

信息处理方法、装置、电子设备、存储介质及程序产品

本申请要求于 2023 年 04 月 18 日提交中国专利局、申请号为 202310458532.9 名称为“信息处理方法、装置、电子设备、存储介质及程序产品”的中国专利申请的优先权，其全部内容通过引用结合在本申请中。

5 技术领域

本申请涉及特征识别技术领域，具体而言，本申请涉及一种信息处理方法、装置、电子设备、存储介质及程序产品。

背景

10 在目前的特征识别设备上，能够提供特征识别能力及后续的特定服务，例如支付服务、签到服务等。使用者在该设备上录入待识别特征识别成功后，就能够打开设备接入方的服务程序，并进一步完成身份登录（例如扫码登录）等行为。

只要使用者能够成功登录服务程序，那么使用者与特征识别设备的后续一切交互都可以通过服务程序完成。

技术内容

15 根据本申请实施例的一个方面，提供了一种信息处理方法，由特征识别设备执行，所述特征识别设备中运行有第一应用和第二应用，并提供有第一应用框架，所述第一应用框架提供所述第一应用的运行环境，该方法包括：

第一应用框架响应于所述第一应用触发的预定接口调用请求，将基于预定接口调用请求生成的第一事件，插入到特征识别设备中的操作系统对应的内容提供器中；

20 第二应用接收到内容提供器中生成的第一事件插入的通知后，从内容提供器中确定第一事件的事件信息，并基于事件信息执行对特征数据的检测；

第二应用在检测到特征数据后，确定特征数据对应的对象身份信息，并将对象身份信息发送给第一应用。

25 根据本申请实施例的另一个方面，提供了一种信息处理装置，运行于特征识别设备，所述特征识别设备中运行有第一应用和第二应用，并提供有第一应用框架，所述第一应用框架提供所述第一应用的运行环境，该装置包括：

第一插入模块，用于响应于触发的预定接口调用请求，将基于预定接口调用请求生成的第一事件插入到特征识别设备中的操作系统对应的内容提供器中；

30 检测模块，用于接收到内容提供器中生成的第一事件插入的通知后，从内容提供器中确定第一事件的事件信息，并基于事件信息执行对特征数据的检测；

发送模块，用于在检测到特征数据后，确定特征数据对应的对象身份信息，并将对象身份信息发送给第一应用。

附图说明

35 为了更清楚地说明本申请实施例中的技术方案，下面将对本申请实施例描述中所需要使用的附图作简单地介绍。

- 图 1a 为在刷掌设备中使用设备接入方的 IoT 小程序的示意图；
- 图 1b 为扫码方式登录 IoT 小程序的示意图；
- 图 2a 为本申请实施例提供的一种信息处理方法的流程示意图；
- 图 2b 为本申请实施例提供的一种信息处理方法的流程示意图；

- 图 3a 为本申请实施例提供的 ContentProvider 实现的一种示意图；
图 3b 为本申请实施例提供的 ContentProvider 实现的另一种示意图；
图 4 为本申请实施例提供的一种身份信息的数据传输效果的示意图；
图 5 为本申请实施例提供的 URI 的结构标准的示意图；
5 图 6 为本申请实施例提供的一种从运行在 WMPF 中的 IoT 小程序到刷掌 APP 的调用的完整流程的示意图；
图 7 为本申请实施例提供的一种信息处理装置的结构示意图；
图 8 为本申请实施例提供的一种电子设备的结构示意图。

实施方式

- 10 下面结合本申请中的附图描述本申请的实施例。应理解，下面结合附图所阐述的实施方式，是用于解释本申请实施例的技术方案的示例性描述，对本申请实施例的技术方案不构成限制。

- 本技术领域技术人员可以理解，除非特意声明，这里使用的单数形式“一”、“一个”和“该”也可包括复数形式。应该进一步理解的是，本申请实施例所使用的术语“包括”以及“包含”是指相应特征可以实现为所呈现的特征、信息、数据、步骤、操作、元件和/或组件，但不排除实现为
15 本技术领域所支持其他特征、信息、数据、步骤、操作、元件、组件和/或它们的组合等。应该理解，当我们称一个元件被“连接”或“耦接”到另一元件时，该一个元件可以直接连接或耦接到另一元件，也可以指该一个元件和另一元件通过中间元件建立连接关系。此外，这里使用的“连接”或“耦接”可以包括无线连接或无线耦接。这里使用的术语“和/或”指示该术语所限定的项目中的至少一个，例如“A 和/或 B”可以实现为“A”，或者实现为“B”，或者实现为“A 和 B”。

- 20 为使本申请的目的、技术方案和优点更加清楚，下面将结合附图对本申请实施方式作进一步地详细描述。

首先对本申请涉及的几个名词进行介绍和解释：

- (1) 刷掌：在无接触的情况下，使用者只需伸出手掌，即可让识别设备进行识别，验证使用者的身份。其中，刷掌技术可以依靠掌纹识别、静脉识别等，但不限于此。

- 25 (2) 特征识别设备：即线下的特征识别终端（如刷掌设备、刷脸设备、刷指纹设备等），使用者可在此设备上完成特征识别（如手掌识别、人脸识别、指纹识别等）的流程，并进行后续的特定操作，例如支付操作、签到操作等，但不限于此。

- (3) 特征识别服务：使用者在特征识别设备上开通特征识别服务后，会绑定到使用者指定且唯一的账号上。

- 30 (4) 设备接入方：即特征识别服务的第三方接入方，比如大型商超、企业等。设备接入方会基于特征识别设备提供的服务，开发运行在特征识别设备上的服务程序，也可以被认为是服务程序开发者。

- (5) IoT（Internet of Things，物联网）小程序：能够在脱离预定应用环境的硬件设备中运行的小程序。

- 35 (6) WMPF（Mini-Program Framework，小程序框架）：即 IoT 小程序硬件框架，是用于承载并运行 IoT 小程序的框架。为解决在非通用型计算设备上使用小程序的诉求，小程序硬件框架应运而生。WMPF 作为一个运行环境，能让硬件设备（非通用型计算设备例如特征识别设备）在缺乏条件运行预定应用客户端例如微信客户端的情况下运行小程序。

- 目前的特征识别设备中，设备接入方可以使用服务程序来承载自身的业务，并使用特征识别设备的特征识别相关能力。例如以特征识别设备是刷掌设备，设备接入方开发的服务程序是 IoT 小程序为例，在刷掌设备中使用设备接入方的 IoT 小程序来承载其业务的整体应用模式大致可以描述为
40 如图 1a 所示的流程：

- 在刷掌设备上，以刷掌设备的提供方开发的刷掌 APP（Application，应用或应用程序）作为起点，它提供了核心的刷掌识别能力和支付服务（也可以是其他服务，该示例中以支付服务为例进行介绍）。在一些实施例中，刷掌 APP 可以以 Android 应用程序包（Android application package，APK）的形式运行，也可以以 Android 应用程序存档文件（Android Archive，AAR）的形式集成。
45 用户在该刷掌设备上刷掌并经过刷掌 APP 识别掌纹成功后，就能进一步启动设备接入方的业务小程序应用。在一些实施例中，设备接入方的业务小程序应用可以以 IoT 小程序的形式运行。刷掌 APP

