



(12) 发明专利

(10) 授权公告号 CN 113687947 B

(45) 授权公告日 2025. 03. 28

(21) 申请号 202110981773.2

(22) 申请日 2021.08.25

(65) 同一申请的已公布的文献号

申请公布号 CN 113687947 A

(43) 申请公布日 2021.11.23

(73) 专利权人 京东方科技集团股份有限公司

地址 100015 北京市朝阳区酒仙桥路10号

(72) 发明人 郭小琴 张敬宇 孙玉海 张洋

张长峰

(74) 专利代理机构 北京律智知识产权代理有限公司

公司 11438

专利代理师 王辉 阚梓瑄

(51) Int. Cl.

G06F 9/50 (2006.01)

G06V 40/16 (2022.01)

(56) 对比文件

CN 104363300 A, 2015.02.18

CN 110944146 A, 2020.03.31

审查员 刘文帅

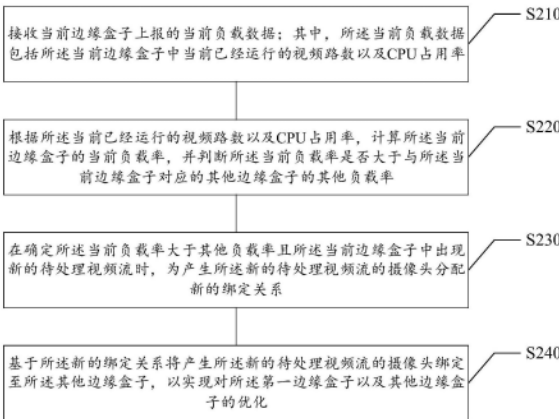
权利要求书4页 说明书17页 附图8页

(54) 发明名称

边缘盒子的优化方法及装置、存储介质、电子设备

(57) 摘要

本公开是关于一种边缘盒子的优化方法及装置、存储介质、电子设备,涉及人工智能技术领域,该方法包括:接收当前边缘盒子上报的当前负载数据;根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现对所述第一边缘盒子以及其他边缘盒子的优化。本公开解决了边缘盒子的利用率不均的问题。



1. 一种边缘盒子的优化方法,其特征在于,配置于中心服务器,所述边缘盒子的优化方法包括:

接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;

根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;

在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;

基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,并基于所述其他边缘盒子对所述新的待处理视频流进行处理,以实现所述当前边缘盒子以及其他边缘盒子的优化;

其中,对所述新的待处理视频流进行处理包括:根据所述摄像头的流地址拉取所述新的待处理视频流,并对所述新的待处理视频流进行解码,得到多帧待识别图像帧;根据所述摄像头中绑定的算法类别,调用所述当前边缘盒子中与所述算法类别对应的算法模型,并利用所述算法模型对所述待识别图像帧进行识别,得到当前识别结果;其中,算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法中的至少一种。

2. 根据权利要求1所述的边缘盒子的优化方法,其特征在于,根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,包括:

获取所述当前边缘盒子所能支持的总视频路数,并计算所述当前已经运行的视频路数在所述总视频路数中的路数占比;

根据所述路数占比以及所述路数占比的第一权重值、所述CPU占用率以及所述CPU占用率的第二权重值,计算所述当前边缘盒子的当前负载率。

3. 根据权利要求1所述的边缘盒子的优化方法,其特征在于,所述边缘盒子的优化方法还包括:

接收所述当前边缘盒子和/或所述其他边缘盒子发送的第一注册请求;

对所述第一注册请求进行解析,得到所述当前边缘盒子和/或其他边缘盒子的第一属性信息;其中,所述第一属性信息包括盒子名称、MAC地址、算力以及所能支持的总视频路数中的一种或多种;

根据所述第一属性信息对所述当前边缘盒子和/或所述其他边缘盒子进行注册。

4. 根据权利要求3所述的边缘盒子的优化方法,其特征在于,所述边缘盒子的优化方法还包括:

接收摄像头发送的第二注册请求,并对所述第二注册请求进行解析,得到所述摄像头的第二属性信息;其中,所述第二属性信息包括摄像头名称、IP地址、端口地址、流地址以及绑定的算法类别中的一种或多种;

根据所述第二属性信息对所述摄像头进行注册。

5. 根据权利要求4所述的边缘盒子的优化方法,其特征在于,所述边缘盒子的优化方法还包括:

根据所述当前边缘盒子的算力以及所能支持的总视频路数,计算所述当前边缘盒子所

能容纳的第一摄像头个数;和/或

根据所述其他边缘盒子的算力以及所能支持的总视频路数,计算所述其他边缘盒子所能容纳的第二摄像头个数;

根据所述第一摄像头个数和/或第二摄像头个数、所述MAC地址以及所述端口地址、IP地址,建立所述摄像头与所述当前边缘盒子和/或所述其他边缘盒子之间的绑定关系;

基于所述绑定关系对所述摄像头与所述当前边缘盒子和/或其他边缘盒子进行绑定,并将绑定关系发送至所述当前边缘盒子和/或其他边缘盒子。

6.一种边缘盒子的优化方法,其特征在于,配置于边缘盒子,所述边缘盒子的优化方法包括:

根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧;

在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理,包括:根据所述摄像头的流地址拉取所述待处理视频流,并对所述待处理视频流进行解码,得到多帧待识别图像帧;根据所述摄像头中绑定的算法类别,调用所述当前边缘盒子中与所述算法类别对应的算法模型,并利用所述算法模型对所述待识别图像帧进行识别,得到当前识别结果;其中,算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法中的至少一种;

计算所述当前边缘盒子中当前已经运行的视频路数,并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率,生成当前负载数据;

将所述当前负载数据上报至中心服务器,以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率,并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化,包括:在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,并基于所述其他边缘盒子对所述新的待处理视频流进行处理,以实现所述当前边缘盒子以及其他边缘盒子的优化。

7.根据权利要求6所述的边缘盒子的优化方法,其特征在于,利用所述人脸识别算法对所述待识别图像帧进行识别,得到当前识别结果,包括:

利用预设的人脸检测与关键点定位工具,检测所述待识别图像帧的待识别人脸区域;

在所述待识别人脸区域中提取所述待识别图像帧的待识别脸部关键点,并根据所述待识别脸部关键点,计算所述待识别人脸特征;

对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对,得到所述当前识别结果。

8.根据权利要求7所述的边缘盒子的优化方法,其特征在于,对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对,得到所述当前识别结果,包括:

计算所述待识别人脸特征以及所述当前边缘盒子中的特征值库中的原始人脸特征之间的欧式距离;

根据所述欧式距离,得到所述待识别人脸特征与所述原始人脸特征之间的相似度比对结果。

9. 根据权利要求8所述的边缘盒子的优化方法,其特征在于,所述边缘盒子的优化方法还包括:

判断所述相似度比对结果是否大于预设阈值;

若是,则根据所述相似度比对结果生成识别成功的消息,并将所述识别成功的消息上报至所述中心服务器;

若否,则对所述待识别人脸特征进行广播发送给所述其他边缘盒子,并在其他边缘盒子中对所述待识别人脸特征进行识别,得到其他识别结果。

10. 根据权利要求9所述的边缘盒子的优化方法,其特征在于,所述边缘盒子的优化方法还包括:

如果任一其他识别结果为识别成功,则将该识别成功的其他识别结果上报至所述中心服务器;

提取与识别成功的其他识别结果对应的其他边缘盒子中的特征值库中与所述待识别人脸特征对应的原始人脸特征;

利用所述原始人脸特征对所述当前边缘盒子中的特征值库进行更新。

11. 根据权利要求9所述的边缘盒子的优化方法,其特征在于,所述边缘盒子的优化方法还包括:

如果所有的其他识别结果均为识别失败,则将所述待识别人脸特征上报至所述中心服务器,以使得所述中心服务器对所述待识别人脸特征进行识别。

12. 一种边缘盒子的优化装置,其特征在于,配置于中心服务器,所述边缘盒子的优化装置包括:

负载数据接收模块,用于接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;

第一计算模块,用于根据所述当前已经运行的视频路数以及CPU占用率计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;

第一绑定关系分配模块,用于在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;

第一摄像头绑定模块,用于基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,并根据所述其他边缘盒子对所述新的待处理视频流进行处理,以实现所述当前边缘盒子以及其他边缘盒子的优化;

其中,对所述新的待处理视频流进行处理包括:根据所述摄像头的流地址拉取所述新的待处理视频流,并对所述新的待处理视频流进行解码,得到多帧待识别图像帧;根据所述摄像头中绑定的算法类别,调用所述当前边缘盒子中与所述算法类别对应的算法模型,并利用所述算法模型对所述待识别图像帧进行识别,得到当前识别结果;其中,算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法中的至少一种。

13. 一种边缘盒子的优化装置,其特征在于,配置于边缘盒子,所述边缘盒子的优化装置包括:

摄像头确定模块,用于根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边

缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧;

第一视频流处理模块,用于在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理,包括:根据所述摄像头的流地址拉取所述待处理视频流,并对所述待处理视频流进行解码,得到多帧待识别图像帧;根据所述摄像头中绑定的算法类别,调用所述当前边缘盒子中与所述算法类别对应的算法模型,并利用所述算法模型对所述待识别图像帧进行识别,得到当前识别结果;其中,算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法中的至少一种;

负载数据生成模块,用于计算所述当前边缘盒子中当前已经运行的视频路数,并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率,生成当前负载数据;

负载数据上报模块,用于将所述当前负载数据上报至中心服务器,以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率,并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化,包括:在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,并基于所述其他边缘盒子对所述新的待处理视频流进行处理,以实现所述当前边缘盒子以及其他边缘盒子的优化。

14.一种计算机可读存储介质,其上存储有计算机程序,其特征在于,所述计算机程序被处理器执行时实现权利要求1-11任一项所述的边缘盒子的优化方法。

15.一种电子设备,其特征在于,包括:

处理器;以及

存储器,用于存储所述处理器的可执行指令;

其中,所述处理器配置为经由执行所述可执行指令来执行权利要求1-11任一项所述的边缘盒子的优化方法。

边缘盒子的优化方法及装置、存储介质、电子设备

技术领域

[0001] 本公开实施例涉及人工智能技术领域,具体而言,涉及一种边缘盒子的优化方法、边缘盒子的优化装置、计算机可读存储介质以及电子设备。

背景技术

[0002] 随着物联网、人工智能等相关技术的普及,将人工智能算法在中心服务器运行计算的方式,存在着延时时间长、网络负载高、成本高、安全性能低等问题。为了解决该问题,采用了将人工智能算法在中心服务器计算的方式转移到边缘端进行计算。

[0003] 需要说明的是,在上述背景技术部分发明的信息仅用于加强对本公开的背景的理解,因此可以包括不构成对本领域普通技术人员已知的现有技术的信息。

发明内容

[0004] 本公开的目的在于提供一种边缘盒子的优化方法、边缘盒子的优化装置、计算机可读存储介质以及电子设备,进而至少在一定程度上克服由于相关技术的限制和缺陷而导致的边缘盒子的利用率不均的问题。

[0005] 根据本公开的一个方面,提供一种边缘盒子的优化方法,配置于中心服务器,所述边缘盒子的优化方法包括:

[0006] 接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;

[0007] 根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;

[0008] 在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;

[0009] 基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现所述第一边缘盒子以及其他边缘盒子的优化。

[0010] 在本公开的一种示例性实施例中,根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,包括:

[0011] 获取所述当前边缘盒子所能支持的总视频路数,并计算所述当前已经运行的视频路数在所述总视频路数中的路数占比;

[0012] 根据所述路数占比以及所述路数占比的第一权重值、所述CPU占用率以及所述CPU占用率的第二权重值,计算所述当前边缘盒子的当前负载率。

[0013] 在本公开的一种示例性实施例中,所述边缘盒子的优化方法还包括:

[0014] 接收所述当前边缘盒子和/或所述其他边缘盒子发送的第一注册请求;

[0015] 对所述第一注册请求进行解析,得到所述当前边缘盒子和/或其他边缘盒子的第一属性信息;其中,所述第一属性信息包括盒子名称、MAC地址、算力以及所能支持的总视频

路数中的一种或多种;

[0016] 根据所述第一属性信息对所述当前边缘盒子和/或所述其他边缘盒子进行注册。

[0017] 在本公开的一种示例性实施例中,所述边缘盒子的优化方法还包括:

[0018] 接收摄像头发送的第二注册请求,并对所述第二注册请求进行解析,得到所述摄像头的第二属性信息;其中,所述第二属性信息包括摄像头名称、IP地址、端口地址、流地址以及绑定的算法类别中的一种或多种;

[0019] 根据所述第二属性信息对所述摄像头进行注册。

[0020] 在本公开的一种示例性实施例中,所述边缘盒子的优化方法还包括:

[0021] 根据所述当前边缘盒子的算力以及所能支持的总视频路数,计算所述当前边缘盒子所能容纳的第一摄像头个数;和/或

[0022] 根据所述其他边缘盒子的算力以及所能支持的总视频路数,计算所述其他边缘盒子所能容纳的第二摄像头个数;

[0023] 根据所述第一摄像头个数和/或第二摄像头个数、所述MAC地址以及所述端口地址、IP地址,建立所述摄像头与所述当前边缘盒子和/或所述其他边缘盒子之间的绑定关系;

[0024] 基于所述绑定关系对所述摄像头与所述当前边缘盒子和/或其他边缘盒子进行绑定,并将绑定关系发送至所述当前边缘盒子和/或其他边缘盒子。

[0025] 根据本公开的一个方面,提供一种边缘盒子的优化方法,配置于边缘盒子,所述边缘盒子的优化方法包括:

[0026] 根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧;

[0027] 在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理;

[0028] 计算所述当前边缘盒子中当前已经运行的视频路数,并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率,生成当前负载数据;

[0029] 将所述当前负载数据上报至中心服务器,以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率,并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化。

[0030] 在本公开的一种示例性实施例中,对所述待处理视频流进行处理,包括:

[0031] 根据所述摄像头的流地址拉取所述待处理视频流,并对所述待处理视频流进行解码,得到多帧待识别图像帧;

[0032] 根据所述摄像头中绑定的算法类别,调用所述当前边缘盒子中与所述算法类别对应的算法模型,并利用所述算法模型对所述待识别图像帧进行识别,得到当前识别结果;其中,所述算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法中的至少一种。

[0033] 在本公开的一种示例性实施例中,利用所述人脸识别算法对所述待识别图像帧进行识别,得到当前识别结果,包括:

[0034] 利用预设的人脸检测与关键点定位工具,检测所述待识别图像帧的待识别人脸区域;

[0035] 在所述待识别人脸区域中提取所述待识别图像帧的待识别脸部关键点,并根据所

述待识别脸部关键点,计算所述待识别人脸特征;

[0036] 对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对,得到所述当前识别结果。

[0037] 在本公开的一种示例性实施例中,对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对,得到所述当前识别结果,包括:

[0038] 计算所述待识别人脸特征以及所述当前边缘盒子中的特征值库中的原始人脸特征之间的欧式距离;

[0039] 根据所述欧式距离,得到所述待识别人脸特征与所述原始人脸特征之间的相似度比对结果。

[0040] 在本公开的一种示例性实施例中,所述边缘盒子的优化方法还包括:

[0041] 判断所述相似度比对结果是否大于预设阈值;

[0042] 若是,则根据所述相似度比对结果生成识别成功的消息,并将所述识别成功的消息上报至所述中心服务器;

[0043] 若否,则对所述待识别人脸特征进行广播发送给所述其他边缘盒子,并在其他边缘盒子中对所述待识别人脸特征进行识别,得到其他识别结果。

[0044] 在本公开的一种示例性实施例中,所述边缘盒子的优化方法还包括:

[0045] 如果任一其他识别结果为识别成功,则将该识别成功的其他识别结果上报至所述中心服务器;

[0046] 提取与识别成功的其他识别结果对应的其他边缘盒子中的特征值库中与所述待识别人脸特征对应的原始人脸特征;

[0047] 利用所述原始人脸特征对所述当前边缘盒子中的特征值库进行更新。

[0048] 在本公开的一种示例性实施例中,所述边缘盒子的优化方法还包括:

[0049] 如果所有的其他识别结果均为识别失败,则将所述待识别人脸特征上报至所述中心服务器,以使得所述中心服务器对所述待识别人脸特征进行识别。

[0050] 根据本公开的一个方面,提供一种边缘盒子的优化装置,配置于中心服务器,所述边缘盒子的优化装置包括:

[0051] 负载数据接收模块,用于接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;

[0052] 第一计算模块,用于根据所述当前已经运行的视频路数以及CPU占用率计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;

[0053] 第一绑定关系分配模块,用于在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;

[0054] 第一摄像头绑定模块,用于基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现所述第一边缘盒子以及其他边缘盒子的优化。

[0055] 根据本公开的一个方面,提供一种边缘盒子的优化装置,配置于边缘盒子,所述边缘盒子的优化装置包括:

[0056] 摄像头确定模块,用于根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧;

[0057] 第一视频流处理模块,用于在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理;

[0058] 负载数据生成模块,用于计算所述当前边缘盒子中当前已经运行的视频路数,并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率,生成当前负载数据;

[0059] 负载数据上报模块,用于将所述当前负载数据上报至中心服务器,以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率,并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化。

[0060] 根据本公开的一个方面,提供一种计算机可读存储介质,其上存储有计算机程序,所述计算机程序被处理器执行时实现上述任意一项所述的边缘盒子的优化方法。

[0061] 根据本公开的一个方面,提供一种电子设备,包括:

[0062] 处理器;以及

[0063] 存储器,用于存储所述处理器的可执行指令;

[0064] 其中,所述处理器配置为经由执行所述可执行指令来执行上述任意一项所述的边缘盒子的优化方法。

[0065] 本公开实施例提供的一种边缘盒子的优化方法,一方面,由于可以根据当前已经运行的视频路数以及CPU占用率,计算当前边缘盒子的当前负载率,并判断当前负载率是否大于与当前边缘盒子对应的其他边缘盒子的其他负载率;并在确定当前负载率大于其他负载率且当前边缘盒子中出现新的待处理视频流时,为产生新的待处理视频流的摄像头分配新的绑定关系;最后基于新的绑定关系将产生新的待处理视频流的摄像头绑定至其他边缘盒子,进而可以通过其他边缘盒子对新的待处理视频流进行处理,实现了对第一边缘盒子以及其他边缘盒子的优化,解决了现有技术中无法根据边缘盒子的运算负载情况将摄像头重新自动分配给其他边缘盒子,进而使得边缘盒子的利用率不均的问题,降低了当前边缘盒子的负载,同时提高了其他边缘盒子的利用率;另一方面,可以基于新的绑定关系将产生新的待处理视频流的摄像头自动的绑定至其他边缘盒子,进而提高了摄像头的绑定效率,进而提升了新的待处理视频流的处理效率。

[0066] 应当理解的是,以上的一般描述和后文的细节描述仅是示例性和解释性的,并不能限制本公开。

附图说明

[0067] 此处的附图被并入说明书中并构成本说明书的一部分,示出了符合本公开的实施例,并与说明书一起用于解释本公开的原理。显而易见地,下面描述中的附图仅仅是本公开的一些实施例,对于本领域普通技术人员来讲,在不付出创造性劳动的前提下,还可以根据这些附图获得其他的附图。

[0068] 图1示意性示出一种在边缘盒子中进行计算的场景示例图。

- [0069] 图2示意性示出根据本公开示例实施例的一种在中心服务器侧的边缘盒子的优化方法的流程图。
- [0070] 图3示意性示出根据本公开示例实施例的一种边缘盒子的优化系统的框图。
- [0071] 图4示意性示出根据本公开示例实施例的一种边缘盒子的注册方法的流程图。
- [0072] 图5示意性示出根据本公开示例实施例的一种摄像头的注册方法的流程图。
- [0073] 图6示意性示出根据本公开示例实施例的一种边缘盒子与摄像头之间的绑定关系建立的方法流程图。
- [0074] 图7示意性示出根据本公开示例实施例的一种原始绑定关系的示例图。
- [0075] 图8示意性示出根据本公开示例实施例的一种重新分配的绑定关系的示例图。
- [0076] 图9示意性示出根据本公开示例实施例的一种在边缘盒子侧的边缘盒子的优化方法的流程图。
- [0077] 图10示意性示出根据本公开示例实施例的一种对待处理视频流进行处理的方法流程图。
- [0078] 图11示意性示出根据本公开示例实施例的一种脸部轮廓关键点的示例图。
- [0079] 图12示意性示出根据本公开示例实施例的另一种边缘盒子的优化方法的流程图。
- [0080] 图13示意性示出根据本公开示例实施例的一种人脸特征比对的场景示例图。
- [0081] 图14示意性示出根据本公开示例实施例的一种在中心服务器侧的边缘盒子的优化装置的框图。
- [0082] 图15示意性示出根据本公开示例实施例的一种在边缘盒子侧的边缘盒子的优化装置的框图。
- [0083] 图16示意性示出根据本公开示例实施例的一种用于实现上述边缘盒子的优化方法的电子设备。

具体实施方式

[0084] 现在将参考附图更全面地描述示例实施方式。然而,示例实施方式能够以多种形式实施,且不应被理解为限于在此阐述的范例;相反,提供这些实施方式使得本公开将更加全面和完整,并将示例实施方式的构思全面地传达给本领域的技术人员。所描述的特征、结构或特性可以以任何合适的方式结合在一个或更多实施方式中。在下面的描述中,提供许多具体细节从而给出对本公开的实施方式的充分理解。然而,本领域技术人员将意识到,可以实践本公开的技术方案而省略所述特定细节中的一个或更多,或者可以采用其它的方法、组元、装置、步骤等。在其它情况下,不详细示出或描述公知技术方案以避免喧宾夺主而使得本公开的各方面变得模糊。

[0085] 此外,附图仅为本公开的示意性图解,并非一定是按比例绘制。图中相同的附图标记表示相同或类似的部分,因而将省略对它们的重复描述。附图中所示的一些方框图是功能实体,不一定必须与物理或逻辑上独立的实体相对应。可以采用软件形式来实现这些功能实体,或在一个或多个硬件模块或集成电路中实现这些功能实体,或在不同网络和/或处理器装置和/或微控制器装置中实现这些功能实体。

