 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法封装文件ID	AlgoFilePackageID	String	40	R	-
2	算法封装文件名称	Name	String	128	R	-
3	版本	Version	String	20	R	-
4	作者	Author	String	20	R	-
5	描述	Description	String	1024	0	-

表 B.7 AlgoFilePackage 属性（续）

序号	名称	标识符	类型	长度	必选/可选	备注
6	状态	Status	Integer	-	R	0: 待下载 1: 已下载-算法封装文件, 缺可执行文件 2: 已下载-完整文件, 含可执行文件 3: 已成功运行 4: 文件异常
7	标签组	Tags	String[]	-	0	包的标签描述信息, 可按需添加, 方便查询和筛选。详见: T/BAX 0006.2—2023附录A.1 yaml说明
8	可执行文件	Bin	Object	-	R	内部有如下字段: 类型: Type 远程地址: remoteUri 本地地址: localUri 详见: T/BAX 0006.2—2023中附录A.1 yaml说明。
9	算法运行时	AlgoRuntime	Object	-	R	内部有如下字段: 启动命令: cmd 环境变量: Env Xpu资源: Xpu 详见: T/BAX 0006.2—2023中附录A.1 yaml说明。

d) 请求体示例

GET

/VITD/AlgoFilePackages/?AlgoFilePackageID=1101150192120600000120231012203630000101

e) 响应体示例:

```
{
  "Data": {
    "AlgoFilePackageID": "1101150192508600000120231012203630000101",
    "Name": "xxxx",
    "Version": "3.4.0",
    "Author": "r-face",
    "Description": "xxx",
```

```

    "Tags":[
        "xxx"]
    ,
    "Bin":{
        "Type":"Image",
        "remote_uri":"","
        "local_uri":""
    },
    "algo_runtime":{
        "cmd":"","
        Env":[
            {
                "Name": "_FILELOG_PATH",
                "Value": "/Data/limbo-log"
            },
            {
                "Name": "USE_AIPAAS_PROTO",
                "Value": "1"
            }
        ],
        "Xpu": [
            {
                "Quota": 0.5,
                "Limit": 0,
                "Vendor": "NVIDIA ",
                "Accelerator": "2070",
                "VITDacities": {
                    "Stream_Video_Quota": 6,
                    "Stream_VITDture_Quota": 0,
                    "Analyze_Quota": 0
                },
                "Resolution": "PIXELS_200W"
            }
        ]
    },
    "Code": 0,

```

```
    "Message": "success"
  }
```

B.5 感知前端算法封装文件内可执行文件下发

- a) URI: /VITD/AlgoFilePackages:pushExecFile
- b) 请求参数

表 B.8 AlgoFilePackageExecFilePushRequest 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	可执行文件类型	Type	String	10	R	取值范围如下： Image: 用于算法引擎集成模式，例如服务器算法下发的容器镜像 artifact: 用于算法包集成模式，例如嵌入式算法下发的包含可执行文件的打包文件
2	文件MD5	MD5	String	32	R	-
3	文件名	Name	String	128	0	-
4	下载地址	Url	String	1024	R	-
5	关联算法封装文件ID	AlgoFilePackageID	String	40	R	-

- c) 响应参数

表 B.9 AlgoFilePackageExecFilePushResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	结果状态码	Code	Integer		R	0: 成功，其他: 失败
2	结果描述	Message	String	1024	R	文字描述
3	数据	Data	Object		R	参见示例
4	请求ID	RequestID	String	128	0	请求ID

- e) 请求体示例

```
{
  "Type": "Image",
  "MD5": "d577273ff885c3f84dadb8578bb41620",
```

```
    "Url": "http://127.0.0.1/xxx.tar",
    "AlgoFilePackageID": "1101150192508600000120231012203630000101"
  }
```

e) 响应体示例:

```
{
  "Data": {},
  "Code": 0,
  "Message": "success"
}
```

B.6 边缘解析设备算法封装文件下发

- a) URI: /EAD/AlgoFilePackages
- b) 请求参数: 见表 B.1 AlgoFilePackagePushRequest 属性信息
- c) 响应参数: 见表 B.2 AlgoFilePackagePushResponse 属性信息
- d) 请求体示例

```
{
  "AlgoFilePackageID": "1101150192508600000120231012203630000101",
  "MD5": "d577273ff885c3f84dadb8578bb41399",
  "Url": "https://xx/xx.tar",
  "AuthAddress": "127.0.0.1:8080",
  "LicenseFile": "/auth_license.txt"
}
```

e) 响应体示例:

```
{
  "AlgoFilePackageID": "1101150192508600000120231012203630000101",
  "Code": 0,
  "Message": "success"
}
```