可以先启动 WMPF, WMPF 再启动设备接入方的 IoT 小程序。其中, WMPF 是 IoT 小程序的运行环境, 可以以 APK 的形式运行。之后, 用户便能够在刷掌之后, 在刷掌设备上通过打开的小程序, 完成登录等行为。在一些实施例中, 可以采用如图 1b 所示的扫码方式登录, 即扫描二维码登录, 刷掌设备会申请使用该使用者的身份打开登录小程序, 同意登录后即可使用该刷掌设备上的小程序; 或者也可以采用其他方式登录, 例如输入账号、密码、验证码等。

以常规的情况看, 这样的链路是可行的。只要使用者能够成功打开小程序, 那么使用者的后续一切操作和交互都可以在小程序中完成。但是, 由于特征识别过程和服务程序登录过程是割裂的, 使用者录入待识别特征后还要再次完成身份登录, 操作繁琐, 影响使用者体验, 因此这种登录小程序的方式有以下两个缺陷:

(1) 小程序无法获得刷掌者的身份。作为示例地, 使用者很有可能多个账号, 而刷掌服务绑定了其中一个账号, 小程序没有办法保障使用者一定是使用绑定了刷掌服务的账号进行登录。这样可能导致一个关键的问题: 小程序没有办法保障刷掌者的账号, 和刷掌后继续使用小程序的账号是同一个账号, 也就没有办法进一步提供可靠的服务。

(2) 使用者刷掌后还要再次扫码完成身份登录, 操作繁琐, 影响使用者体验。

本申请提供的信息处理方法、装置、电子设备、计算机可读存储介质以及计算机程序产品, 如图 2a 所示, 该信息处理方法包括: 步骤 S21, 第一应用框架响应于在第一应用中触发的预定接口调用请求, 将基于所述预定接口调用请求生成的第一事件, 插入到所述特征识别设备中的操作系统对应的内容提供者中; 步骤 S22, 第二应用接收到所述内容提供者中生成的所述第一事件插入的通知后, 从所述内容提供者中确定所述第一事件的事件信息, 并基于所述事件信息执行对特征数据的检测; 步骤 S23, 所述第二应用在检测到所述特征数据后, 确定所述特征数据对应的对象身份信息, 并将所述对象身份信息发送给所述第一应用。通过该信息处理方法可以解决现有技术的如上技术问题。

下面通过对几个示例性实施方式的描述, 对本申请实施例的技术方案以及本申请的技术方案产生的技术效果进行说明。需要指出的是, 下述实施方式之间可以相互参考、借鉴或结合, 对于不同实施方式中相同的术语、相似的特征以及相似的实施步骤等, 不再重复描述。

本申请实施例中提供了一种信息处理方法, 由特征识别设备执行, 所述特征识别设备中运行有第一应用和第二应用, 并提供有第一应用框架, 所述第一应用框架提供所述第一应用的运行环境, 如图 2b 所示, 该方法包括:

步骤 S201: 第一应用框架响应于第一应用触发的预定接口调用请求, 将基于预定接口调用请求生成的第一事件, 插入到特征识别设备中的操作系统对应的内容提供者中;

对于本申请实施例中, 第一应用是指承载设备接入方业务的服务程序, 例如可以是 IoT 小程序, 或者也可以是 APP 或网页等, 但不限于此。

第一应用框架是承载并运行第一应用的框架, 例如第一应用是 IoT 小程序, 则第一应用框架可以是 WMPF, 但不限于此。

本申请实施例中, 使用者在特征识别设备上录入特征数据后, 例如使用者在刷掌设备上刷掌后, 会由在特征识别设备中运行的第一应用触发所述预定接口调用请求, 并将该预定接口调用请求传递到第一应用框架中。

第一应用框架响应于该预定接口调用请求, 会基于预定接口调用请求生成第一事件, 并将第一事件插入到特征识别设备中的操作系统对应的内容提供者中。具体地, 第一应用框架会将第一应用对预定接口的调用封装成事件信息, 以得到第一事件, 再该第一时间插入 (insert) 到内容提供者中。其中, 事件信息封装的结构大致分为三个部份, 事件名称、事件唯一标识和事件参数。事件名称是基于预定接口调用请求确定的, 指示该第一事件是什么事件。事件唯一标识指示第一事件与当前录入的特征数据相关的信息, 用于区分在同一特征识别设备不同时间录入的特征数据。事件参数指示第一事假的后续处理所需的参数。

其中, 内容提供者能够提供统一的访问应用数据的接口, 供其他应用程序进行调用, 允许把自己的应用数据根据需求开放给其他应用程序进行查找、插入、更新或删除, 而不用担心因为直接开放数据库权限而带来的安全问题。

以特征识别设备是搭载 Android 系统的硬件设备为例, 特征识别设备中的操作系统对应的内容提供者可以是指 Android 系统的四大组件之一的 ContentProvider。具体地, 将 Android 中特有的 ContentProvider 作为底层桥梁, 通过与第一应用框架约定的预定接口 (用于第一应用获取特征数据对应的对象身份信息) 协议, 实现设备接入方的第一应用从特征识别服务侧获取对象身份信息的

能力。

步骤 S202: 第二应用接收到内容提供器中有第一事件插入的通知后, 从内容提供器中确定第一事件的事件信息, 并基于事件信息执行对特征数据的检测;

5 本申请实施例中, 第一应用框架将第一事件插入到内容提供器中后, 内容提供器会生成一个第一事件插入的通知, 并将该通知发送到第二应用, 以通知第二应用插入了一个第一事件。

第二应用接收到内容提供器的第一事件插入的通知后, 可以根据插入的第一事件, 从内容提供器中提取出第一事件的事件信息, 并根据事件信息中的事件唯一标识, 确定与该事件唯一标识对应的特征数据, 并开启对使用者录入的特征数据的检测。所述特征数据可以包括: 刷掌数据例如掌纹数据和/或静脉数据等, 脸部特征数据, 或者指纹数据等等。

10 步骤 S203: 第二应用在检测到特征数据后, 确定特征数据对应的对象身份信息, 并将对象身份信息发送给第一应用。

本申请一些实施例中, 当使用者录入特征数据后, 第二应用便可检测到特征数据。在检测到特征数据后, 第二应用会确定特征数据对应的对象身份信息, 获取了当前录入特征数据的使用者的对象身份信息后, 即表示当前使用者特征识别成功。

15 对于本申请实施例, 获取的对象身份信息可以具体是指当前使用者在特征识别服务下的身份ID (Identity or Identifier, 标识)。在一些实施例中, 若特征识别服务是与预定应用相关联的, 对象身份信息可以是预定软件的使用者在该特征识别服务产品下的 OpenID。实际应用中, 预定软件的使用者访问不同的小程序、移动应用、网站应用、小商店等都会有唯一的 OpenID, 但同一个使用者访问不同的产品生成的 OpenID 是不一样的。因此, 对于特征识别服务, 每个使用者的 OpenID 也都是唯一的。