[0086] 在一些通过边缘盒子进行计算的方式中,主要是通过如下方式实现的:首先,人工的在中心服务器端,按照各边缘盒子的算力分配不同数量的摄像头,当检测到摄像头中出

现人脸时,在该边缘盒子中对该摄像头产生的视频流进行处理。具体的,参考图1所示,以银行网点应用场景为例,按图1给网点内的边缘盒子分配摄像头后,可能会出现边缘盒子1分配的3个摄像头同时需要进行计算,而其他边缘盒子的摄像头都没有开启或摄像头画面没有人员进入触发边缘运算,因此造成的结果就是边缘盒子1满负载运算,而边缘盒子2和边缘盒子3处于闲置状态,也即边缘盒子1的负载过高,而边缘盒子2以及边缘盒子3的利用率较低。因此,上述方法无法根据边缘盒子的运算负载情况将摄像头重新自动分配给其他边缘盒子,进而使得边缘盒子的利用率不均。

[0087] 基于此,本示例实施方式中首先提供了一种边缘盒子的优化方法,该方法可以运行于中心服务器、服务器集群或云服务器等;当然,本领域技术人员也可以根据需求在其他平台运行本公开的方法,本示例性实施例中对此不做特殊限定。参考图2所示,该边缘盒子的优化方法可以包括以下步骤:

[0088] 步骤S210.接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;

[0089] 步骤S220.根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;

[0090] 步骤S230.在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;

[0091] 步骤S240.基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现所述第一边缘盒子以及其他边缘盒子的优化。

[0092] 上述边缘盒子的优化方法中,一方面,由于可以根据当前已经运行的视频路数以及CPU占用率,计算当前边缘盒子的当前负载率,并判断当前负载率是否大于与当前边缘盒子对应的其他边缘盒子的其他负载率;并在确定当前负载率大于其他负载率且当前边缘盒子中出现新的待处理视频流时,为产生新的待处理视频流的摄像头分配新的绑定关系;最后基于新的绑定关系将产生新的待处理视频流的摄像头绑定至其他边缘盒子,进而可以通过其他边缘盒子对新的待处理视频流进行处理,实现了对第一边缘盒子以及其他边缘盒子的优化,解决了现有技术中无法根据边缘盒子的运算负载情况将摄像头重新自动分配给其他边缘盒子,进而使得边缘盒子的利用率不均的问题,降低了当前边缘盒子的负载,同时提高了其他边缘盒子的利用率;另一方面,可以基于新的绑定关系将产生新的待处理视频流的摄像头自动的绑定至其他边缘盒子,进而提高了摄像头的绑定效率,进而提升了新的待处理视频流的处理效率。

[0093] 以下,将结合附图对本公开示例实施例边缘盒子的优化方法进行详细的解释以及说明。

[0094] 首先,对本公开示例实施例的发明目的进行解释以及说明。具体的,本公开示例实施例提供了一种AI边缘计算智能系统的优化方法,通过在边缘盒子端将前置算法模型到AI芯片中;系统初始运行时,在中心服务器注册所有边缘盒子和摄像头,手动将摄像头绑定到不同盒子上,并将绑定关系下发给边缘盒子;对于接入的摄像头,边缘盒子会采集摄像头的图像,在边缘端运行算法实现人脸识别,人脸年龄性别识别,人数统计等AI功能,定时将当前盒子的运算负载情况上报给中心服务器;中心服务器根据盒子的运算负载情况将剩余摄

像头重新自动分配给边缘盒子,保证所有边缘盒子平均分配计算任务;中心服务器的特征值库分发给各边缘端,首先在本地边缘端进行人脸特征值比对,没有结果则广播给其他边缘端进行特征值比对,最后没有结果则上报给中心服务器进行特征值比对。本发明通过在云端中心服务器实时计算边缘端设备的负载能力来进行计算任务的最优分配,充分挖掘边缘端设备的算力;通过对人脸特征库比对的优化提高人脸特征比对的速度,保证人脸识别的实时性。

[0095] 其次,对本公开示例实施例的边缘盒子的优化系统进行解释以及说明。具体的,参考图3所示,该边缘盒子的优化系统可以包括中心服务器310、边缘盒子320以及摄像头330。其中,中心服务器分别与边缘盒子以及摄像头连接,边缘盒子与摄像头连接;中心服务器以及边缘盒子分别用于实现本公开示例实施例所记载的边缘盒子的优化方法,摄像头用于生产视频流。同时,本公开示例实施例中所包括的边缘盒子(边缘盒子1、边缘盒子2、…、边缘盒子j等等)可以包括多个,每个边缘盒子的算力不同,当然也可以部分相同部分不同;摄像头也可以包括多个(摄像头1,摄像头2,…,摄像头k等等),绑定在同一个边缘盒子中的摄像头所生产的视频流不同。

[0096] 进一步的,对本公开示例实施例中的边缘盒子的注册过程、摄像头的注册过程以及绑定关系的建立过程进行解释以及说明。

[0097] 首先,参考图4所示,边缘盒子在中心服务器中的注册过程可以包括以下步骤:

[0098] 步骤S410,接收所述当前边缘盒子和/或所述其他边缘盒子发送的第一注册请求;

[0099] 步骤S420,对所述第一注册请求进行解析,得到所述当前边缘盒子和/或其他边缘盒子的第一属性信息;其中,所述第一属性信息包括盒子名称、MAC地址、算力以及所能支持的总视频路数中的一种或多种;

[0100] 步骤S430,根据所述第一属性信息对所述当前边缘盒子和/或所述其他边缘盒子进行注册。

[0101] 图4示意性示出的实施例中,通过在中心服务器中对边缘盒子的第一属性信息进行注册,进而完成对该边缘盒子的注册;其中,该边缘盒子的第一属性信息可以包括边缘盒子的盒子名称、MAC(Media Access Control,媒体存取控制)地址、算力、最多能支持的视频路数等等。其中,盒子算力越高支持的视频路数越高;例如,假设边缘盒子1支持视频路数是4路,边缘盒子2支持视频路数是8路,边缘盒子3支持视频路数是2路。进一步的,当完成对边缘盒子的注册以后,即可根据该边缘盒子的算力为其分配对应数量的摄像头,进而实现在边缘盒子中完成对摄像头产生的视频流进行处理,以达到提高视频流的处理效率的目的。

[0102] 其次,参考图5所示,摄像头在中心服务器中的注册过程可以包括以下步骤:

[0103] 步骤S510,接收摄像头发送的第二注册请求,并对所述第二注册请求进行解析,得到所述摄像头的第二属性信息;其中,所述第二属性信息包括摄像头名称、IP地址、端口地址、流地址以及绑定的算法类别中的一种或多种;

[0104] 步骤S520,根据所述第二属性信息对所述摄像头进行注册。

[0105] 图5示意性示出的实施例中,通过在中心服务器注册中对摄像头的第二属性信息进行注册,进而完成对该摄像头的注册;其中,该第二属性信息可以包括摄像头名称、IP(Internet Protocol,国际互联网协议)地址、端口、流地址、绑定的算法类别等信息。其中,

流地址是标准网络视频流地址,边缘盒子可以根据流地址拉流解码出来一帧帧的视频画面(待识别图像帧);摄像头绑定的算法类别是指该路摄像头的视频流的作用,其可以包括人脸识别或者人脸性别年龄识别或者人头检测等等。进一步的,当完成摄像头的注册以后,即可为该摄像头分配对应的边缘盒子,进而在边缘盒子中对该摄像头生产的视频流进行处理。

[0106] 最后,参考图6所示,摄像头与边缘盒子之间的绑定关系建立过程可以包括以下步骤:

[0107] 步骤S610,根据所述当前边缘盒子的算力以及所能支持的总视频路数,计算所述当前边缘盒子所能容纳的第一摄像头个数;和/或

[0108] 步骤S620,根据所述其他边缘盒子的算力以及所能支持的总视频路数,计算所述其他边缘盒子所能容纳的第二摄像头个数;

[0109] 步骤S630,根据所述第一摄像头个数和/或第二摄像头个数、所述MAC地址以及所述端口地址、IP地址,建立所述摄像头与所述当前边缘盒子和/或所述其他边缘盒子之间的绑定关系;

[0110] 步骤S640,基于所述绑定关系对所述摄像头与所述当前边缘盒子和/或其他边缘盒子进行绑定,并将绑定关系发送至所述当前边缘盒子和/或其他边缘盒子。

[0111] 在图6示意性示出的实施例中,可以在中心服务器中,自动的初始化配置边缘盒子和摄像头的绑定关系,并根据边缘盒子的算力和所能支持的总视频路数配置不同数量的摄像头;同时,如果配置的摄像头数量超出边缘盒子支持的路数,则给出告警提示;其中,边缘盒子与摄像头之间的绑定关系具体可以参考图7所示。此处需要补充说明的是,为了保证摄像头地址的唯一性,可以根据该摄像头的IP地址以及端口地址,计算该摄像头的摄像头地址,进而可以避免由于地址错误导致的无法对视频流进行处理的问题。

[0112] 以下,结合图3-图6,对图2中所示出的边缘盒子的优化方法中所涉及的各步骤进行详细的解释以及说明。

[0113] 在步骤S210中,接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率。

[0114] 具体的,当中心服务器完成绑定关系的建立后,将绑定关系下发给所有的边缘盒子,边缘盒子收到该绑定关系后,根据摄像头的流地址进行拉流解码。例如,当摄像头1中有人进入时,边缘盒子1开始1路视频处理,并将负载数据上报给中心服务器;中心服务器接收当前边缘盒子上报的负载数据,该负载数据中可以包括当前已经运行的视频路数、CPU (Central Processing Unit,中央处理器)占用率,当然还可以包括该当前边缘盒子的MAC地址以及盒子名称等信息;此处需要补充说明的是,上述当前已经运行的视频路数可以被定义为当前已经运行的视频路的数量,一个视频路处理一个摄像头产生的视频流,也即当前边缘盒子中所处理的视频流的数量与当前已经运行的视频路数相同。

[0115] 在步骤S220中,根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率。

[0116] 在本示例实施例中,首先,根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率。具体的,可以包括:首先,获取所述当前边缘盒子所能

支持的总视频路数,并计算所述当前已经运行的视频路数在所述总视频路数中的路数占比;其次,根据所述路数占比以及所述路数占比的第一权重值、所述CPU占用率以及所述CPU占用率的第二权重值,计算所述当前边缘盒子的当前负载率。具体的,假设第一权重值为80%,第二权重值为20%,记当前边缘盒子所能支持的总视频路数为N,当前已经运行的视频路数为M,当前边缘盒子CPU占用K,则该当前边缘盒子的当前负载率L1为:

[0117] $L1 = (1 - M/N) * 80\% + K * 20\%$ 。

[0118] 其次,当得到该当前负载率以后,即可判断该当前负载率是否大于其他负载率;其中,其他负载率的计算方式与当前负载率的计算方式相同,此处不再赘述。

[0119] 在步骤S230中,在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系。

[0120] 具体的,假设此时中心服务器计算出的当前边缘盒子的当前负载率L1高于其他边缘盒子2的其他负载率L2和其他边缘盒子3的其他负载率L3,则当当前边缘盒子中出现新的待处理视频流时,则可以基于图2中的绑定关系,当摄像头2中有人进入画面时,中心服务器根据所有边缘盒子的负载状态自动将摄像头2绑定到其他边缘盒子2;同样,当摄像头3中有人进入画面时,中心服务器根据当前边缘盒子的负载数据自动将摄像头3绑定到其他边缘盒子3;其中,重新分配的绑定关系具体可以参考图8所示。

[0121] 在步骤S240中,基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现所述第一边缘盒子以及其他边缘盒子的优化。

[0122] 具体的,可以基于新的绑定关系将摄像头2绑定到其他边缘盒子2,在其他边缘盒子2上对摄像头2生产的待处理视频流进行处理,同时将其他边缘盒子2的负载数据上报给中心服务器;同样,当摄像头3中有人进入画面时,可以基于新的绑定关系将摄像头3绑定到其他边缘盒子3,在其他边缘盒子3上对摄像头3生产的待处理视频流进行处理,同时将其他边缘盒子3的负载数据上报给中心服务器。通过该方法,中心服务器可以根据所有的边缘盒子的负载率,自动的对摄像头进行动态分配,保证所有边缘盒子平均分配计算任务,在降低边缘盒子的负载的同时提高边缘盒子的利用率。

[0123] 本公开示例实施例还提供了另一种边缘盒子的优化方法,配置于边缘盒子。参考图9所示,该边缘盒子的优化方法可以包括:

[0124] 步骤S910,根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧。

[0125] 具体的,当前边缘盒子接收到绑定关系以后,即可确定自身被绑定的摄像头,并检测摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧,也即检测摄像头中是否有人进入画面。其中,对包括目标对象的视频帧进行检测可以通过常用的图像检测算法进行实现,例如R-CNN或者Fast R-CNN等等,本示例对此不做特殊限制。

[0126] 步骤S920,在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理。

[0127] 具体的,在检测到当前视频流中有人进入画面时,即可将当前视频流作为待处理视频流;若没有检测到当前视频流中有人进入画面,则无需处理。进一步的,可以运行一未运行的视频路对该待处理视频流进行处理。

[0128] 步骤S930, 计算所述当前边缘盒子中当前已经运行的视频路数, 并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率, 生成当前负载数据。

[0129] 具体的, 可以统计该当前边缘盒子当前已经运行的所有视频路的路数, 再根据该当前已经运行的所有视频路的路数以及该处理该待处理视频流产生的CPU占用率, 生成当前负载数据。

[0130] 步骤S940, 将所述当前负载数据上报至中心服务器, 以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率, 并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化。

[0131] 图9所示出的边缘盒子的优化方法中, 由于中心服务器可以通过边缘盒子自动上报的当前负载数据计算当前边缘盒子的当前负载率, 进而比对当前负载率与其他边缘盒子的其他负载率之间大小, 以根据比对结果对摄像头进行动态调整, 进而解决现有技术中的无法根据边缘盒子的运算负载情况将摄像头重新自动分配给其他边缘盒子, 进而使得边缘盒子的利用率不均的问题, 降低了当前边缘盒子的负载, 同时提高了其他边缘盒子的利用率。

[0132] 图10示意性示出一种对待处理视频流进行处理的方法流程图。参考图10所示, 对待处理视频流进行处理可以包括以下步骤:

[0133] 步骤S1010, 根据所述摄像头的流地址拉取所述待处理视频流, 并对所述待处理视频流进行解码, 得到多帧待识别图像帧。

[0134] 具体的, 首先, 根据摄像头的流地址拉取待处理视频流, 其次, 对待处理视频流进行解码, 进而得到多帧包括人像的待识别图像帧; 其中, 对视频流进行解码, 可以基于常用的解码器即可; 所得到的待识别图像帧可以是YUV格式的图像, 也可以是RGB格式的图像, 本示例对此不做特殊限制。

[0135] 步骤S1020, 根据所述摄像头中绑定的算法类别, 调用所述当前边缘盒子中与所述算法类别对应的算法模型, 并利用所述算法模型对所述待识别图像帧进行识别, 得到当前识别结果; 其中, 算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法等等。

[0136] 在本示例实施例中, 当摄像头中绑定的算法类别为人脸识别算法时, 利用人脸识别算法对待识别图像帧进行识别, 得到当前识别结果, 可以包括: 首先, 利用预设的人脸检测与关键点定位工具, 检测所述待识别图像帧的待识别人脸区域; 其次, 在所述待识别人脸区域中提取所述待识别图像帧的待识别脸部关键点, 并根据所述待识别脸部关键点, 计算所述待识别人脸特征; 最后, 对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对, 得到所述当前识别结果。

[0137] 进一步的, 对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对, 得到所述当前识别结果, 包括: 计算所述待识别人脸特征以及所述当前边缘盒子中的特征值库中的原始人脸特征之间的欧式距离; 根据所述欧式距离, 得到所述待识别人脸特征与所述原始人脸特征之间的相似度比对结果。

[0138] 详细而言, 可以采用机器学习或深度学习模型训练得到的通用人脸检测与关键点定位工具, 得到用户人脸图片(待融合脸)的脸部关键点; 其中, 脸部关键点为ibug-68标准, 脸部关键点包括脸部轮廓关键点以及五官关键点; 同时, 脸部轮廓关键点例如可以是图11中所示的标号为1-17标识的部分, 五官关键点包括眉毛关键点(18-22, 以及23-27)、鼻梁关

键点(28-31)、鼻翼关键点(32-36)、眼睛关键点(37-42,以及43-48)、嘴唇关键点(外侧:49-60,内侧:61-68);其次,当得到脸部关键点以后,可以根据该脸部关键点计算待识别人脸特征;进一步的,欧氏距离的计算方法具体可以如下公式(1)所示:

$$[0139] \quad dist(X, Y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}; \quad \text{公式 (1)}$$

[0140] 其中,X标识原始人脸特征,Y表示待识别人脸特征,n表示原始人脸特征以及待识别人脸特征的特征点数量。此处需要补充说明的是,相较于采用其他方式计算待识别人脸特征与原始人脸特征之间的相似度,采用计算欧式距离的方式,可以在确保计算精确度的情况下减少计算效率,进而可以减少系统的负担,还可以提升用户体验。

[0141] 此处需要进一步补充说明的是,各边缘盒子中的特征值库中的原始人脸特征是通过如下方式得到的:首先,需要在中心服务器生成一个全量的包括原始人脸特征的人脸特征值库,以银行网点应用场景为例,这个特征值库一般在200万级别以上;其中,该原始人脸特征可以通过采集身份证上的人脸图像得到;然后,中心服务器将200万的全特征值库会分发给网点下的各个边缘盒子,以使得每个边缘盒子中拥有不少于2万个原始人脸特征的特征值库。

[0142] 在本示例的又一实施例中,如果摄像头中绑定的算法类别为人脸性别年龄算法时,具体的识别过程可以为:根据提取到的待识别人脸特征计算人脸属性特征,进而将该人脸属性特征输入至人脸形变年龄算法模型中,得到对应的性别年龄;同时,如果摄像头中绑定的算法类别为人头检测算法,则可以直接将该待识别图像帧输入至人头检测算法模型中,得到各个时间段进入银行的数量(以银行场景为例,当然也可以是其他场景),同时,为了提高人数的统计结果的准确率,还需要对识别出的图像进行去重等操作,此处不再一一赘述。

[0143] 图12示意性示出本公开示例实施例的另一种边缘盒子的优化方法。

[0144] 参考图12所示,该边缘盒子的优化方法还可以以下步骤:

[0145] 步骤S1210,判断所述相似度比对结果是否大于预设阈值;

[0146] 步骤S1220,若是,则根据所述相似度比对结果生成识别成功的消息,并将所述识别成功的消息上报至所述中心服务器;

[0147] 步骤S1230,若否,则对所述待识别人脸特征进行广播发送给所述其他边缘盒子,并在其他边缘盒子中对所述待识别人脸特征进行识别,得到其他识别结果。

[0148] 进一步的,如果任一其他识别结果为识别成功,则将该识别成功的其他识别结果上报至所述中心服务器;同时,提取与识别成功的其他识别结果对应的其他边缘盒子中的特征值库中与所述待识别人脸特征对应的原始人脸特征;最后,利用所述原始人脸特征对所述当前边缘盒子中的特征值库进行更新。

[0149] 更进一步的,如果所有的其他识别结果均为识别失败,则将所述待识别人脸特征上报至所述中心服务器,以使得所述中心服务器对所述待识别人脸特征进行识别。

[0150] 详细而言,参考图13所示,当在边缘盒子2中进行人脸识别时,如果相似度比对结果大于预设阈值,则根据该相似度比对结果生成识别成功的消息,并直接将比对成功结果(识别成功的消息)上报给中心服务器的业务应用模块;如果边缘盒子2的特征值库不存在待识别的人脸特征值(也即相似度比对结果小于预设阈值),则人脸识别比对失败,此时边

缘盒子2将待识别人脸特征广播发送给其他所有边缘盒子(例如边缘盒子1以及边缘盒子3);并在边缘盒子1和边缘盒子3中同时将待识别人脸特征值进行比对得到其他识别结果;如果在其中任意一个盒子中比对成功(也即任一其他识别结果为识别成功),则直接将该识别成功的其他识别结果上报给中心服务器,同时提取与识别成功的其他识别结果对应的其他边缘盒子中的特征值库中与待识别人脸特征对应的原始人脸特征;并利用该原始人脸特征对当前边缘盒子(边缘盒子2)中的特征值库进行更新;当然,如果都比对失败,则将待识别人脸特征值发送给中心服务器进行比对;若中心服务器比对成功,可以为所有的边缘盒子分配该待识别人脸特征对应的原始人脸特征;若是失败,则通知人工服务。

[0151] 图12所示出的边缘盒子的优化方法中,当当前边缘盒子的人脸识别比对没有结果时,先将人脸特征值广播发送给其他所有边缘盒子,在其他边缘盒子的特征值库中同时进行比对,比对出来将结果上报给中心服务器,同时发送给当前边缘盒子进行特征值库更新,该比对速度会比在中心服务器的全特征值库比对速度快的多,进而解决了现有技术中如果在边缘盒子中的人脸识别比对没有结果,则直接将人脸特征值发送给中心服务器全特征值库中进行比对,进而使得人脸识别比对的延时时间会较长的问题,提高了待识别人脸特征的比对速度,同时也降低了中心服务器的负担。

[0152] 本公开示例实施例还提供了一种边缘盒子的优化装置,配置于中心服务器。参考图14所示,该边缘盒子的优化装置可以包括负载数据接收模块1410、第一计算模块1420、第一绑定关系分配模块1430以及第一摄像头绑定模块1440。其中:

[0153] 负载数据接收模块1410可以用于接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;

[0154] 第一计算模块1420可以用于根据所述当前已经运行的视频路数以及CPU占用率计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;

[0155] 第一绑定关系分配模块1430可以用于在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;

[0156] 第一摄像头绑定模块1440可以用于基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现与所述第一边缘盒子以及其他边缘盒子的优化。