B.7 边缘解析设备算法封装文件删除

- a) URI: /<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数
 - { EmptyResponse
- d) 请求体示例

DELETE /EAD/AlgoFilePackages/1101150192120600000120231012203630000101

e) 响应体示例:

```
{
  "Code":0,
  "Message":"success"
}
```

B.8 边缘解析设备算法封装文件列表

a) URI: /EAD/AlgoFilePackages:search

b) 请求参数: 见表 B.3 AlgoFilePackageSearchRequest 属性信息

c) 响应参数: 见表 B.4 AlgoFilePackageSearchResponse 属性信息

d) 请求体示例

```
{
  "PageNumber":1,"
  PageSize":1
}
```

e) 响应体示例:

```
{
  "Data":{
    "PageNumber":1,
    "PageSize":1,
    "TotalSize":101,"
    Records":[
      {
        "AlgoFilePackageID":"1101150192508600000120231012203630
000101",
        "Name":"xx-headcount_alarm-2.0.0",
        "Version":"2.0.0",
        "Author":"zhangsan",
        "Description":"head count alert algorithm","
        Status":1
      }
    ],
    "Code":0,
    "Message":"success"
  }
}
```

B.9 边缘解析设备算法封装文件详情

- a) URI: /EAD/AlgoFilePackages/<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数: 见表 B.6 AlgoFilePackageDetailResponse 属性信息
- d) 请求体示例

GET

/EAD/AlgoFilePackages?AlgoFilePackageID=1101150192120600000120231012203630000101

- e) 响应体示例:

```
{
  "Data": {
    "AlgoFilePackageID": "1101150192508600000120231012203630000101",
    "Name": "xxxx",
    "Version": "3.4.0",
    "Author": "r-face",
    "Description": "xxx",
    "Tags": [
      "xxx"
    ],
    "Bin": {
      "Type": "Image",
      "remote_uri": "",
      "local_uri": ""
    },
    "algo_runtime": {
      "cmd": "",
      "Env": [
        {
          "Name": "_FILELOG_PATH",
          "Value": "/Data/limbo-log"
        },
        {
          "Name": "USE_AIPAAS_PROTO",
          "Value": "1"
        }
      ],
      "Xpu": [
        {
```



```

        "Quota":0.5,
        "Limit":0,
        "Vendor":"NVIDIA ",
        "Accelerator":"2070",
        "EADacities":{
            "Stream_Video_Quota":6,
            "Stream_EADture_Quota":0,
            "Analyze_Quota":0
        },
        "Resolution":"PIXELS_200W"
    }
]
}
},
"Code":0,
"Message":"success"
}

```

B.10 边缘解析设备算法封装文件内可执行文件下发

- a) URI: /EAD/AlgoFilePackages:pushExecFile
- b) 请求参数: 见表 B.8 AlgoFilePackageExecFilePushRequest 属性信息
- c) 响应参数: 见表 B.9 AlgoFilePackageExecFilePushResponse 属性信息
- e) 请求体示例

```

{
    "Type":"Image",
    "MD5 ":"d577273ff885c3f84dadb8578bb41620",
    "Url":"http://127.0.0.1/xxx.tar",
    "AlgoFilePackageID":"11011501925086000001202310122036
30000101"}

```

- e) 响应体示例:

```

{
    "Data":{},
    "Code":0,
    "Message":"success"
}

```

附录 C
(规范性)
算法包和算法引擎管理

C.1 感知前端算法包与算法引擎加载

- a) URI: /VITD/Machine/AlgoPackages/<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数:
EmptyResponse
- d) 请求示例:
GET
/VITD/Machine/AlgoPackages?AlgoFilePackageID=1101150192120600000120231012203630000 101
- e) 响应示例:

```
{  
  "Code": "0",  
  "Message": "success"  
}
```

C.2 感知前端算法包与算法引擎卸载

- a) URI: /VITD/Machine/AlgoPackages/<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数:
EmptyResponse
- d) 请求示例:
DELETE /VITD/Machine/AlgoPackages/1101150192120600000120231012203630000
- e) 响应示例:

```
{  
  "Code": "0",  
  "Message": "success"  
}
```

