



(12) 发明专利

(10) 授权公告号 CN 112328824 B

(45) 授权公告日 2024.06.18

(21) 申请号 202010707675.5

(22) 申请日 2020.07.21

(65) 同一申请的已公布的文献号  
申请公布号 CN 112328824 A

(43) 申请公布日 2021.02.05

(73) 专利权人 北京沃东天骏信息技术有限公司  
地址 100176 北京市大兴区北京经济技术  
开发区科创十一街18号院2号楼4层  
A402室  
专利权人 北京京东世纪贸易有限公司

(72) 发明人 王云锋

(74) 专利代理机构 中科专利商标代理有限责任  
公司 11021  
专利代理师 吕朝蕙

(51) Int.Cl.

G06F 16/583 (2019.01)

G06F 8/41 (2018.01)

(56) 对比文件

何敏丽;成刚虎.ICC Profile及基于ICC的  
色彩管理技术.广东印刷.2006,(06),全文.

张晓燕;胡涛.Windows 98与Windows NT5.0  
的色彩管理系统——ICM2.0.中国印刷.2000,  
(09),全文.

审查员 刘娜

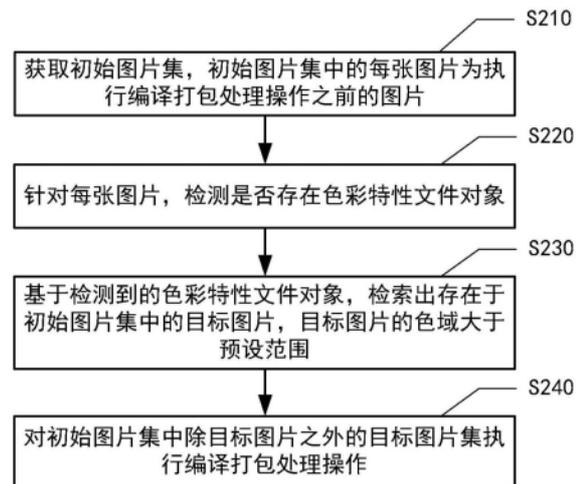
权利要求书2页 说明书10页 附图5页

(54) 发明名称

图片检测方法及其系统、计算机系统和计算  
机可读介质

(57) 摘要

本公开提供了一种图片检测方法,包括:获  
取初始图片集,初始图片集中的每张图片为执行  
编译打包处理操作之前的图片;针对每张图片,  
检测是否存在色彩特性文件对象;基于检测到的  
色彩特性文件对象,检索出存在于初始图片集中  
的目标图片,目标图片的色域大于预设范围;对  
初始图片集中除目标图片之外的目标图片集执  
行编译打包处理操作。此外,本公开还提供了一  
种图片检测系统、计算机系统和计算机可读介  
质。



1. 一种图片检测方法,包括:

获取初始图片集,所述初始图片集中的每张图片为执行编译打包处理操作之前的图片;

针对所述每张图片,检测是否存在色彩特性文件对象;

基于检测到的所述色彩特性文件对象,检索出存在于所述初始图片集中的目标图片,所述目标图片的色域大于预设范围;

对所述初始图片集中除所述目标图片之外的目标图片集执行编译打包处理操作;

其中,所述针对所述每张图片,检测是否存在色彩特性文件对象包括:

针对所述每张图片,读取图片摘要信息;

基于所述图片摘要信息,检测是否存在所述色彩特性文件对象;

所述基于检测到的所述色彩特性文件对象,检索出存在于所述初始图片集中的目标图片包括:

基于检测到的所述色彩特性文件对象,检测所述图片摘要信息中是否包含指定显示类型信息,其中,所述指定显示类型信息用于表征图片的色域大于所述预设范围;

将所述图片摘要信息中包含所述指定显示类型信息的图片,作为检索出的存在于所述初始图片集中的目标图片。

2. 根据权利要求1所述的方法,其中,所述读取图片摘要信息包括:

创建图像处理库对象;

利用所述图像处理库对象,读取所述图片摘要信息。

3. 根据权利要求1所述的方法,其中,在检索出存在于所述初始图片集中的目标图片之后,所述方法还包括:

保存所述目标图片的图片摘要信息。

4. 一种图片检测系统,包括:

获取模块,用于获取初始图片集,所述初始图片集中的每张图片为执行编译打包处理操作之前的图片;

检测模块,用于针对所述每张图片,检测是否存在色彩特性文件对象;

检索模块,用于基于检测到的所述色彩特性文件对象,检索出存在于所述初始图片集中的目标图片,所述目标图片的色域大于预设范围;

处理模块,用于对所述初始图片集中除所述目标图片之外的目标图片集执行编译打包处理操作;

所述检测模块包括:

读取子模块,用于针对所述每张图片,读取图片摘要信息;

第一检测子模块,用于基于所述图片摘要信息,检测是否存在所述色彩特性文件对象;

所述检索模块包括:

第二检测子模块,用于基于检测到的所述色彩特性文件对象,检测所述图片摘要信息中是否包含指定显示类型信息,其中,所述指定显示类型信息用于表征图片的色域大于所述预设范围;

处理子模块,用于将所述图片摘要信息中包含所述指定显示类型信息的图片,作为检索出的存在于所述初始图片集中的目标图片。

5. 根据权利要求4所述的系统,其中,所述读取子模块包括:
  - 创建单元,用于创建图像处理库对象;
  - 读取单元,用于利用所述图像处理库对象,读取所述图片摘要信息。
6. 根据权利要求4所述的系统,其中,在检索出存在于所述初始图片集中的目标图片之后,所述系统还包括:
  - 保存模块,用于保存所述目标图片的图片摘要信息。
7. 一种计算机系统,包括:
  - 一个或多个处理器;以及
  - 存储装置,用于存储一个或多个程序,其中,当所述一个或多个程序被所述一个或多个处理器执行时,使得所述一个或多个处理器实现权利要求1至3中任一项所述的方法。
8. 一种计算机可读介质,其上存储有可执行指令,该指令被处理器执行时使处理器实现权利要求1至3中任一项所述的方法。

## 图片检测方法及其系统、计算机系统和计算机可读介质

### 技术领域

[0001] 本公开涉及图片处理领域,更具体地,涉及一种图片检测方法及其系统、计算机系统和计算机可读介质。

### 背景技术

[0002] 为了赢得更多的用户,并给用户提供良好的用户体验,大部分应用程序将最低的iPhone的操作系统iOS的适配系统都设置为大于或等于8.0。但是由于iOS之间存在的差异性,使得在iOS10以下的系统在加载某些(例如对所有颜色在标准颜色空间上的投影范围,即色域大于预设范围的P3)图片时,应用程序商店(App Store)包会出现异常闪退的现象。因此,P3图片的引入无形当中会给应用程序带来巨大的隐患。

[0003] 现有技术没有提供直接的获取图片P3信息的接口,P3图片检查一直沿用assetutil工具进行解压得到assets.json文件,再对assets.json文件进行搜索P3字样进行排查.assetutil工具在使用上有很大的局限性,且该方式依赖于电脑系统高级权限的获取,即便是使用脚本进行排查,也无法做到全自动化,且覆盖率低。

### 发明内容

[0004] 有鉴于此,本公开提供了一种图片检测方法及其系统、计算机系统和计算机可读介质。

[0005] 本公开的一个方面提供了一种图片检测方法,包括:获取初始图片集,上述初始图片集中的每张图片为执行编译打包处理操作之前的图片,针对上述每张图片,检测是否存在色彩特性文件对象,基于检测到的上述色彩特性文件对象,检索出存在于上述初始图片集中的目标图片,上述目标图片的色域大于预设范围,对上述初始图片集中除上述目标图片之外的目标图片集执行编译打包处理操作。

[0006] 根据本公开的实施例,上述针对上述每张图片,检测是否存在色彩特性文件对象包括:针对上述每张图片,读取图片摘要信息,基于上述图片摘要信息,检测是否存在上述色彩特性文件对象。