20 本申请实施例中, 第二应用获取了录入特征数据的使用者的对象身份信息后, 将对象身份信息发送给第一应用, 以使用该对象身份信息登录第一应用。

即对于第一应用而言, 仅需要调用预定接口 (即触发预定接口调用请求), 即可主动获取来自特征识别服务侧识别的当前特征数据的对象身份信息, 不再依赖使用者再次手动扫码来完成登录。

25 通过本申请实施例提供的信息处理方法登录第一应用, 可以克服现有技术的缺陷, 实现以下两个技术效果:

(1) 针对第一应用而言, 能够通过这个预定接口准确地获得录入特征数据的使用者的对象身份信息, 保障录入特征数据的使用者的对象身份信息, 和后续使用第一应用的对象身份信息是同一个账号, 可以避免使用者个人信息的泄漏, 进而保证了个人信息的安全, 同时避免了由于获取用户的身份信息错误而造成的刷掌失败, 进而可以使第一应用进一步提供更可靠的服务。

30 (2) 针对使用者而言, 省去了使用者录入特征数据后还要再次进行身份登录的操作, 大幅简化使用者的操作流程, 优化使用者的使用体验。

本申请实施例中, 为了能够达到让第一应用直接主动获取录入特征数据的使用者的对象身份信息目的, 提供了一种第一应用和提供特征识别服务的第二应用之间的通信桥梁, 即特征识别设备中的操作系统对应的内容提供器。

35 对于基于 Android 系统的特征识别设备, Android 中的 ContentProvider 也能够很好地应对这种“跨 APP”的“数据共享”的诉求, 因此, 在基于 Android 系统的特征识别设备中使用 ContentProvider 来作为这个通道, 具备较强的可行性。

下面以 Android 系统中的 ContentProvider 为例, 介绍通过本申请实施例实现该目的的可行的实施方式。

40 ContentProvider 是 Android 中的一种组件, 用于共享数据。它可以让一个应用程序提供给其他应用程序访问其数据的接口, 实现跨应用共享数据。

ContentProvider 的实现步骤如下:

1. 继承 ContentProvider 类并实现其中的抽象方法。
- 45 2. 在 Android 系统清单文件中注册 ContentProvider。
3. 在 ContentProvider 的实现中, 应用程序通过 URI (Uniform Resource Identifier, 统一资源标识符) 匹配来识别哪些请求应该由该 ContentProvider 来处理。

其中, URI 是一种统一资源标识符, 用于标识 ContentProvider 提供的数据库。

4. 在 ContentProvider 的实现中, 应用程序可以实现 query()、insert()、update() 和 delete() 等方法。

50 其中, 这些方法用于处理不同的请求类型, 比如对于数据库 (Database) 的查找、插入、更新、

删除等操作,如图 3a 所示。

5. 在 ContentProvider 的实现中,通过特定的数据对象(ContentValues)来封装数据并返回给请求方。

6. 在 ContentProvider 的实现中,应用程序可以通过特定的访问对象(ContentResolver)来访问 ContentProvider 中的数据。

其中,ContentResolver 是一个 Android 系统提供的类,它用于与 ContentProvider 进行交互,可以通过它的方法来控制进行查询、插入、更新和删除等操作,如图 3b 所示。

通过 ContentProvider 实现数据共享可以实现跨应用程序的数据共享,同时保证了数据的安全性和隐私性。同时,通过 ContentProvider 可以使应用程序的数据更加容易被其他应用程序访问和利用,增强了应用程序的可扩展性。

本申请实施例中,对于步骤 S201,第一应用框架对内容提供者使用了 insert() 操作,将第一事件插入到内容提供者中,实现对数据库中的对象身份信息的查询效果。由于对象身份信息不是直接在数据库中查询的,而是在第二应用通过特征识别结果查询得到的,因此插入操作可以触发通知第二应用执行相应的操作,从而实现将查询得到的对象身份信息发送给第一应用(步骤 S203)的目的。

本申请实施例中,步骤 S203 中的“将对象身份信息发送给第一应用”具体可以包括:

步骤 S2031:第二应用向内容提供者插入基于对象身份信息生成的第二事件;

本申请实施例中,在当前使用者的特征数据识别成功,获取了其对应的对象身份信息后,第二应用便会基于对象身份信息生成的第二事件,并反向地在内容提供者中插入第二事件。

步骤 S2032:第一应用框架接收到内容提供者的第二事件的插入通知后,从第二事件中提取对象身份信息,并将对象身份信息传递给第一应用。

本申请实施例中,第二应用将第二事件插入到内容提供者中后,内容提供者会生成一个第二事件插入的通知,并将该通知发送到第一应用框架,以通知到第一应用框架插入了一个第二事件。

第一应用框架接收到内容提供者的第二事件的插入通知后,便可以根据内容提供者中插入的第二事件,提取出对象身份信息,并将获得的对象身份信息,传递给第一应用,以使用该对象身份信息登录第一应用。

本申请实施例中,对于步骤 S2031,第二应用对内容提供者使用了 insert() 操作,将第二事件插入到内容提供者中,实现对数据库中查询到的对象身份信息的数据传输效果,如图 4 所示。由于对象身份信息是第一应用框架被动接收的,因此插入操作可以用于触发通知第一应用框架执行相应的操作,从而实现接收对象身份信息并传递给第一应用(步骤 S2032)的目的。

本申请实施例中,为第二应用执行步骤 S203 提供了一种实现方式,具体可以包括:第二应用在检测到特征数据后,确定特征数据对应的对象身份信息,并回调第一应用,以将对象身份信息发送给第一应用。

在本申请实施例中,第一应用可以提供一个回调接口,在第一应用发起预定接口调用请求来调用预定接口后,该回调接口也会被调用。具体而言,在第二应用将对象身份信息发送给第一应用时,可以通过第一应用提供的回调接口进行回调(callback)第一应用,将对象身份信息发送给第一应用。

结合上述至少一个实施例,对象身份信息是经过第一应用框架传递给第一应用的。则步骤 S2032 可以包括:第一应用框架接收到内容提供者中有第二事件插入的通知后,从第二事件中提取对象身份信息,并通过回调接口回调第一应用,将对象身份信息传递给第一应用。

在本申请实施例中,在第一应用框架约定好的预定接口被调用后,也会调用第一应用的回调接口。具体而言,在第一应用框架从第二事件中提取出对象身份信息后,可以通过第一应用提供的回调接口进行回调(callback)第一应用,将对象身份信息发送给第一应用。

本申请实施例中,为步骤 S201 提供了一种可行的实施方式,具体地,可以包括步骤:第一应用框架将第一应用对预定接口的调用封装成事件信息,得到第一事件;第一应用框架将第一事件插入到特征识别设备中的操作系统对应的内容提供者中;其中,事件信息包括事件名称、事件唯一标识和事件参数中的至少一项。