[0157] 在本公开的一种示例性实施例中,根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,包括:

[0158] 获取所述当前边缘盒子所能支持的总视频路数,并计算所述当前已经运行的视频路数在所述总视频路数中的路数占比;

[0159] 根据所述路数占比以及所述路数占比的第一权重值、所述CPU占用率以及所述CPU占用率的第二权重值,计算所述当前边缘盒子的当前负载率。

[0160] 在本公开的一种示例性实施例中,所述边缘盒子的优化装置还包括:

[0161] 第一注册请求接收模块,可以用于接收所述当前边缘盒子和/或所述其他边缘盒子发送的第一注册请求;

[0162] 第一注册请求解析模块,可以用于对所述第一注册请求进行解析,得到所述当前

边缘盒子和/或其他边缘盒子的第一属性信息;其中,所述第一属性信息包括盒子名称、MAC地址、算力以及所能支持的总视频路数中的一种或多种;

[0163] 第一注册模块,可以用于根据所述第一属性信息对所述当前边缘盒子和/或所述其他边缘盒子进行注册。

[0164] 在本公开的一种示例性实施例中,所述边缘盒子的优化装置还包括:

[0165] 第二注册请求解析模块,可以用于接收摄像头发送的第二注册请求,并对所述第二注册请求进行解析,得到所述摄像头的第二属性信息;其中,所述第二属性信息包括摄像头名称、IP地址、端口地址、流地址以及绑定的算法类别中的一种或多种;

[0166] 第二注册模块,可以用于根据所述第二属性信息对所述摄像头进行注册。

[0167] 在本公开的一种示例性实施例中,所述边缘盒子的优化装置还包括:

[0168] 第一摄像头个数计算模块,可以用于根据所述当前边缘盒子的算力以及所能支持的总视频路数,计算所述当前边缘盒子所能容纳的第一摄像头个数;和/或

[0169] 第二摄像头个数计算模块,可以用于根据所述其他边缘盒子的算力以及所能支持的总视频路数,计算所述其他边缘盒子所能容纳的第二摄像头个数;

[0170] 绑定关系建立模块,可以用于根据所述第一摄像头个数和/或第二摄像头个数、所述MAC地址以及所述端口地址、IP地址,建立所述摄像头与所述当前边缘盒子和/或所述其他边缘盒子之间的绑定关系;

[0171] 绑定关系发送模块,可以用于基于所述绑定关系对所述摄像头与所述当前边缘盒子和/或其他边缘盒子进行绑定,并将绑定关系发送至所述当前边缘盒子和/或其他边缘盒子。

[0172] 本公开还提供了另一种边缘盒子的优化装置,配置于边缘盒子。参考图15所示,该边缘盒子的优化装置可以包括摄像头确定模块1510、第一视频流处理模块1520、负载数据生成模块1530以及负载数据上报模块1540。其中:

[0173] 摄像头确定模块1510可以用于根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧;

[0174] 第一视频流处理模块1520可以用于在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理;

[0175] 负载数据生成模块1530可以用于计算所述当前边缘盒子中当前已经运行的视频路数,并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率,生成当前负载数据;

[0176] 负载数据上报模块1540可以用于将所述当前负载数据上报至中心服务器,以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率,并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化。

[0177] 在本公开的一种示例性实施例中,对所述待处理视频流进行处理,包括:

[0178] 根据所述摄像头的流地址拉取所述待处理视频流,并对所述待处理视频流进行解码,得到多帧待识别图像帧;

[0179] 根据所述摄像头中绑定的算法类别,调用所述当前边缘盒子中与所述算法类别对

应的算法模型,并利用所述算法模型对所述待识别图像帧进行识别,得到当前识别结果;其中,所述算法模型包括人脸识别算法、人脸性别年龄算法以及人头检测算法中的至少一种。

[0180] 在本公开的一种示例性实施例中,利用所述人脸识别算法对所述待识别图像帧进行识别,得到当前识别结果,包括:

[0181] 利用预设的人脸检测与关键点定位工具,检测所述待识别图像帧的待识别人脸区域;

[0182] 在所述待识别人脸区域中提取所述待识别图像帧的待识别脸部关键点,并根据所述待识别脸部关键点,计算所述待识别人脸特征;

[0183] 对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对,得到所述当前识别结果。

[0184] 在本公开的一种示例性实施例中,对所述待识别人脸特征以及预设的特征值库中的原始人脸特征进行特征比对,得到所述当前识别结果,包括:

[0185] 计算所述待识别人脸特征以及所述当前边缘盒子中的特征值库中的原始人脸特征之间的欧式距离;

[0186] 根据所述欧式距离,得到所述待识别人脸特征与所述原始人脸特征之间的相似度比对结果。

[0187] 在本公开的一种示例性实施例中,所述边缘盒子的优化装置还包括:

[0188] 第一判断模块,可以用于判断所述相似度比对结果是否大于预设阈值;

[0189] 第一结果上报模块,可以用于若是,则根据所述相似度比对结果生成识别成功的消息,并将所述识别成功的消息上报至所述中心服务器;

[0190] 待识别人脸特征广播模块,可以用于若否,则对所述待识别人脸特征进行广播发送给所述其他边缘盒子,并在其他边缘盒子中对所述待识别人脸特征进行识别,得到其他识别结果。

[0191] 在本公开的一种示例性实施例中,所述边缘盒子的优化装置还包括:

[0192] 第二结果上报模块,可以用于如果任一其他识别结果为识别成功,则将该识别成功的其他识别结果上报至所述中心服务器;

[0193] 原始人脸特征提取模块,可以用于提取与识别成功的其他识别结果对应的其他边缘盒子中的特征值库中与所述待识别人脸特征对应的原始人脸特征;

[0194] 特征值库更新模块,可以用于利用所述原始人脸特征对所述当前边缘盒子中的特征值库进行更新。

[0195] 在本公开的一种示例性实施例中,所述边缘盒子的优化装置还包括:

[0196] 待识别人脸特征上报模块,可以用于如果所有的其他识别结果均为识别失败,则将所述待识别人脸特征上报至所述中心服务器,以使得所述中心服务器对所述待识别人脸特征进行识别。

[0197] 上述边缘盒子的优化装置中各模块的具体细节已经在对应的边缘盒子的优化方法中进行了详细的描述,因此此处不再赘述。

[0198] 应当注意,尽管在上文详细描述中提及了用于动作执行的设备的若干模块或者单元,但是这种划分并非强制性的。实际上,根据本公开的实施方式,上文描述的两个或更多模块或者单元的特征和功能可以在一个模块或者单元中具体化。反之,上文描述的一个模

块或者单元的特征和功能可以进一步划分为由多个模块或者单元来具体化。

[0199] 此外,尽管在附图中以特定顺序描述了本公开中方法的各个步骤,但是,这并非要求或者暗示必须按照该特定顺序来执行这些步骤,或是必须执行全部所示的步骤才能实现期望的结果。附加的或备选的,可以省略某些步骤,将多个步骤合并为一个步骤执行,以及/或者将一个步骤分解为多个步骤执行等。

[0200] 在本公开的示例性实施例中,还提供了一种能够实现上述方法的电子设备。

[0201] 所属技术领域的技术人员能够理解,本公开的各个方面可以实现为系统、方法或程序产品。因此,本公开的各个方面可以具体实现为以下形式,即:完全的硬件实施方式、完全的软件实施方式(包括固件、微代码等),或硬件和软件方面结合的实施方式,这里可以统称为“电路”、“模块”或“系统”。

[0202] 下面参照图16来描述根据本公开的这种实施方式的电子设备1600。图16显示的电子设备1600仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0203] 如图16所示,电子设备1600以通用计算设备的形式表现。电子设备1600的组件可以包括但不限于:上述至少一个处理单元1610、上述至少一个存储单元1620、连接不同系统组件(包括存储单元1620和处理单元1610)的总线1630以及显示单元1640。

[0204] 其中,所述存储单元存储有程序代码,所述程序代码可以被所述处理单元1610执行,使得所述处理单元1610执行本说明书上述“示例性方法”部分中描述的根据本公开各种示例性实施方式的步骤。例如,所述处理单元1610可以执行如图2中所示的步骤S210:接收当前边缘盒子上报的当前负载数据;其中,所述当前负载数据包括所述当前边缘盒子中当前已经运行的视频路数以及CPU占用率;步骤S220:根据所述当前已经运行的视频路数以及CPU占用率,计算所述当前边缘盒子的当前负载率,并判断所述当前负载率是否大于与所述当前边缘盒子对应的其他边缘盒子的其他负载率;步骤S230:在确定所述当前负载率大于其他负载率且所述当前边缘盒子中出现新的待处理视频流时,为产生所述新的待处理视频流的摄像头分配新的绑定关系;步骤S240:基于所述新的绑定关系将产生所述新的待处理视频流的摄像头绑定至所述其他边缘盒子,以实现与所述第一边缘盒子以及其他边缘盒子的优化。

[0205] 所述处理单元1610还可以执行如图9中所示的步骤S910:根据当前边缘盒子与摄像头之间的绑定关系,确定所述当前边缘盒子所绑定的摄像头,并检测所述摄像头拍摄的当前视频流中是否出现包括目标对象的视频帧;步骤S920:在检测到所述当前视频流中出现包括目标对象的视频帧时,将所述当前视频流作为待处理视频流,并运行一视频路对所述待处理视频流进行处理;步骤S930:计算所述当前边缘盒子中当前已经运行的视频路数,并根据所述当前已经运行的视频路数以及处理所述待处理视频流产生的CPU占用率,生成当前负载数据;步骤S940:将所述当前负载数据上报至中心服务器,以使得所述中心服务器根据所述当前负载数据计算所述当前边缘盒子的当前负载率,并根据所述当前负载率对所述当前边缘盒子以及与所述当前边缘盒子对应的其他边缘盒子进行优化。

[0206] 存储单元1620可以包括易失性存储单元形式的可读介质,例如随机存取存储单元(RAM) 16201和/或高速缓存存储单元16202,还可以进一步包括只读存储单元(ROM) 16203。

[0207] 存储单元1620还可以包括具有一组(至少一个)程序模块16205的程序/实用工具16204,这样的程序模块16205包括但不限于:操作系统、一个或者多个应用程序、其它程序

模块以及程序数据,这些示例中的每一个或某种组合中可能包括网络环境的实现。

[0208] 总线1630可以为表示几类总线结构中的一种或多种,包括存储单元总线或者存储单元控制器、外围总线、图形加速端口、处理单元或者使用多种总线结构中的任意总线结构的局域总线。

[0209] 电子设备1600也可以与一个或多个外部设备1700(例如键盘、指向设备、蓝牙设备等)通信,还可与一个或者多个使得用户能与该电子设备1600交互的设备通信,和/或与使得该电子设备1600能与一个或多个其它计算设备进行通信的任何设备(例如路由器、调制解调器等等)通信。这种通信可以通过输入/输出(I/O)接口1650进行。并且,电子设备1600还可以通过网络适配器1660与一个或者多个网络(例如局域网(LAN),广域网(WAN)和/或公共网络,例如因特网)通信。如图所示,网络适配器1660通过总线1630与电子设备1600的其它模块通信。应当明白,尽管图中未示出,可以结合电子设备1600使用其它硬件和/或软件模块,包括但不限于:微代码、设备驱动器、冗余处理单元、外部磁盘驱动阵列、RAID系统、磁带驱动器以及数据备份存储系统等。