[0007] 根据本公开的实施例,上述读取图片摘要信息包括:创建图像处理库对象,利用上述图像处理库对象,读取上述图片摘要信息。

[0008] 根据本公开的实施例,上述基于检测到的上述色彩特性文件对象,检索出存在于上述初始图片集中的目标图片包括:基于检测到的上述色彩特性文件对象,检测上述图片摘要信息中是否包含指定显示类型信息,其中,上述指定显示类型信息用于表征图片的色域大于上述预设范围,将上述图片摘要信息中包含上述指定显示类型信息的图片,作为检索出的存在于上述初始图片集中的目标图片。

[0009] 根据本公开的实施例,在检索出存在于上述初始图片集中的目标图片之后,上述方法还包括:保存上述目标图片的图片摘要信息。

[0010] 本公开的另一个方面提供了一种图片检测系统,包括:获取模块,用于获取初始图

片集,上述初始图片集中的每张图片为执行编译打包处理操作之前的图片,检测模块,用于针对上述每张图片,检测是否存在色彩特性文件对象,检索模块,用于基于检测到的上述色彩特性文件对象,检索出存在于上述初始图片集中的目标图片,上述目标图片的色域大于预设范围,处理模块,用于对上述初始图片集中除上述目标图片之外的目标图片集执行编译打包处理操作。

[0011] 根据本公开的实施例,上述检测模块包括:读取子模块,用于针对上述每张图片,读取图片摘要信息,第一检测子模块,用于基于上述图片摘要信息,检测是否存在上述色彩特性文件对象。

[0012] 根据本公开的实施例,上述读取子模块包括:创建单元,用于创建图像处理库对象,读取单元,用于利用上述图像处理库对象,读取上述图片摘要信息。

[0013] 根据本公开的实施例,上述检索模块包括:第二检测子模块,用于基于检测到的上述色彩特性文件对象,检测上述图片摘要信息中是否包含指定显示类型信息,其中,上述指定显示类型信息用于表征图片的色域大于上述预设范围,处理子模块,用于将上述图片摘要信息中包含上述指定显示类型信息的图片,作为检索出的存在于上述初始图片集中的目标图片。

[0014] 根据本公开的实施例,在检索出存在于上述初始图片集中的目标图片之后,上述系统还包括:保存模块,用于保存上述目标图片的图片摘要信息。

[0015] 本公开的另一方面提供了一种计算机系统,包括:一个或多个处理器;存储装置,用于存储一个或多个程序,其中,当一个或多个程序被一个或多个处理器执行时,使得一个或多个处理器实现上述任一项的方法。

[0016] 本公开的另一方面提供了一种计算机可读介质,其上存储有可执行指令,该指令被处理器执行时使处理器实现上述任一项的方法。

[0017] 通过本公开的实施例,针对iOS工程中的P3这一目标图片,在对初始图片集中的每张图片执行编译打包处理操作之前,对每张图片的色彩特性文件对象进行检测,以检索出存在于初始图片集中的目标图片,最终对初始图片集中除目标图片之外的目标图片集执行编译打包处理操作,使得目标图片的检测做到全自动化,在不依赖于编译打包的情况下,能全面逐一的对所有iOS工程中使用到的图片进行排查,彻底解放人工检测,有效提高图片检测效率,降低系统安全隐患。

## 附图说明

[0018] 通过以下参照附图对本公开实施例的描述,本公开的上述以及其他目的、特征和优点将更为清楚,在附图中:

[0019] 图1示意性示出了可以应用本公开实施例的图片检测方法及其系统的示例性系统架构;

[0020] 图2示意性示出了根据本公开实施例的图片检测方法的流程图;

[0021] 图3示意性示出了根据本公开另一实施例的图片检测方法的流程图;

[0022] 图4示意性示出了根据本公开另一实施例的图片检测方法的流程图;

[0023] 图5示意性示出了根据本公开实施例的图片检测系统的框图;以及

[0024] 图6示意性示出了根据本公开实施例的适于实现图片检测方法及其系统的计算机

系统的框图。

### 具体实施方式

[0025] 以下,将参照附图来描述本公开的实施例。但是应该理解,这些描述只是示例性的,而并非要限制本公开的范围。此外,在以下说明中,省略了对公知结构和技术的描述,以避免不必要地混淆本公开的概念。