C.3 感知前端算法包与算法引擎列表查询

- a) URI: /VITD/Machine/AlgoPackages
- b) 请求参数:

表 C.1 AlgoPackageListRequest 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法包运行 状态	Status	String	20	0	算法包运行状态（RUNNING: 运行 中（安装即运行），STOPPED（卸 载）），不传表示查全部状态的包

c) 响应参数:

表 C.2 AlgoPackageListResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	结果状态码	Code	String	256	R	结果状态码(0: 成功, 其他: 失败)
2	结果描述	Message	String	1024	R	结果描述
3	数据	Data	MachineAlgoPackage []	-	R	算法包列表
4	请求ID	Request ID	String	128	0	请求ID

表 C.3 MachineAlgoPackage 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法包名称	AlgoPackageName	String	128	R	算法包名称
2	算法包类型	AlgoPackageType	String	20	R	算法包类型, BIN: 二进制算法包, ENGINE: 算法引擎
3	算法包版本号	AlgoPackageVersion	String	20	R	算法包版本号
4	算法封装文件ID	AlgoFilePackageID	String	40	R	算法封装文件ID
5	算法包运行状态	Status	String	20	0	算法包运行状态 (RUNNING: 运行中, STOPPED (卸载))
6	创建时间	CreatedDate	String	32	0	创建时间 yyyy-MM-dd HH:mm:ss.sss, 例: 2019-06-01 23:15:22.123

d) 请求示例

```
{
  "Status": "RUNNING"
}
```

e) 响应示例:

```
{
  "Code": "0",
  "Message": "success",
  "Data": [
    {
      "AlgoFilePackageID": "1101150192132600000120231012203630000103",
      "AlgoPackageName": "人体结构化",
      "AlgoPackageVersion": "1.0.0",
      "AlgoPackageType": "BIN",
      "Status": "RUNNING",
      "CreateDate": "2023-04-24 23:15:22"
    }
  ]
}
```

C.4 感知前端算法包与算法引擎详情查询

- a) URI: /VITD/Machine/AlgoPackages/<AlgoFilePackageID>
- b) 请求参数

表 C.4 请求参数属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	算法封装文件ID	AlgoFilePackageID	String	40	R	算法封装文件ID

c) 响应参数:

表 C.5 AlgoPackageDetailResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	结果状态码	Code	String	256	R	结果状态码 (0: 成功, 其他: 失败)
2	结果描述	Message	String	1024	R	结果描述
3	数据	Data	MachineAlgoPackage	-	R	算法包信息
4	请求ID	RequestID	String	128	0	请求ID

d) 请求示例:

```
GET
/VITD/Machine/AlgoPackages/?AlgoFilePackageID=1101150192120600000120231012203630000
```

e) 响应示例:

```
{
  "Code": "0",
  "Message": "success",
  "Data": {
    "AlgoFilePackageID": "1101150192132600000120231012203630000104",
    "AlgoPackageName": "人体属性事件",
    "AlgoPackageVersion": "1.0.0",
    "AlgoPackageType": "BIN",
    "Status": "RUNNING",
    "CreateDate": "2023-04-24 23:15:22"
  }
}
```

C.5 边缘解析设备算法包与算法引擎加载

a) URI: /EAD/Machine/AlgoPackages/<AlgoFilePackageID>

b) 请求参数: 算法封装文件 ID

c) 响应参数:

EmptyResponse

d) 请求示例:

GET

/EAD/Machine/AlgoPackages?AlgoFilePackageID=1101150192120600000120231012203630000

e) 响应示例:

```
{
  "Code": "0",
  "Message": "success"
}
```

C.6 边缘解析设备算法包与算法引擎卸载

a) URI: /EAD/Machine/AlgoPackages/<AlgoFilePackageID>

b) 请求参数: 算法封装文件 ID

c) 响应参数:

EmptyResponse

d) 请求示例:

DELETE /EAD/Machine/AlgoPackages/1101150192120600000120231012203630000

e) 响应示例:

```
{
  "Code": "0",
```

```
    "Message": "success"
  }
```

C.7 边缘解析设备算法包与算法引擎列表查询

- a) URI: /EAD/Machine/AlgoPackages
- b) 请求参数: 见表 C.1 AlgoPackageListRequest 属性信息
- c) 响应参数: 见表 C.2 AlgoPackageListResponse 属性信息
- d) 请求示例:

```
{
  "Status": "RUNNING"
}
```