在本申请实施例中,第一应用框架会将第一应用对预定接口的调用封装为一个第一事件,再插入到内容提供者中。其中,事件信息封装的结构大致分为三个部份,事件名称、事件唯一标识和事件参数。

具体地,事件名称是基于预定接口调用请求确定的,指示该第一事件是什么事件。内容提供者

和第二应用可以通过事件名称的匹配来确定对该事件对应的处理方式,即内容提供器通知到第二应用有第一事件插入,第二应用开启对特征数据的检测。一个示例中,由于该第一事件是基于第一应用的请求生成的,用于第一应用直接获取录入特征数据的使用者的对象身份信息,该第一事件的事件名称可以为 GetOpenID。

5 事件唯一标识指示第一事件与当前录入的特征数据相关的信息,用于区分在同一特征识别设备不同时间录入的特征数据。在一些实施例中,事件唯一标识可以是基于以下至少一项确定的:接收到预定接口调用请求的时间戳、生成第一事件的时间戳等。在一些实施例中,可以直接将一个或多个时间戳作为事件唯一标识,也可以是将一个或多个时间戳进行一定的运算或改编后作为事件唯一标识。

10 事件参数指示第一事件的后续处理所需的参数,实际应用中,本领域技术人员可以根据实际情况对具体的参数类型和内容进行设置,本申请实施例在此不做限定。

一个示例中,基于这些第一事件的事件信息,第一应用框架将第一事件插入内容提供器的结构可以如下所示:

事件名称: GetOpenID

15 事件唯一标识: 当前时间戳

事件参数: 可选

本申请实施例中,步骤 S2031 具体可以包括步骤:第二应用将第一事件的事件信息和对象身份信息进行封装,生成第二事件;第二应用向内容提供器插入第二事件;其中,第一事件的事件信息包括事件名称和事件唯一标识中的至少一项。

20 在本申请实施例中,第二应用会将第一事件的事件信息和对象身份信息封装为一个第二事件,再反向插入第二事件到内容提供器中。其中,第二事件的事件名称可以直接采用第一事件的事件名称,或者,第二事件的事件名称与第一事件的事件名称不同,第二事件的事件信息中可以同时包括第一事件的事件名称和第二事件的事件名称。

同理地,第二事件的事件唯一标识可以直接采用第一事件的事件唯一标识,或者,第二事件的事件唯一标识与第一事件的事件唯一标识不同,第二事件的事件信息中可以同时包括第一事件的事件唯一标识和第二事件的事件唯一标识。在一些实施例中,第二事件的事件唯一标识是基于以下至少一项确定的:接收到预定接口调用请求的时间戳、生成第一事件的时间戳、开始检测特征数据的时间戳、检测到特征数据的时间戳、生成第一事件的时间戳等。在一些实施例中,可以直接将一个或多个时间戳作为第二事件的事件唯一标识,也可以是将一个或多个时间戳进行一定的运算或改编后作为第二事件的事件唯一标识,也可以将第一事件的事件唯一标识和第二事件的事件唯一标识一起进行一定的运算或改编后作为第二事件的事件唯一标识,本申请实施例在此不做限定。

30 一个示例中,基于这些事件信息和获取的对象身份信息,第二应用将第二事件插入内容提供器的结构可以如下所示:

事件名称: GetOpenID

35 事件唯一标识: 检测时所生成的时间戳

事件参数: 使用者的 OpenID (对象身份信息)

对于本申请实施例,通过内容提供器实现了一种高效、安全的通道,这种通道能够让第一应用框架主动获取第二应用的部分数据。

40 本申请实施例中,可以通过 URI 来标识插入内容提供器的事件,内容提供器能够根据 URI 匹配并识别出各事件及其对应的处理方式。

即本申请实施例中,对于步骤 S201,还可以包括步骤:第一应用框架通过第一统一资源标识符(第一 URI)对插入内容提供器的第一事件进行标识,第一统一资源标识符(第一 URI)用于内容提供器识别第一事件,并向第二应用发送内容提供器中有第一事件插入的通知。

同理地,对于步骤 S2031,还可以包括步骤:第二应用通过第二统一资源标识符(第二 URI)对插入内容提供器的第二事件进行标识,第二统一资源标识符(第二 URI)用于内容提供器识别第二事件,并向第一应用框架发送有第二事件插入的通知。

其中,统一资源标识符(URI)又被称为通用资源标志符。

以特征识别设备搭载 Android 系统为例,Android 系统遵循 URI 的标准,定义了一套专用的 URI (即: Content URIs)。其结构标准如图 5 所示。其中:

50 主题名 (Schema): ContentProvider 的 URI 前缀主题名为 Content。

授权信息 (Authority): ContentProvider 授权的唯一标识符,例如结合第一应用框架 (以

WMPF 为例)及两者之间的处理过程,上述第一 URI 的授权信息可以是 com.wmpf.cli.provider,上述第二 URI 的授权信息可以是 com.wmpf.comm.provider。

- 表名(Path): ContentProvider 指向数据库中的某个表名,例如结合第一应用框架(以 WMPF 为例)及两者之间的处理过程,上述第一 URI 指向的表名可以是 invokeChannelEvent(注册检测事件),上述第二 URI 指向的表名可以是 notifyInvokeChannelEvent(反向通知事件发生)。

记录(ID): 表中的某个记录(若无指定,则返回全部记录)

以 ContentProvider 功能作为基础,结合第一应用框架(以 WMPF 为例),可预先确定的标识第一事件和标识第二事件的标准 URI 结构分别为:

content://com.wmpf.cli.provider/invokeChannelEvent

- content://com.wmpf.comm.provider/notifyInvokeChannelEvent

本申请实施例中,特征识别设备可以为设备接入方提供一些功能,包括但不限于:

(1) 事件注册功能

第一应用还具有事件注册功能,注册一个第三事件,能够发起预定接口调用请求,以及接收通过回调接口发送的对象身份信息。

- 一个示例中,该功能的伪代码如下:

//注册一个 event(第三事件)名为'GetOpenID'的事件

```
wmpf.Channel.registerEvent({
```

```
  event:'GetOpenID',
```

```
  success(res) {
```

- ```
 console.log(res)
```

```
 }
```

```
})
```

```
const callback=(res) => {
```

- ```
  //第二应用会在获取到录入特征数据的使用者的 OpenID 后触发'GetOpenID' event 第三事件,向第一应用发送 OpenID
```

```
  console.log('OpenID =' + res.data)
```

```
}
```

则本申请实施例中,在特征识别设备中运行的第一应用触发预定接口调用请求之前,还可以包括步骤:针对第一应用注册第三事件。

- (2) 开启检测功能

第一应用还可以具有开启检测功能,基于该功能,可以将上述第三事件置为检测状态,检测已注册的第三事件。只有第三事件处于检测状态,才能接收到第二应用发送的对象身份信息。

一个示例中,该功能的伪代码如下:

//检测已注册过的 GetOpenID 事件

- ```
wmpf.Channel.on('GetOpenID',callback)
```

则本申请实施例中,步骤 S203 具体包括步骤:若第三事件处于检测状态,第二应用在检测到特征数据后,确定特征数据对应的对象身份信息,触发第三事件,将对象身份信息发送给第一应用。

即第二应用仅在第三事件处于检测状态情况下,检测特征数据后,确定特征数据对应的对象身份信息,以及将对象身份信息发送给第一应用,若第三事件处于非检测状态,第二应用可以不执行这些步骤,以节约计算资源。

- (3) 取消检测功能

第一应用还可以具有取消检测功能,基于该功能,可以将上述第三事件置为非检测状态,不检测已注册的第三事件。第三事件处于非检测状态,无法接收到第二应用发送的对象身份信息。

一个示例中,该功能的伪代码如下:

- ```
//取消检测已注册过的'GetOpenID'事件
```

```
wmpf.Channel.off(GetOpenID,callback)
```

则本申请实施例提供的方法还可以包括步骤:针对第一应用将第三事件变更为取消检测状态。

可以理解,开启检测功能和取消检测功能可以配合使用,设备接入方可以自定义开启检测功能和取消检测功能的生效时间,例如每有一个使用者来使用一次特征识别设备就开启一次检测,使用完毕后就取消这次检测;又例如,在预定时间段的起始时间和结束时间分别开启检测和取消检测,预定时间段可以是与营业时间、工作时间等相关;再例如,在特征识别设备启动后开启检测,并一

直处于检测状态,不使用取消检测功能等,本领域技术人员可根据实际情况进行设置,本申请实施例在此不做限定。

(4) 取消注册功能

第一应用还可以具有取消注册功能,第一应用可以取消注册第三事件,则第一应用无法发起预定接口调用请求,以及无法接收通过回调接口发送的对象身份信息。

一个示例中,该功能的伪代码如下:

```
//取消注册 event 名为'GetOpenID'的事件
wmpf.Channel.unregisterEvent({
  event:'GetOpenID',
  success(res) {
    console.log(res)
  }
})
```

则本申请实施例提供的方法还可以包括步骤:针对第一应用取消注册第三事件。

可以理解,事件注册功能和取消注册功能也可以配合使用,设备接入方可以自定义事件注册功能和取消注册功能的生效时间,具体可以参见对开启检测功能和取消检测功能配合使用的介绍,在此不再赘述。

对于上述至少一个实施例,特征识别设备具体可以是指刷掌设备。其中,刷掌设备相较于刷脸设备等其他特征识别设备,与本申请实施例的方案更加适配。这是因为相较于刷脸设备等可能存在脸部遮挡、多张人脸等情况,刷掌设备的特性使得刷掌区域可误识别的特征较少,二次确认的必要性较低,安全指数更高,因此可以通过仅刷掌一次实现身份认证和设备接入方服务登录的双重目的。

另外,刷掌识别技术对比指纹识别技术更不容易造假。掌纹等可能因为一些物理、环境等因素被改变,而刷掌识别会更加稳定。

本申请实施例中,刷掌设备可以为设备接入方提供在脱离预定应用客户端的环境下,直接在刷掌设备上打开第三方程序(第一应用)的能力。具体而言,为设备接入方提供一个预定接口「wmpf.Channel.on('GetOpenID', callback)」,第一应用在注册好对应的事件检测后,只需要调用该预定接口,就能够让第一应用直接获取到刷掌者的 OpenID。使用者也就不再通过扫码的方式在小程序中登录。

下面以刷掌设备搭载 Android 系统,第一应用是 IoT 小程序,第一应用框架是 WMPF,第二应用是刷掌 APP 为例,介绍一种完成从运行在 WMPF 中的 IoT 小程序到刷掌 APP 的调用的完整流程,如图 6 所示,小程序在调用了预定接口「wmpf.Channel.on('GetOpenID', callback)」后,会经过图 6 中所示的时序调用:

6.1、由小程序中触发的「wmpf.Channel.on('GetOpenID', callback)」调用请求,会传递到 WMPF 中;

6.2、WMPF 会将对预定接口的本次调用封装为一个事件(即第一事件),并插入到 ContentProvider 中,即对 ContentProvider 执行操作:

```
insert(content://com.wmpf.cli.provider/invokeChannelEvent)
```

其中,封装的结构可以大致分为三个部份:事件名称、事件唯一标识和事件参数;

6.3、ContentProvider 会通知刷掌 APP,有一个事件插入了,刷掌 APP 便可以根据插入的第一事件,提取出事件名称等信息,并启动 GetOpenID 事件对应的检测逻辑,开启对刷掌数据的检测;

6.4、当刷掌识别成功获取了当前刷掌人的 OpenID 后,刷掌 APP 便会反向插入一个事件(即第二事件),即对 ContentProvider 执行操作:

```
insert(content://com.wmpf.comm.provider/notifyInvokeChannelEvent)
```

并封装好对应的事件名、6.2 步中的事件唯一标识、用户的 OpenID;

6.5、ContentProvider 会通知到 WMPF,有一个事件(第二事件)插入了,WMPF 便可以根据插入的第二事件,提取出事件名称等信息,并获得刷掌用户的 OpenID;

6.6、WMPF 将获得的 OpenID,按照返回值 res 传递给小程序,其中,res 中包含 OpenID,即'OpenID ='+ res.data。

这样,就完成了小程序获取刷掌人的用户身份(OpenID)的全过程。解决了小程序登录的可靠性问题,保证了用户信息的安全,也避免了用户还要再次主动扫码的操作流程。

本申请实施例的技术方案,基于 Android 中特有的 ContentProvider 作为底层桥梁,通过与

IoT 小程序硬件框架(WMPF)约定的刷掌获取用户身份的接口协议,实现了设备接入方开发的小程序从刷掌服务侧获取用户身份的能力。即小程序仅需要调用该特殊的接口,即可在自身层主动获取来自刷掌服务侧的当前刷掌人的用户身份。不再依赖刷掌人再次手动扫码完成登录。

本申请实施例的技术方案能够一次性解决两个痛点:

- 5 (1) 针对小程序而言,能够通过这个新的接口直接准确地获得刷掌人的身份信息,保证了用户信息的安全,避免了由于获取用户的身份信息的错误而造成的刷掌失败。

(2) 针对刷掌人而言,省去了刷掌后还要再次扫码登录的操作,大幅简化了刷掌人的使用流程,优化了使用体验。

- 10 本申请实施例的技术方案可以应用于常规场景,运行在刷掌设备上的 IoT 小程序是一个独立的环境,不会与设备的其他服务能力产生干扰。

本申请实施例的技术方案也可以应用于落地到具体刷掌业务的场景中,比如健身房刷掌门禁、刷掌设备自助收银等场景,因为这些业务方通常都会使用小程序承载,因此通过本方案支持的这种在 IoT 小程序中直接获取刷掌人身份信息的能力,让设备接入方仅需要开发自己的小程序,即可在小程序中获取到刷掌人的身份,用以服务自身的业务场景。