[0210] 通过以上的实施方式的描述,本领域的技术人员易于理解,这里描述的示例实施方式可以通过软件实现,也可以通过软件结合必要的硬件的方式来实现。因此,根据本公开实施方式的技术方案可以以软件产品的形式体现出来,该软件产品可以存储在一个非易失性存储介质(可以是CD-ROM,U盘,移动硬盘等)中或网络上,包括若干指令以使得一台计算设备(可以是个人计算机、服务器、终端装置、或者网络设备等)执行根据本公开实施方式的方法。

[0211] 在本公开的示例性实施例中,还提供了一种计算机可读存储介质,其上存储有能够实现本说明书上述方法的程序产品。在一些可能的实施方式中,本公开的各个方面还可以实现为一种程序产品的形式,其包括程序代码,当所述程序产品在终端设备上运行时,所述程序代码用于使所述终端设备执行本说明书上述“示例性方法”部分中描述的根据本公开各种示例性实施方式的步骤。

[0212] 根据本公开的实施方式的用于实现上述方法的程序产品,其可以采用便携式紧凑盘只读存储器(CD-ROM)并包括程序代码,并可以在终端设备,例如个人电脑上运行。然而,本公开的程序产品不限于此,在本文件中,可读存储介质可以是任何包含或存储程序的有形介质,该程序可以被指令执行系统、装置或者器件使用或者与其结合使用。

[0213] 所述程序产品可以采用一个或多个可读介质的任意组合。可读介质可以是可读信号介质或者可读存储介质。可读存储介质例如可以为但不限于电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者任意以上的组合。可读存储介质的更具体的例子(非穷举的列表)包括:具有一个或多个导线的电连接、便携式盘、硬盘、随机存取存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。

[0214] 计算机可读信号介质可以包括在基带中或者作为载波一部分传播的数据信号,其中承载了可读程序代码。这种传播的数据信号可以采用多种形式,包括但不限于电磁信号、光信号或上述的任意合适的组合。可读信号介质还可以是可读存储介质以外的任何可读介质,该可读介质可以发送、传播或者传输用于由指令执行系统、装置或者器件使用或者与其结合使用的程序。

[0215] 可读介质上包含的程序代码可以用任何适当的介质传输,包括但不限于无线、有线、光缆、RF等等,或者上述的任意合适的组合。

[0216] 可以以一种或多种程序设计语言的任意组合来编写用于执行本公开操作的程序代码,所述程序设计语言包括面向对象的程序设计语言—诸如Java、C++等,还包括常规的过程式程序设计语言—诸如“C”语言或类似的设计语言。程序代码可以完全地在用户计算设备上执行、部分地在用户设备上执行、作为一个独立的软件包执行、部分在用户计算设备上部分在远程计算设备上执行、或者完全在远程计算设备或服务器上执行。在涉及远程计算设备的情形中,远程计算设备可以通过任意种类的网络,包括局域网(LAN)或广域网(WAN),连接到用户计算设备,或者,可以连接到外部计算设备(例如利用因特网服务提供商来通过因特网连接)。

[0217] 此外,上述附图仅是根据本公开示例性实施例的方法所包括的处理的示意性说明,而不是限制目的。易于理解,上述附图所示的处理并不表明或限制这些处理的时间顺序。另外,也易于理解,这些处理可以是例如在多个模块中同步或异步执行的。

[0218] 本领域技术人员在考虑说明书及实践这里发明的发明后,将容易想到本公开的其他实施例。本申请旨在涵盖本公开的任何变型、用途或者适应性变化,这些变型、用途或者适应性变化遵循本公开的一般性原理并包括本公开未发明的本技术领域中的公知常识或惯用技术手段。说明书和实施例仅被视为示例性的,本公开的真正范围和精神由权利要求指出。