- e) 响应示例:

```
{
  "Code": "0",
  "Message": "success",
  "Data": [
    {
      "AlgoFilePackageID": "1101150192120600000120231012203630000103",
      "AlgoPackageName": "人体结构化",
      "AlgoPackageVersion": "1.0.0",
      "AlgoPackageType": "BIN",
      "Status": "RUNNING",
      "CreateDate": "2023-04-24 23:15:22"
    }
  ]
}
```

C.8 边缘解析设备算法包与算法引擎详情查询

- a) URI: /EAD/Machine/AlgoPackages/<AlgoFilePackageID>
- b) 请求参数: 算法封装文件 ID
- c) 响应参数: 见表 C.5 AlgoPackageDetailResponse 属性信息
- d) 请求示例:

GET
/EAD/Machine/AlgoPackages ?AlgoFilePackageID=1101150192120600000120231012203630000

- e) 响应示例:

```
{
  "Code": "0",
  "Message": "success",
  "Data": {
```



```
"AlgoFilePackageID": "1101150192120600000120231012203630000104",  
"AlgoPackageName": "人体属性事件",  
"AlgoPackageVersion": "1.0.0",  
"AlgoPackageType": "BIN",  
"Status": "RUNNING",  
"CreateDate": "2023-04-24 23:15:22"  
}  
}
```



附录 D
(规范性)
任务管理

D.1 感知前端任务创建

- a) URI: /VITD/AlgoTasks
- b) 请求参数

表 D.1 AlgoTaskCreateRequest 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	重启策略	RestartPolicy	String	10	R	不重启: NEVER 重启: ALWAYS
2	任务信息	TaskInfo	AlgoTaskInfo	-	R	-
3	是否创建即启动	CreateAndStartTask	Boolean	-	0	创建即启动任务用于区分 是否在创建任务时即启 动,或是稍后按需再启动, 默认为创建即启动。 取值说明: 创建即启动: true 创建不启动: false
4	算法封装文件ID	AlgoFilePackageID	String	40	0	-

表 D.2 AlgoTaskInfo 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	任务标签	Labels	Object	-	0	指定如下任务运行的算法名 以及算法版本,可以自动调 度到合适的算法服务上。 AlgoCaBinName: 算法名称 Version: 算法版本
2	算法规则参数	RulesParams	RulesParams[]	-	R	详见: T/BAX 0006.2—2023 中表B.11解析规则参数 RuleParams特征属性。
3	流信息	StreamOptions	StreamOptions	-	R	详见: T/BAX 0006.2—2023 中表B.8设备参数配置 StreamOptions特征属性
4	解析任务ID	ID	String	128	R/0	R: 作为响应消息体,当服务 端收到客户端请求创建或查 询时,处理完成后需要返回 此ID。 0: 作为请求消息体,客户端 无须填写此字段。

c) 响应参数

表 D.3 AlgoTaskCreateResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	请求结果状态码	Code	Integer	-	R	0: 成功, 其他: 失败
2	请求结果描述	Message	String	1024	R	结果描述
3	数据	Data	AlgoTaskInfoResult	-	R	参见示例
4	请求Id	RequestId	String	128	0	请求ID

表 D.4 AlgoTaskInfoResult 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	任务信息	TaskInfo	AlgoTaskInfo	-	R	参见示例: 返回创建的任务信息, 数据类型为: AlgoTaskInfo
2	算法服务ID	AlgoServiceID	String	128	0	任务创建运行后关联的算法服务ID, 由系统调度到相关的算法服务上, 可通过查阅详情看到任务与服务的关系。
3	任务状态	Status	String	20	R	等待:PENDING 运行:RUNNING 暂停:PAUSED 完成:FINISHED 错误:ERROR
4	重启策略	RestartPolicy	String	10	R	不重启:NEVER 重启: ALWAYS
5	是否创建即启动	CreateAndStartTask	Boolean	-	0	创建即启动任务 用于区分是否在创建任务时即启动, 或是稍后按需再启动, 默认为创建即启动。 取值说明: 创建即启动: true 创建不启动: false
6	资源占用额度	Quota	Integer	-	0	返回任务资源占用情况。