- 15 需要说明的是,在上述各实施例中,所涉及到的与使用者有关的各种信息(如特征数据、对象身份信息 etc),当本申请实施例运用到具体产品或技术中时,需要获得对象许可或者同意,且相关数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准。也就是说,本申请实施例中如果涉及到与对象有关的数据,这些数据需要经由对象授权同意、且符合国家和地区的相关法律法规和标准的情况下获取的。

- 20 在一些实施例中,本申请实施例所涉及的信息处理可以基于云技术(Cloud technology)实现。云技术是指在广域网或局域网内将硬件、软件、网络等系列资源统一起来,实现数据的计算、储存、处理和共享的一种托管技术。云技术基于云计算商业模式应用的网络技术、信息技术、整合技术、管理平台技术、应用技术等的总称,可以组成资源池,按需所用,灵活便利。云计算技术将变成重要支撑。云计算则是指 IT 基础设施的交付和使用模式,指通过网络以按需、易扩展的方式获得所需资源;广义云计算指服务的交付和使用模式,指通过网络以按需、易扩展的方式获得所需服务。这种服务可以是 IT 和软件、互联网相关,也可是其他服务。随着互联网、实时数据流、连接设备多样化的发展,以及搜索服务、社会网络、移动商务和开放协作等需求的推动,云计算迅速发展起来。不同于以往的并行分布式计算,云计算的产生从理念上将推动整个互联网模式、企业管理模式发生革命性的变革。

- 30 本申请实施例所涉及的数据库(Database),简而言之可视为电子化的文件柜——存储电子文件的处所,用户可以对文件中的数据进行新增、查询、更新、删除等操作。所谓“数据库”是以一定方式储存在一起、能与多个用户共享、具有尽可能小的冗余度、与应用程序彼此独立的数据集合。

- 35 在一些实施例中,本申请实施例提供的信息处理方法,部分步骤可以基于人工智能(Artificial Intelligence, AI)技术实现。比如,特征识别等。AI 是利用数字计算机或者数字计算机控制的机器模拟、延伸和扩展人的智能,感知环境、获取知识并使用知识获得最佳结果的理论、方法、技术及应用系统。换句话说,人工智能是计算机科学的一个综合技术,它企图了解智能的实质,并生产一种新的能以人类智能相似的方式做出反应的智能机器。人工智能也就是研究各种智能机器的设计原理与实现方法,使机器具有感知、推理与决策的功能。本方案可能涉及人工智能软件技术中的计算机视觉技术、机器学习技术 etc 方向。

- 40 其中,计算机视觉技术(Computer Vision, CV)计算机视觉是一门研究如何使机器“看”的科学,更进一步的讲,就是指用摄影机和电脑代替人眼对目标进行识别和测量等机器视觉,并进一步做图形处理,使电脑处理成为更适合人眼观察或传送给仪器检测的图像。作为一个科学学科,计算机视觉研究相关的理论和技术,试图建立能够从图像或者多维数据中获取信息的人工智能系统。计算机视觉技术通常包括图像处理、图像识别、图像语义理解、图像检索、OCR、视频处理、视频语义理解、视频内容/行为识别、三维物体重建、3D 技术、虚拟现实、增强现实、同步定位与地图构建、自动驾驶、智慧交通等技术,还包括常见的人脸识别、指纹识别、手掌识别等生物特征识别技术。

- 45 机器学习(Machine Learning, ML)是一门多领域交叉学科,涉及概率论、统计学、逼近论、凸分析、算法复杂度理论等多门学科。专门研究计算机怎样模拟或实现人类的学习行为,以获取新的知识或技能,重新组织已有的知识结构使之不断改善自身的性能。机器学习是人工智能的核心,是使计算机具有智能的根本途径,其应用遍及人工智能的各个领域。机器学习和深度学习通常包括

神经网络、置信网络、强化学习、迁移学习、归纳学习、式教学习等技术。

随着人工智能技术研究和进步,人工智能技术在多个领域展开研究和应用,例如常见的智能家居、智能穿戴设备、虚拟助理、智能音箱、智能营销、无人驾驶、自动驾驶、无人机、机器人、智能医疗、智能客服、车联网、自动驾驶、智慧交通等,相信随着技术的发展,人工智能技术将在更多的领域得到应用,并发挥越来越重要的价值。

本申请实施例提供了一种信息处理装置,运行于特征识别设备,所述特征识别设备中运行有第一应用和第二应用,并提供有第一应用框架,所述第一应用框架提供所述第一应用的运行环境,如图7所示,该信息处理70可以包括:第一插入模块701、检测模块702以及发送模块703,其中,

第一插入模块701用于响应于第一应用触发的预定接口调用请求,将基于预定接口调用请求生成的第一事件,插入到特征识别设备中的操作系统对应的内容提供器中;

检测模块702用于接收到内容提供器中生成的第一事件插入的通知后,从内容提供器中确定第一事件的事件信息,并基于事件信息执行对特征数据的检测;

发送模块703用于在检测到特征数据后,确定特征数据对应的对象身份信息,并将对象身份信息发送给第一应用。

在一些实施例中,该装置还包括:第二插入模块704;

发送模块703在用于将对象身份信息发送给第一应用时,

第二插入模块704用于在内容提供器中插入基于对象身份信息生成的第二事件;

发送模块703具体用于接收到内容提供器中的第二事件的插入通知后,从第二事件中提取对象身份信息,并将对象身份信息传递给第一应用。

在一些实施例中,第二插入模块704在用于向内容提供器插入基于对象身份信息生成的第二事件时,具体用于:

将第一事件的事件信息和对象身份信息进行封装,生成第二事件;

在内容提供器中插入第二事件;

其中,第一事件的事件信息包括事件名称和事件唯一标识中的至少一项。

在一些中,第一插入模块701在用于将基于预定接口调用请求生成的第一事件插入到特征识别设备中的操作系统对应的内容提供器中时,具体用于:

将第一应用对预定接口的调用封装成事件信息,得到第一事件;

将第一事件插入到特征识别设备中的操作系统对应的内容提供器中;

其中,事件信息包括事件名称、事件唯一标识和事件参数中的至少一项。

在一些实施例中,该装置还包括:第一标识模块705;

在第一插入模块701用于将基于预定接口调用请求生成的第一事件插入到特征识别设备中的操作系统对应的内容提供器时,

第一标识模块705用于通过第一统一资源标识符,对插入到内容提供器的第一事件进行标识,第一统一资源标识符用于内容提供器识别第一事件,生成并向第二应用发送内容提供器中第一事件的插入通知。

在一些实施例中,该装置还包括:第二标识模块706;

在第二插入模块704用于向内容提供器插入基于对象身份信息生成的第二事件时,

第二标识模块706用于通过第二统一资源标识符,对插入到内容提供器的第二事件进行标识,第二统一资源标识符用于内容提供器识别第二事件,生成并向第一应用框架发送第二事件的插入通知。

在一些实施例中,发送模块703在用于检测到特征数据后,确定特征数据对应的对象身份信息,并将对象身份信息发送给第一应用时,具体用于:

在检测到特征数据后,确定特征数据对应的对象身份信息,并通过回调接口回调第一应用,以将对象身份信息发送给第一应用。

在一些实施例中,事件唯一标识是基于以下至少一项确定的:

接收到预定接口调用请求的时间戳;

生成第一事件的时间戳;

开始检测特征数据的时间戳;

检测到特征数据的时间戳;

生成第一事件的时间戳。

在一些实施例中,该装置还包括:注册模块707;

在特征识别设备中运行的第一应用触发预定接口调用请求之前,注册模块 707 用于针对第一应用注册第三事件,所述第三事件用于使所述第一应用能够发起所述触发预定接口调用请求,以及接收通过回调接口发送的所述对象身份信息。

5 在一些实施例中,发送模块 703 在用于在检测到特征数据后,确定特征数据对应的对象身份信息,并将对象身份信息发送给第一应用时,具体用于:

若第三事件处于检测状态,第二应用在检测到特征数据后,确定特征数据对应的对象身份信息,触发第三事件,以通过所述回调接口将对象身份信息发送给第一应用;

在一些实施例中,该装置还包括:取消检测模块 708 和取消注册模块 709 中的至少一个;

取消检测模块 708 用于针对第一应用将第三事件变为取消检测状态;

10 取消注册模块 709 用于针对第一应用取消注册第三事件。

在一种可选的实施方式中,特征识别设备为刷掌设备。

在一种可选的实施方式中,第一应用为 IoT 小程序。

其中,其中,第一插入模块 701、第一标识模块 705 配合第一应用框架运行;检测模块 702、第二插入模块 704、第二标识模块 706 配合第二应用运行;注册模块 707、取消检测模块 708、取消注册模块 709 配合第一应用运行。

本申请实施例的装置可执行本申请实施例所提供的方法,其实现原理相类似,本申请各实施例的装置中的各模块所执行的动作是与本申请各实施例的方法中的步骤相对应的,对于装置的各模块的详细功能描述及产生的有益效果具体可以参见前文中所示的对应方法中的描述,此处不再赘述。

20 本申请实施例中提供了一种电子设备,包括存储器、处理器及存储在存储器上的计算机程序,该处理器执行上述计算机程序以实现前述各方法实施例的步骤。

其中,电子设备可以是指特征识别设备,例如刷掌设备等。

在一个可选实施例中提供了一种电子设备,如图 8 所示,图 8 所示的电子设备 800 包括:处理器 801 和存储器 803。其中,处理器 801 和存储器 803 相连,如通过总线 802 相连。在一些实施例中,电子设备 800 还可以包括收发器 804,收发器 804 可以用于该电子设备与其他电子设备之间的数据交互,如数据的发送和/或数据的接收等。需要说明的是,实际应用中收发器 804 不限于一个,该电子设备 800 的结构并不构成对本申请实施例的限定。

30 处理器 801 可以是 CPU (Central Processing Unit, 中央处理器),通用处理器,DSP (Digital Signal Processor, 数据信号处理器),ASIC (Application Specific Integrated Circuit, 专用集成电路),FPGA (Field Programmable Gate Array, 现场可编程门阵列)或者其他可编程逻辑器件、晶体管逻辑器件、硬件部件或者其任意组合。其可以实现或执行结合本申请公开内容所描述的各种示例性的逻辑方框,模块和电路。处理器 801 也可以是实现计算功能的组合,例如包含一个或多个微处理器组合,DSP 和微处理器的组合等。

35 总线 802 可包括一通路,在上述组件之间传送信息。总线 802 可以是 PCI (Peripheral Component Interconnect, 外设部件互连标准)总线或 EISA (Extended Industry Standard Architecture, 扩展工业标准结构)总线等。总线 802 可以分为地址总线、数据总线、控制总线等。为便于表示,图 8 中仅用一条粗线表示,但并不表示仅有一根总线或一种类型的总线。

40 存储器 803 可以是 ROM (Read Only Memory, 只读存储器)或可存储静态信息和指令的其他类型的静态存储设备,RAM (Random Access Memory, 随机存取存储器)或者可存储信息和指令的其他类型的动态存储设备,也可以是 EEPROM (Electrically Erasable Programmable Read Only Memory, 电可擦可编程只读存储器)、CD-ROM (Compact Disc Read Only Memory, 只读光盘)或其他光盘存储、光碟存储(包括压缩光碟、激光碟、光碟、数字通用光碟、蓝光光碟等)、磁盘存储介质、其他磁存储设备、或者能够用于携带或存储计算机程序并能够由计算机读取的任何其他介质,在此不做限定。

45 存储器 803 用于存储执行本申请实施例的计算机程序,并由处理器 801 来控制执行。处理器 801 用于执行存储器 803 中存储的计算机程序,以实现前述方法实施例所示的步骤。

本申请实施例提供了一种计算机可读存储介质,该计算机可读存储介质上存储有计算机程序,计算机程序被处理器执行时可实现前述方法实施例的步骤及相应内容。

本申请实施例还提供了一种计算机程序产品,包括计算机程序,计算机程序被处理器执行时可实现前述方法实施例的步骤及相应内容。

50 本申请的说明书和权利要求书及上述附图中的术语“第一”、“第二”、“1”、“2”等是用于区别类似的对象,而不必用于描述特定的顺序或先后次序。应该理解这样使用的数据在适当情况下可以

互换，以便这里描述的本申请的实施例能够以除图示或文字描述以外的顺序实施。

应该理解的是，虽然本申请实施例的流程图中通过箭头指示各个操作步骤，但是这些步骤的实施顺序并不受限于箭头所指示的顺序。除非本文中有明确的说明，否则在本申请实施例的一些实施场景中，各流程图中的实施步骤可以按照需求以其他的顺序执行。此外，各流程图中的部分或全部步骤基于实际的实施场景，可以包括多个子步骤或者多个阶段。这些子步骤或者阶段中的部分或全部可以在同一时刻被执行，这些子步骤或者阶段中的每个子步骤或者阶段也可以分别在不同的时刻被执行。在执行时刻不同的场景下，这些子步骤或者阶段的执行顺序可以根据需求灵活配置，本申请实施例对此不限制。

以上仅是本申请部分实施场景的可选实施方式，应当指出，对于本技术领域的普通技术人员来说，在不脱离本申请的技术构思的前提下，采用基于本申请技术思想的其他类似实施手段，同样属于本申请实施例的保护范畴。