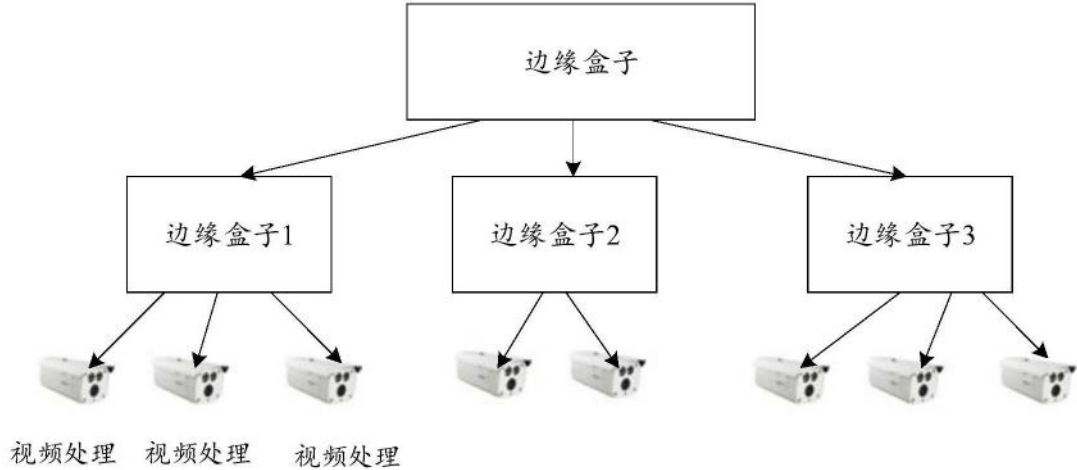


图1

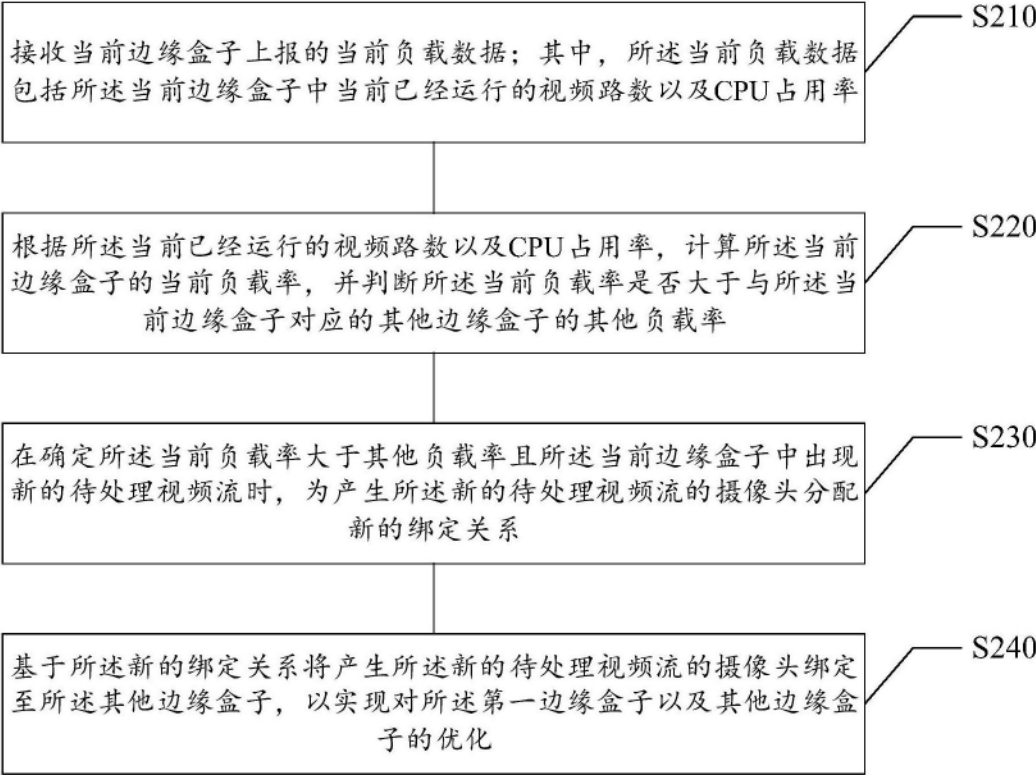


图2

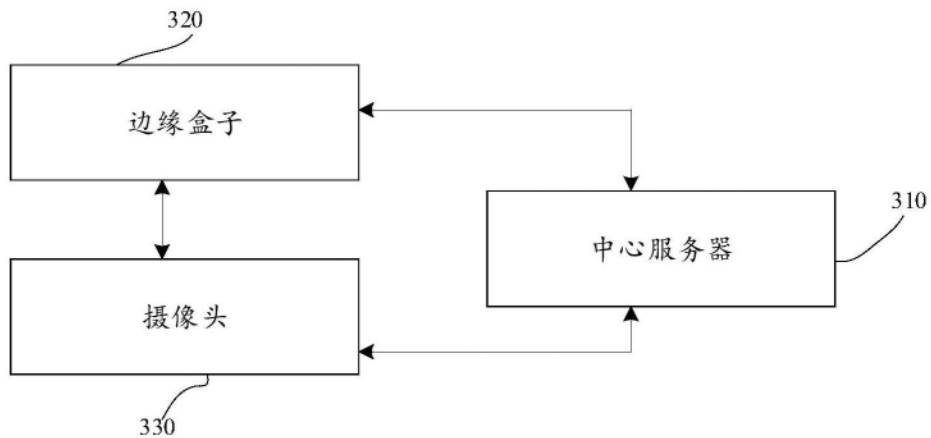


图3

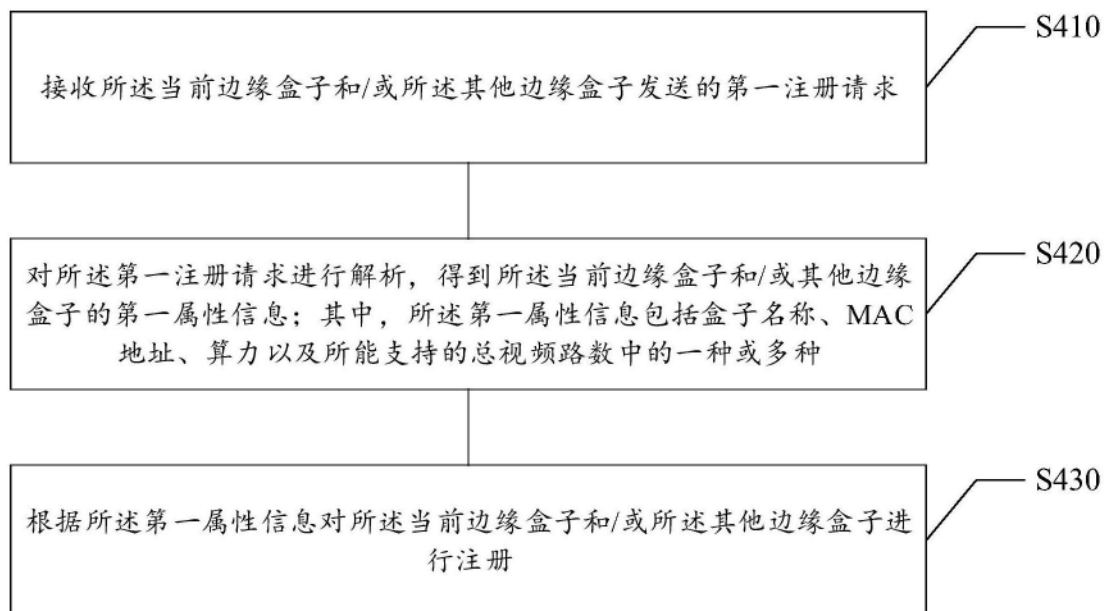


图4

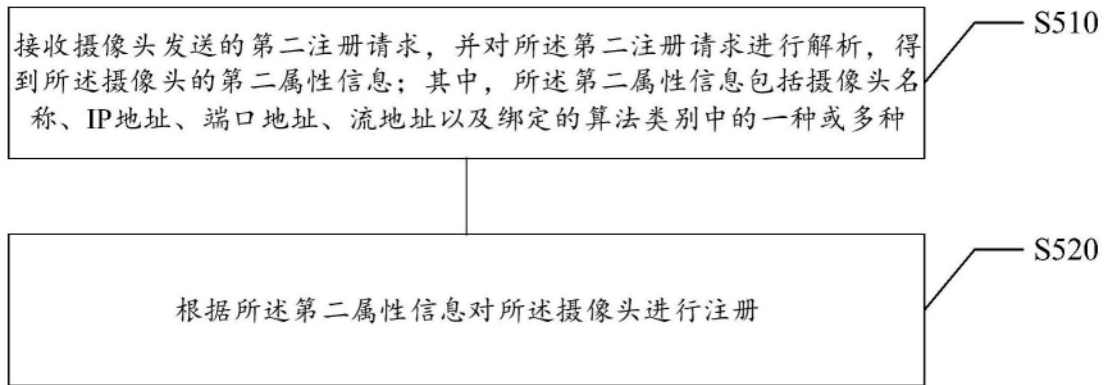


图5

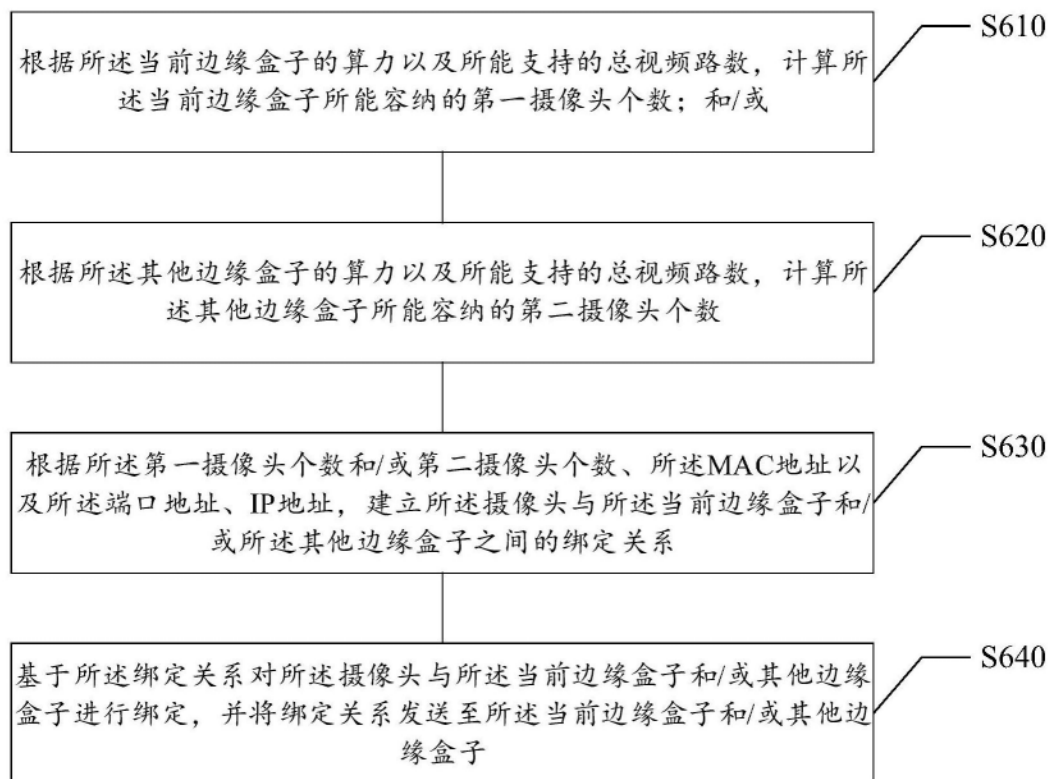


图6

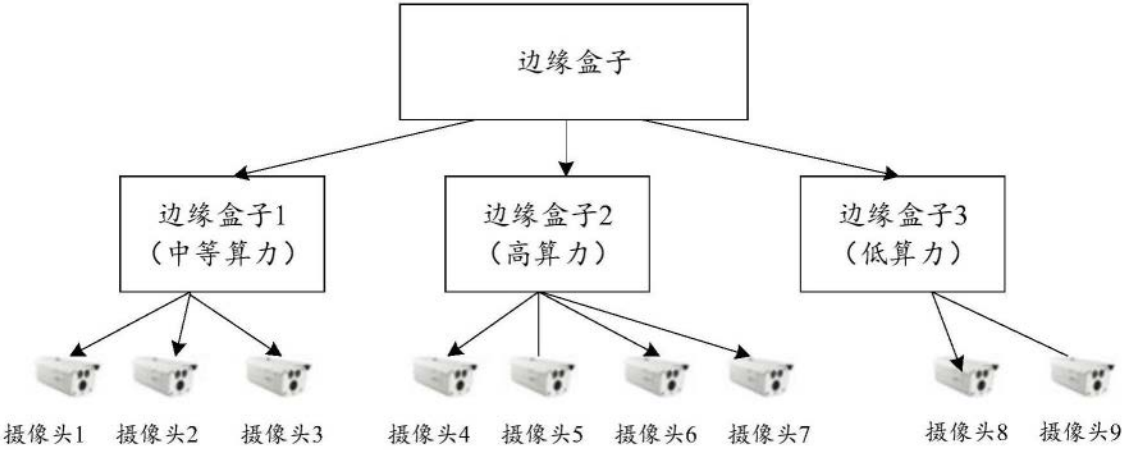


图7

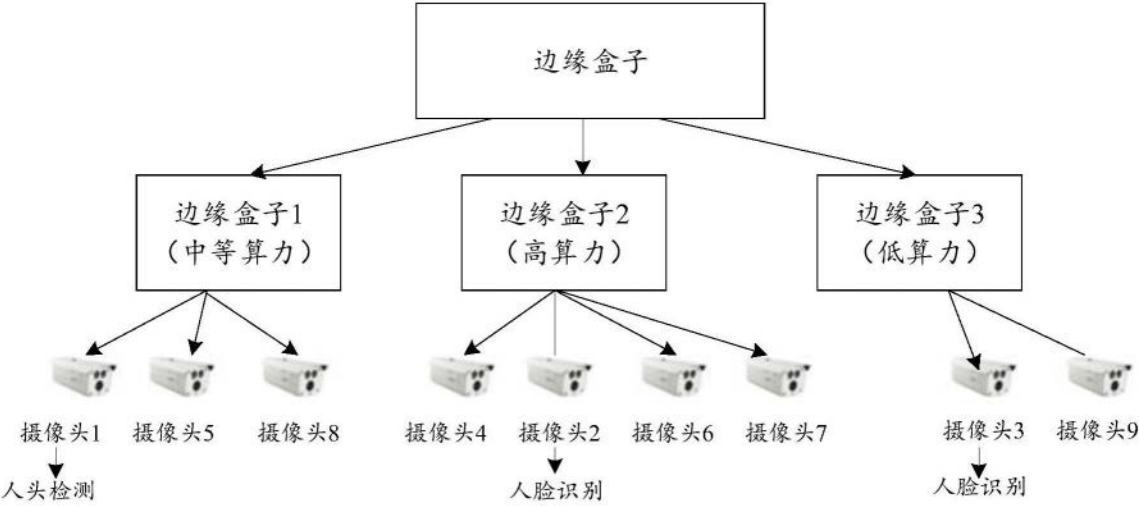


图8