d) 请求体示例

```

{
  "RestartPolicy": "ALWAYS", "
  TaskInfo": {
    "Labels": {
      "AlgoCaBinName": "megcity-a2", "
      Version": "2.0.1"
    },
    "RulesParams": [
      {
        "Areas": [
          {
            "AreaID": 0,
            "AreaType": "POLYGON", "
            Points": [
              {
                "PointX": 0.0, "
                PointY": 1.0
              },
              {
                "PointX": 1.0, "
                PointY": 1.0
              },
              {
                "PointX": 1.0, "
                PointY": 0.0
              },
              {
                "PointX": 0.0, "
                PointY": 0.0
              }
            ]
          }
        ],
        "ResultReceiveUrl": [
          "http://192.168.1.1:8001"
        ],

```

```
"EventType": "ILLEGAL_ADV", "
ExtendParams": {
    "Sensitivity": "低",
    "DurationThreshold": 1,
    "RepeatAlarmIntervalThreshold": 720
},
"Labels": {
},
"Masks": []

}

],
"StreamOptions": {
    "VideoStreamOptions": {
        "Resolution": "PIXELS_200W",
"Url": "rtsp://192.168.200.207/?DeviceID=101234200200000
046&mode=vIeo"
    }
}
}}
```

e) 响应体示例:

```
{
  "Data": {
    "Status": "PENDING",
    "RestartPolicy": "ALWAYS",
    "TaskInfo": {
      "ID": "70553686-096f-488b-8151-9c1e47adafa5",
      "Status": "PENDING",
      "StreamOptions": {
        "VideoStreamOptions": {
          "Url": "rtsp://192.168.200.207/?DeviceID=101234200200000046&mode=video"
        }
      }
    }
  },
}
```

```
    "Code":0,  
    "Message":"success"  
  }
```

D.2 感知前端任务删除

- a) URI: /VITD/AlgoTasks/<ID>
- b) 请求参数: 解析任务 ID
- c) 响应参数
EmptyResponse
- d) 请求体示例
DELETE /VITD/AlgoTasks/70553686-096f-488b-8151-9c1e47adafa5
- e) 响应体示例:
{
 "Code":0,
 "Message":"success"
}

D.3 感知前端任务启动

- a) URI: /VITD/AlgoTasks/<ID>:resume
- b) 请求参数: 任务 ID
- c) 响应参数
EmptyResponse
- d) 请求体示例
无
- e) 响应体示例:
{
 "Code":0,
 "Message":"success"
}

D.4 感知前端任务停止

- a) URI: /VITD/AlgoTasks/<ID>:stop
- b) 请求参数: 任务 ID
- c) 响应参数
EmptyResponse
- d) 请求体示例
无
- e) 响应体示例:


```
{
  "Code":0,
  "Message":"success"
}
```

D.5 感知前端任务查询列表

- a) URI: /VITD/AlgoTasks:search
- b) 请求参数

表 D.5 AlgoTaskSearchRequest 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	事件类型	EventType	String	128	0	-
2	条数	PageSize	Integer	-	0	-
3	页码	PageNumber	Integer	-	0	-
4	重启策略	RestartPolicy	String	10	0	不重启: NEVER 重启: ALWAYS
5	是否创建即启动	CreateAndStartTask	Boolean	-	0	创建即启动任务 用于区分是否在创建 任务时即启动,或是稍 后按需再启动,默认为 创建即启动。 取值说明: 创建即启动: true 创建不启动: false
6	任务状态	Status	String	20	0	等待:PENDING 运行:RUNNING 暂停:PAUSED 完成:FINISHED 错误:ERROR

- c) 响应参数

表 D.6 AlgoTaskSearchResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	请求结果状态码	Code	Integer	-	R	0: 成功, 其他: 失败
2	请求结果描述	Message	String	1024	R	-

表 D. 6 AlgoTaskSearchResponse 属性（续）

序号	名称	标识符	类型	长度	必选/可选	备注
3	数据	Data	AlgoTaskInfoResult[]	-	R	参见示例： 返回符合条件的任务信息数组，完整字段信息可用详情接口。
4	请求ID	RequestId	String	128	0	请求ID

d) 请求体示例

```
{
  "PageNumber":1,
  "PageSize":1
}
```

e) 响应体示例：

```
{
  "Code":0,
  "Message":"success",
  "Data":{
    "PageNumber":1,"
    PageSize":1,
    "TotalSize":101,"
    Records:[
      {
        "RestartPolicy":"ALWAYS",
        "TaskInfo":{
          "ID":"70553686-096f-488b-8151-9c1e47adafa5",
          "Labels":{
            "AlgoCaBinName":"megcity-a2",
            "Version":"2.0.1"
          }
        }
      },
      "Status":"PENDING"
    ]
  }
}
```