权利要求书

- 1、一种信息处理方法，由特征识别设备执行，所述特征识别设备中运行有第一应用和第二应用，并提供有第一应用框架，所述第一应用框架提供所述第一应用的运行环境，包括：
5 所述第一应用框架响应于所述第一应用触发的预定接口调用请求，将基于所述预定接口调用请求生成的第一事件，插入到所述特征识别设备中的操作系统对应的内容提供器中；
所述第二应用接收到所述内容提供器中生成的所述第一事件插入的通知后，从所述内容提供器中确定所述第一事件的事件信息，并基于所述事件信息执行对特征数据的检测；
所述第二应用在检测到所述特征数据后，确定所述特征数据对应的对象身份信息，并将所述对象身份信息发送给所述第一应用。
- 10 2、根据权利要求1所述的信息处理方法，其中，所述将所述对象身份信息发送给所述第一应用，包括：
所述第二应用在所述内容提供器中插入基于所述对象身份信息生成的第二事件；
15 所述第一应用框架接收到所述内容提供器中所述第二事件的插入通知后，从所述第二事件中提取所述对象身份信息，并将所述对象身份信息传递给所述第一应用。
- 3、根据权利要求2所述的信息处理方法，其中，所述第二应用向所述内容提供器插入基于所述对象身份信息生成的第二事件，包括：
20 所述第二应用将所述第一事件的事件信息和所述对象身份信息进行封装，生成所述第二事件；
所述第二应用在所述内容提供器中插入所述第二事件；
其中，所述第一事件的事件信息包括事件名称和事件唯一标识中的至少一项。
- 4、根据权利要求1所述的信息处理方法，其中，所述将基于所述预定接口调用请求生成的第一事件，插入到所述特征识别设备中的操作系统对应的内容提供器中，包括：
25 所述第一应用框架将第一应用对预定接口的调用封装成事件信息，得到所述第一事件；
所述第一应用框架将所述第一事件，插入到所述特征识别设备中的操作系统对应的内容提供器中；
其中，所述事件信息包括事件名称、事件唯一标识和事件参数中的至少一项。
- 30 5、根据权利要求1所述的信息处理方法，其中，所述将基于所述预定接口调用请求生成的第一事件，插入到所述特征识别设备中的操作系统对应的内容提供器中，还包括：
所述第一应用框架通过第一统一资源标识符，对插入到所述内容提供器的所述第一事件进行标识，所述第一统一资源标识符用于所述内容提供器识别所述第一事件，生成并向所述第二应用发送所述内容提供器中所述第一事件的插入通知。
- 35 6、根据权利要求1所述的信息处理方法，其中，所述第二应用在所述内容提供器中插入基于所述对象身份信息生成的第二事件，还包括：
所述第二应用通过第二统一资源标识符，对插入到所述内容提供器的所述第二事件进行标识，所述第二统一资源标识符用于所述内容提供器识别所述第二事件，生成并向所述第一应用框架发送
40 所述第二事件的插入通知。
- 7、根据权利要求1所述的信息处理方法，其中，所述第二应用在检测到所述特征数据后，确定所述特征数据对应的对象身份信息，并将所述对象身份信息发送给所述第一应用，包括：
45 所述第二应用在检测到所述特征数据后，确定所述特征数据对应的对象身份信息，并通过回调接口回调所述第一应用，以将所述对象身份信息发送给所述第一应用。
- 8、根据权利要求3或4所述的信息处理方法，其中，所述事件唯一标识是基于以下至少一项确定的：
接收到所述预定接口调用请求的时间戳；

生成所述第一事件的时间戳;
开始检测所述特征数据的时间戳;
检测到所述特征数据的时间戳;
生成所述第一事件的时间戳。

5

9、根据权利要求 1-7 任一项所述的信息处理方法,其中,在所述第一应用中触发预定接口调用请求之前,还包括:

针对所述第一应用注册第三事件,所述第三事件用于使所述第一应用能够发起所述触发预定接口调用请求,以及接收通过回调接口发送的所述对象身份信息;

10

所述第二应用在检测到所述特征数据后,确定所述特征数据对应的对象身份信息,并将所述对象身份信息发送给所述第一应用,包括:

若所述第三事件处于检测状态,所述第二应用在检测到所述特征数据后,确定所述特征数据对应的对象身份信息,触发所述第三事件,以通过所述回调接口将所述对象身份信息发送给所述第一应用;

15

所述方法还包括以下至少一项:

针对所述第一应用将所述第三事件变更为取消检测状态;

针对所述第一应用取消注册所述第三事件。

10、根据权利要求 1-7 任一项所述的信息处理方法,其中,所述特征识别设备为刷掌设备。

20

11、根据权利要求 1-7 任一项所述的信息处理方法,其中,所述第一应用为物联网 IoT 小程序。

12、一种信息处理装置,运行于特征识别设备,所述特征识别设备中运行有第一应用和第二应用,并提供有第一应用框架,所述第一应用框架提供所述第一应用的运行环境,包括:

25

第一插入模块,用于响应于所述第一应用触发的预定接口调用请求,将基于所述预定接口调用请求生成的第一事件,插入到所述特征识别设备中的操作系统对应的内容提供者中;

检测模块,用于接收到所述内容提供者中生成的所述第一事件插入的通知后,从所述内容提供者中确定所述第一事件的事件信息,并基于所述事件信息执行对特征数据的检测;

30

发送模块,用于在检测到所述特征数据后,确定所述特征数据对应的对象身份信息,并将所述对象身份信息发送给所述第一应用。

13、根据权利要求 12 所述的信息处理装置,其中,所述装置还包括第二插入模块,用于:

所述发送模块在用于将对象身份信息发送给第一应用时,在所述内容提供者中插入基于所述对象身份信息生成的第二事件;

35

所述发送模块,用于接收到所述内容提供者中所述第二事件的插入通知后,从所述第二事件中提取所述对象身份信息,并将所述对象身份信息传递给所述第一应用。

14、根据权利要求 13 所述的信息处理装置,其中,所述第二插入模块用于:

将所述第一事件的事件信息和所述对象身份信息进行封装,生成所述第二事件;

在所述内容提供者中插入所述第二事件;

40

其中,所述第一事件的事件信息包括事件名称和事件唯一标识中的至少一项。

15、根据权利要求 12 所述的信息处理装置,其中,所述第一插入模块用于:

将第一应用对预定接口的调用封装成事件信息,得到所述第一事件;

将所述第一事件,插入到所述特征识别设备中的操作系统对应的内容提供者中;

45

其中,所述事件信息包括事件名称、事件唯一标识和事件参数中的至少一项。

16、根据权利要求 12 所述的信息处理装置,其中,所述装置还包括:第一标识模块用于:

通过第一统一资源标识符,对插入到所述内容提供器的所述第一事件进行标识,所述第一统一资源标识符用于所述内容提供者识别所述第一事件,生成并向所述第二应用发送所述内容提供者中所述第一事件的插入通知。

50

17、根据权利要求 12 所述的信息处理装置，其中，所述装置还包括：第一标识模块用于：
通过第二统一资源标识符，对插入到所述内容提供器的所述第二事件进行标识，所述第二统一
资源标识符用于所述内容提供器识别所述第二事件，生成并向所述第一应用框架发送所述第二事件
的插入通知。

5

18、一种电子设备，包括存储器、处理器及存储在存储器上的计算机程序，其特征在于，所述
处理器执行所述计算机程序以实现权利要求 1-11 任一项所述的方法。

19、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 任一项所述的方法。

10

20、一种计算机程序产品，包括计算机程序，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 任一项所述的方法。

15

20