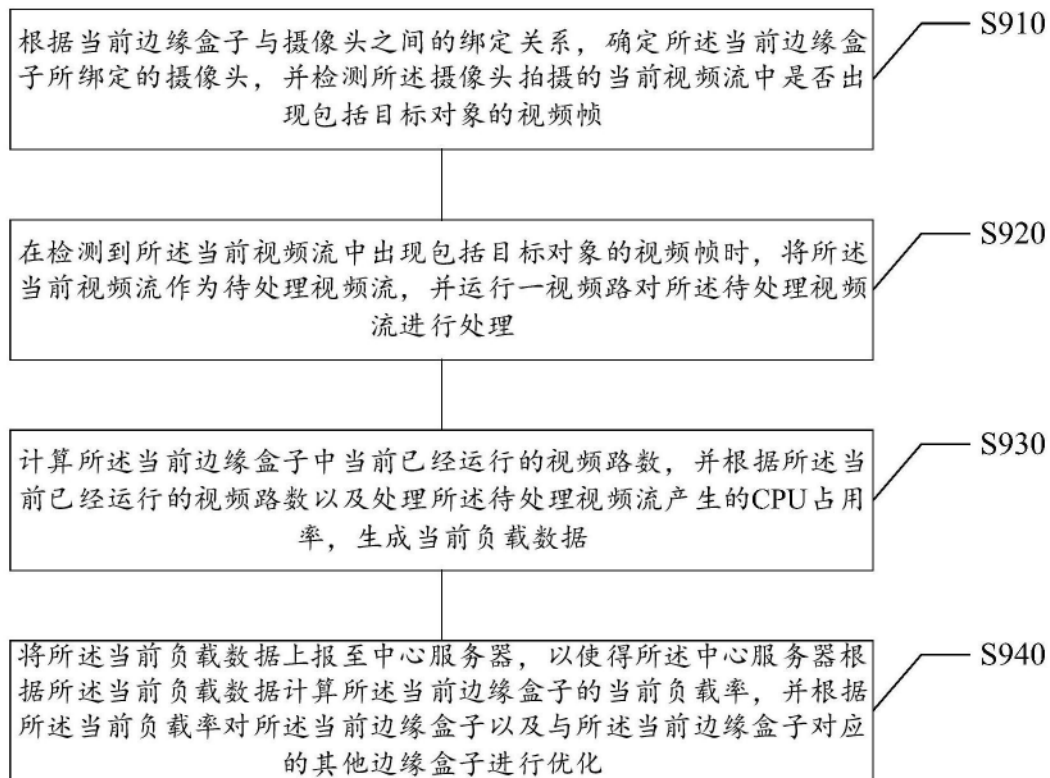


图9

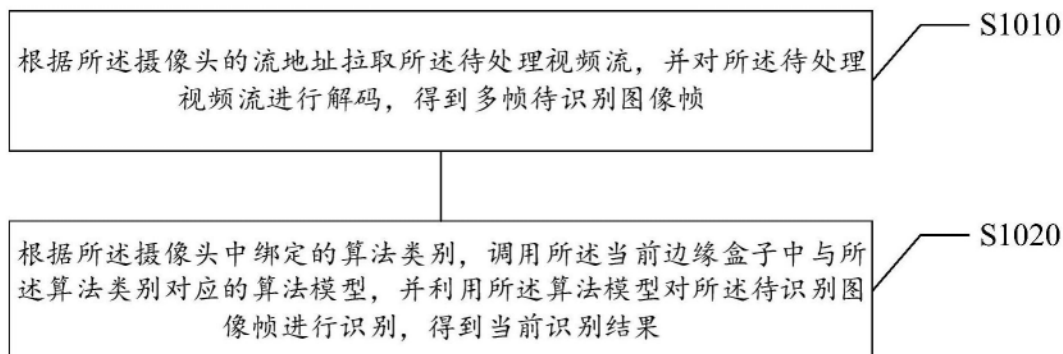


图10

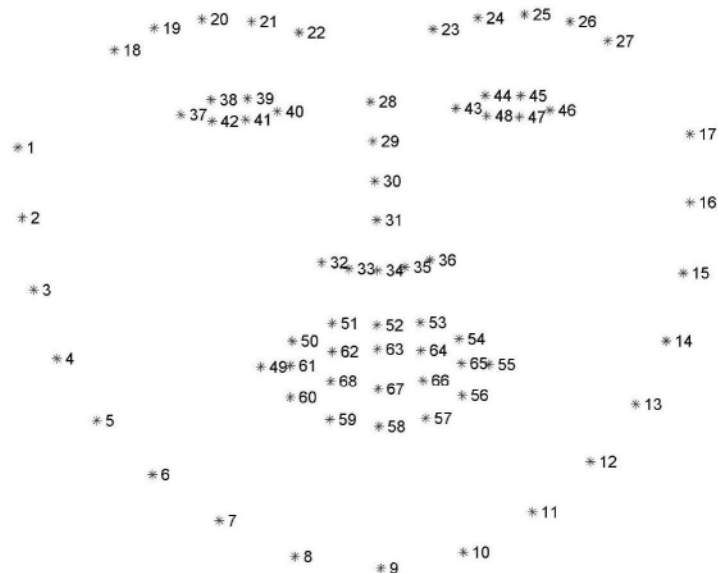


图11

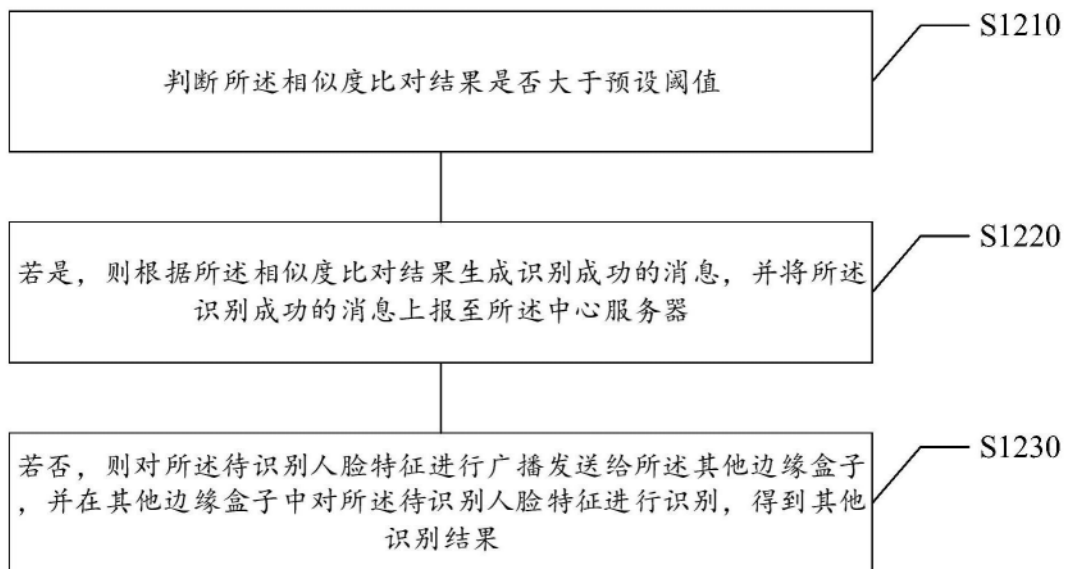


图12

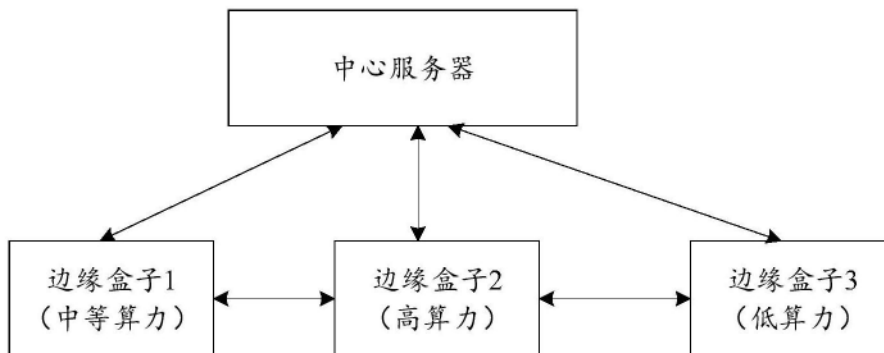


图13

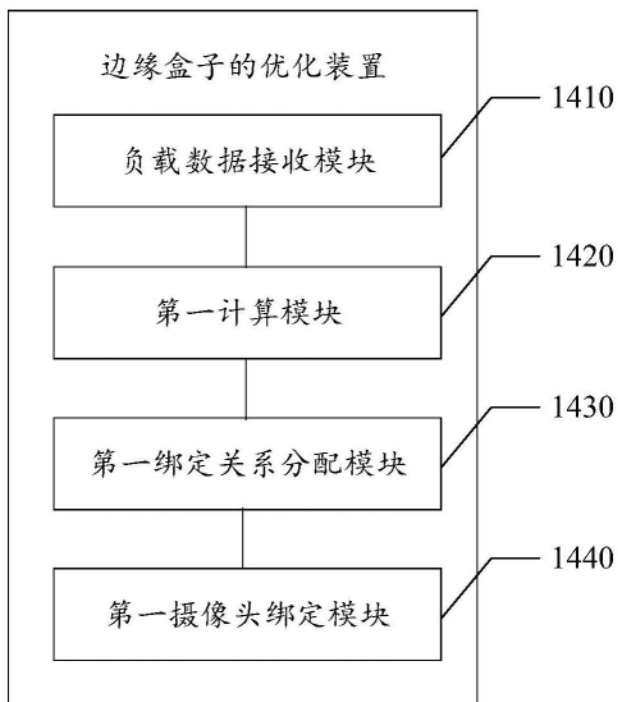


图14

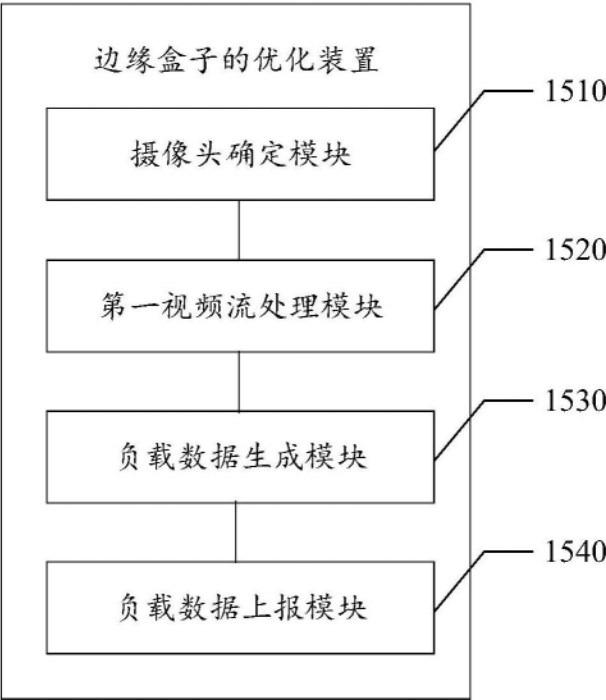


图15

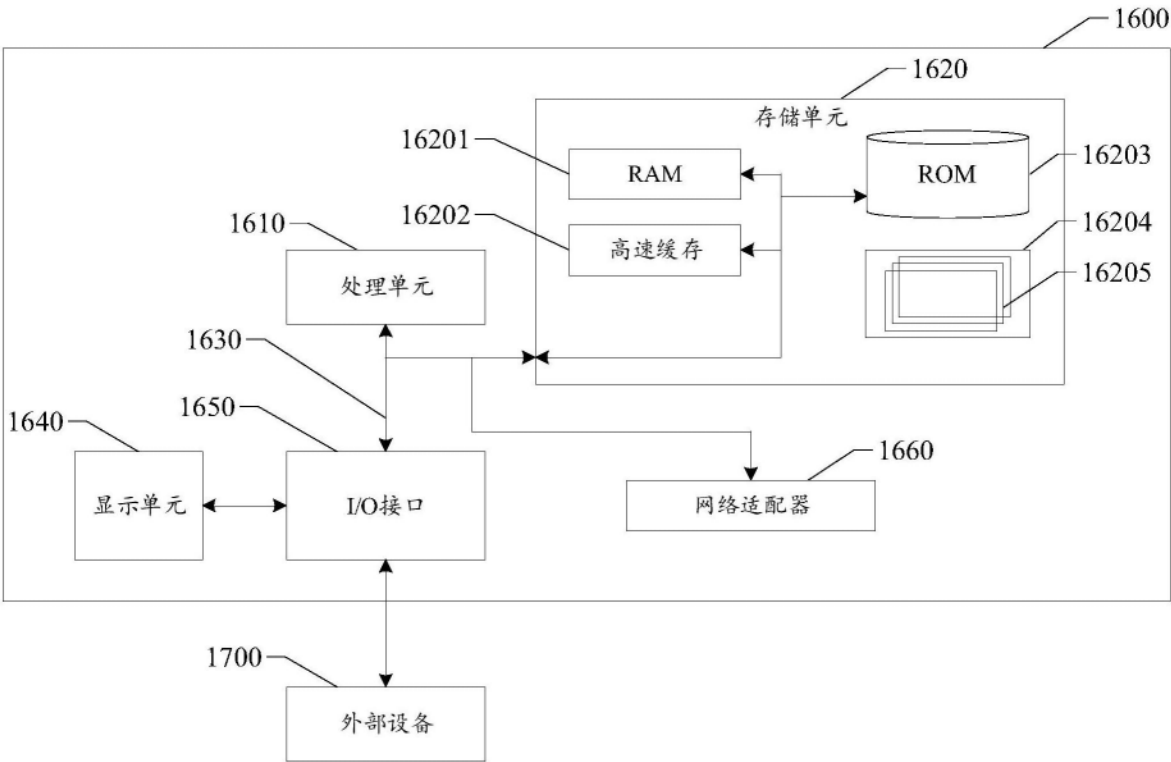


图16