[0026] 在此使用的术语仅仅是为了描述具体实施例,而并非意在限制本公开。在此使用的术语“包括”、“包含”等表明了所述特征、步骤、操作和/或部件的存在,但是并不排除存在或添加一个或多个其他特征、步骤、操作或部件。

[0027] 在此使用的所有术语(包括技术和科学术语)具有本领域技术人员通常所理解的含义,除非另外定义。应注意,这里使用的术语应解释为具有与本说明书的上下文相一致的含义,而不应以理想化或过于刻板的方式来解释。

[0028] 在使用类似于“A、B和C等中至少一个”这样的表述的情况下,一般来说应该按照本领域技术人员通常理解该表述的含义来予以解释(例如,“具有A、B和C中至少一个的系统”应包括但不限于单独具有A、单独具有B、单独具有C、具有A和B、具有A和C、具有B和C、和/或具有A、B、C的系统等)。在使用类似于“A、B或C等中至少一个”这样的表述的情况下,一般来说应该按照本领域技术人员通常理解该表述的含义来予以解释(例如,“具有A、B或C中至少一个的系统”应包括但不限于单独具有A、单独具有B、单独具有C、具有A和B、具有A和C、具有B和C、和/或具有A、B、C的系统等)。本领域技术人员还应理解,实质上任意表示两个或更多可选项目的转折连词和/或短语,无论是在说明书、权利要求书还是附图中,都应被理解为给出了包括这些项目之一、这些项目任一方、或两个项目的可能性。例如,短语“A或B”应当被理解为包括“A”或“B”、或“A和B”的可能性。

[0029] 本公开提供了一种图片检测方法,包括:首先,获取初始图片集,初始图片集中的每张图片为执行编译打包处理操作之前的图片。然后,针对每张图片,检测是否存在色彩特性文件对象。接着,基于检测到的色彩特性文件对象,检索出存在于初始图片集中的目标图片,目标图片的色域大于预设范围。最后,对初始图片集中除目标图片之外的目标图片集执行编译打包处理操作。

[0030] 通过本公开的实施例,针对iOS工程中的P3这一目标图片,在对初始图片集中的每张图片执行编译打包处理操作之前,对每张图片的色彩特性文件对象进行检测,以检索出存在于初始图片集中的目标图片,最终对初始图片集中除目标图片之外的目标图片集执行编译打包处理操作,使得目标图片的检测做到全自动化,在不依赖于编译打包的情况下,能全面逐一的对所有iOS工程中使用到的图片进行排查,彻底解放人工检测,有效提高图片检测效率,降低系统安全隐患。

[0031] 图1示意性示出了可以应用本公开实施例的图片检测方法及其系统的示例性系统架构100。需要注意的是,图1所示仅为可以应用本公开实施例的系统架构的示例,以帮助本领域技术人员理解本公开的技术内容,但并不意味着本公开实施例不可以用于其他设备、系统、环境或场景。

[0032] 如图1所示,根据该实施例的系统架构100可以包括终端设备101、102、103,网络104和服务器105。网络104用以在终端设备101、102、103和服务器105之间提供通信链路的

介质。网络104可以包括各种连接类型,例如有线、无线通信链路或者光纤电缆等等。

[0033] 用户可以使用终端设备101、102、103通过网络104与服务器105交互,以接收或发送消息等。终端设备101、102、103上可以安装有各种通讯客户端应用,例如购物类应用、网页浏览器应用、搜索类应用、即时通信工具、邮箱客户端、社交平台软件等(仅为示例)。

[0034] 终端设备101、102、103可以是具有显示屏并且支持网页浏览的各种电子设备,包括但不限于智能手机、平板电脑、膝上型便携计算机和台式计算机等等。

[0035] 服务器105可以是提供各种服务的服务器,例如对用户利用终端设备101、102、103所浏览的网站提供支持的后台管理服务器(仅为示例)。后台管理服务器可以对接收到的用户请求等数据进行分析等处理,并将处理结果(例如根据用户请求获取或生成的网页、信息或数据等)反馈给终端设备。

[0036] 需要说明的是,本公开实施例所提供的图片检测方法一般可以由服务器105执行。相应地,本公开实施例所提供的图片检测系统一般可以设置于服务器105中。本公开实施例所提供的图片检测方法也可以由不同于服务器105且能够与终端设备101、102、103和/或服务器105通信的服务器或服务器集群执行。相应地,本公开实施例所提供的图片检测系统也可以设置于不同于服务器105且能够与终端设备101、102、103和/或服务器105通信的服务器或服务器集群中。