D.6 感知前端任务查询详情

- a) URI: /VITD/AlgoTasks/<ID>
- b) 请求参数: 任务 ID
- c) 响应参数

表 D.7 AlgoTaskDetailResponse 属性

序号	名称	标识符	类型	长度	必选/可选	备注
1	请求结果状态码	Code	Integer	-	R	0: 成功, 其他: 失败
2	请求结果描述	Message	String	1024	R	结果描述
3	数据	Data	AlgoTaskInfoResult	-	R	参见示例: 返回明细数据。
4	请求Id	RequestId	String	128	0	请求ID

d) 请求体示例

GET /VITD/AlgoTasks?ID=70553686-096f-488b-8151-9cle47adafa5

e) 响应体示例:

```
{
  "Data": {
    "RestartPolicy": "ALWAYS",

    "AlgoServiceID": "76079e21-1b8b-4151-b904-6c56a4635d58",
    "TaskInfo": {
      "ID": "70553686-096f-488b-8151-9cle47adafa5",
      "Labels": {
        "AlgoCaBinName": "megcity-a2",
        "Version": "2.0.1"
      },
      "RulesParams": [
        {
          "Areas": [
            {
              "AreaID": 0,
              "AreaType": "POLYGON",
              "Points": [
```

```

        {
            "PointX":0.0,
            "PointY":1.0
        },
        {
            "PointX":1.0,
            "PointY":1.0
        },
        {
            "PointX":1.0,
            "PointY":0.0
        },
        {
            "PointX":0.0,
            "PointY":0.0
        }
    ]
}
],
"ResultReceiveUrl":[
    "http://192.168.1.1:8001"
],
"EventType":"ILLEGAL_ADV",
"ExtendParams":{
    "Sensitivity":"低",
    "DurationThreshold":1,
    "RepeatAlarmIntervalThreshold":720
},
"Labels":{
},
"Masks":[
]
}
],
"StreamOptions":{
    "VideoStreamOptions":{
        "Resolution":"PIXELS_200W",

```

```

    "Url": "rtsp://192.168.200.207/?DeviceID=101234200200000
    046&mode=vIdeo"
  },
  "Status": "PENDING"
},
"Code": 0,
"Message": "success"
}

```

D.7 边缘解析设备任务创建

- a) URI: /EAD/AlgoTasks
- b) 请求参数: 见表 D.1 AlgoTaskCreateRequest 属性信息
- c) 响应参数: 见表 D.3 AlgoTaskCreateResponse 属性信息
- d) 请求体示例

```

{
  "RestartPolicy": "ALWAYS",
  "TaskInfo": {
    "Labels": {
      "AlgoCaBinName": "megcity-a2",
      "Version": "2.0.1"
    },
    "RulesParams": [
      {
        "Areas": [
          {
            "AreaID": 0,
            "AreaType": "POLYGON",
            "Points": [
              {
                "PointX": 0.0,
                "PointY": 1.0
              },
              {
                "PointX": 1.0,
                "PointY": 1.0
              }
            ]
          }
        ]
      }
    ]
  }
}

```

```
{
    "PointX":1.0,
    "PointY":0.0
},
{
    "PointX":0.0,
    "PointY":0.0
}
]
}
],
"ResultReceiveUrl":[
    "http://192.168.1.1:8001"
],
"EventType":"ILLEGAL_ADV",
"ExtendParams":{
    "Sensitivity":"低",
    "DurationThreshold":1,
    "RepeatAlarmIntervalThreshold":720
},
"Labels":{
},
"Masks":[
]
}
],
"StreamOptions":{
    "VideoStreamOptions":{
        "Resolution":"PIXELS_200W",
        "Url":"rtsp://192.168.200.207/?DeviceID=101234200200000046&mode=video"
    }
}
}
```

e) 响应体示例:

```
{
```



```
    "Data": {
      "Status": "PENDING",
      "RestartPolicy": "ALWAYS",
      "TaskInfo": {
        "ID": "70553686-096f-488b-8151-9c1e47adafa5",
        "Status": "PENDING",
        "StreamOptions": {
          "VideoStreamOptions": {
            "Url": "rtsp://192.168.200.207/?DeviceID=101234200200000046&mode=vIdeo"
          }
        }
      },
      "Code": 0,
      "Message": "success"
    }
  }
```

D.8 边缘解析设备任务删除

- a) URI: /EAD/AlgoTasks/<ID>
- b) 请求参数: 任务 ID
- c) 响应参数
EmptyResponse
- d) 请求体示例
DELETE /EAD/AlgoTasks/70553686-096f-488b-8151-9c1e47adafa5
- e) 响应体示例:
{
 "Code": 0,
 "Message": "success",
}

D.9 边缘解析设备任务启动

- a) URI: /EAD/AlgoTasks/<ID>:resume
- b) 请求参数: 任务 ID
- c) 响应参数
EmptyResponse
- d) 请求体示例
无
- e) 响应体示例:

```
{
  "Code":0,
  "Message":"success"
}
```

D. 10 边缘解析设备任务停止

- a) URI: /EAD/AlgoTasks/<ID>:stop
- b) 请求参数: 任务 ID
- c) 响应参数
EmptyResponse
- d) 请求体示例
无
- e) 响应体示例:

```
{
  "Code":0,
  "Message":"success"
}
```

D. 11 边缘解析设备任务查询列表

- a) URI: /EAD/AlgoTasks:search
- b) 请求参数: 见表 D. 5 AlgoTaskSearchRequest 属性信息
- c) 响应参数: 见表 D. 6 AlgoTaskSearchResponse 属性信息
- d) 请求体示例

```
{
  "PageNumber":1,"
  PageSize":1
}
```

- e) 响应体示例:

```
{
  "Code":0,
  "Message":"success",
  "Data":{
    "PageNumber":1,"
    PageSize":1,
    "TotalSize":101,"
    "Records":[
      {
        "RestartPolicy":"ALWAYS",
        "TaskInfo":{
```

```

        "ID": "70553686-096f-488b-8151-9c1e47adafa5",
        "Labels": {
            "AlgoCaBinName": "megcity-a2",
            "Version": "2.0.1"
        }
    },
    "Status": "PENDING"
}
]
}
}

```

D.12 边缘解析设备任务查询详情

- a) URI: /EAD/AlgoTasks/<ID>
 - b) 请求参数: 任务 ID
 - c) 响应参数: 见表 D.7 AlgoTaskDetailResponse 属性信息
 - d) 请求体示例
- GET /EAD/AlgoTasks?ID=70553686-096f-488b-8151-9c1e47adafa5

- e) 响应体示例:

```

{
    "Data": {
        "RestartPolicy": "ALWAYS",
        "AlgoServiceID": "76079e21-1b8b-4151-b904-6c56a4635d58",
        "TaskInfo": {
            "ID": "70553686-096f-488b-8151-9c1e47adafa5",
            "Labels": {
                "AlgoCaBinName": "megcity-a2",
                "Version": "2.0.1"
            }
        },
        "RulesParams": [
            {
                "Areas": [
                    {
                        "AreaID": 0,
                        "AreaType": "POLYGON",
                        "Points": [
                            {
                                "PointX": 0.0,

```

```

        "PointY":1.0
    },
    {
        "PointX":1.0,
        "PointY":1.0
    },
    {
        "PointX":1.0,
        "PointY":0.0
    },
    {
        "PointX":0.0,
        "PointY":0.0
    }
]
}
],
"ResultReceiveUrl":[
    "http://192.168.1.1:8001"
],
"EventType":"ILLEGAL_ADV",
"ExtendParams":{
    "Sensitivity":"低",
    "DurationThreshold":1,
    "RepeatAlarmIntervalThreshold":720
},
"Labels":{
},
"Masks":[
]
}
],
"StreamOptions":{
    "VideoStreamOptions":{
        "Resolution":"PIXELS_200W",
        "Url":"rtsp://192.168.200.207/?DeviceID=101234200200000046&mode=video"
    }
}

```

```
    }  
  },  
  "Status": "PENDING"  
},  
"Code": 0,  
"Message": "success"  
}
```



北京安全防范行业协会
Beijing Security and Protection Industry Association



北京安全防范行业协会
Beijing Security and Protection Industry Association