[0037] 应该理解,图1中的终端设备、网络和服务器的数目仅仅是示意性的。根据实现需要,可以具有任意数目的终端设备、网络和服务器。

[0038] 图2示意性示出了根据本公开实施例的图片检测方法的流程图。

[0039] 如图2所示,该方法可以包括操作S210~S240。

[0040] 在操作S210,获取初始图片集,初始图片集中的每张图片为执行编译打包处理操作之前的图片。

[0041] 根据本公开的实施例,初始图片集可以是iOS系统工程中所能用到的图片组成的图片集。较优地,可以是iOS系统工程中所能用到的所有图片组成的图片集。初始图片集中的图片可以是jpg格式的图片,也可以是png格式的图片,本公开对此不做限定。

[0042] 为了克服相关技术中,由于研发人员规范性或应用程序执行性能,使得图片编译打包出现遗漏的技术问题,或者即使编译打包没有遗漏,在编译打包之后,识别P3图片的效果不好的技术问题,本公开的若干实施例创新性的将P3图片的检测识别过程,置于编译打包处理操作之前执行。

[0043] 在操作S220,针对每张图片,检测是否存在色彩特性文件对象。

[0044] 根据本公开的实施例,色彩特性文件(icc\_profile)对象是国际色彩联盟(International Color Consortium,ICC)创立的标准的色彩管理系统的核心,是用于描述某一彩色设备的色彩特性的文件,表示该彩色设备的色彩描述方式与标准色彩空间的对应关系,不同的设备之间对于色彩的描述通过标准颜色空间和这些设备的icc\_profile联系起来。设备各自的所有颜色在标准颜色空间上的投影范围,称为色域。本公开中的P3图片,色域空间范围较其他图片的色域空间范围更大。

[0045] 具体地,对于任意一张jpg或者png图片来说,打开之后,图片属性信息中的更多信息里,颜色描述文件为Display P3的图片称为P3图片。

[0046] 在操作S230,基于检测到的色彩特性文件对象,检索出存在于初始图片集中的目

标图片,目标图片的色域大于预设范围。

[0047] 对图片的隐藏信息进行研究时,根据本公开的实施例,对检测到色彩特征文件的图片来说,查找是否包含有P3字样,若有则该图片为目标图片。

[0048] 在操作S240,对初始图片集中除目标图片之外的目标图片集执行编译打包处理操作。

[0049] 根据本公开的实施例,对检索出的目标图片不执行编译打包处理操作,可以有效避免应用程序包出现异常闪退的技术问题,使得应用程序包正常运行而不会闪退。

[0050] 在相关技术中,P3图片检测方法具体为:iOS工程中放入imageasset的图片经打包之后所生成的图片压缩包即Asset.car压缩包,依赖于系统的高级系统权限管理命令sudo(该命令允许系统管理员王普通用户执行一些或全部的root命令的一个工具),使用提供给开发者解压图片打包之后的assetutil工具对Asserts.car进行解压得到assets.json(图片压缩生成的图片信息摘要)文件,再对assets.json文件进行搜索,即扫描该文件,查找DisplayGamut是否包含P3字样,若包含,则抛出该图片,若不包含则结束搜索。相关技术中的P3图片检测方法由于使用assetutil工具,每次排查都需要打包,业内没有很好的解决方案,且该方式依赖于电脑系统高级权限的获取,即便是使用脚本进行排查,也无法做到全自动化,且覆盖率低。传统流程由于需要解压Assets.car得出Assets.json才能开始进行P3字样的扫描,解压过程还依赖于电脑高级权限的支持,编译打包过程有漫长,不可预测,检测过程繁琐。

[0051] 相关技术中,并没有提供过使用色彩特性文件来检测P3图片的技术方案,而本公开可以彻底摒弃相关技术提供的P3图片的检测方式,及时发现并彻底解决问题。

[0052] 进一步地,本公开提供的图片检测方法,脱离传统的置于编译后的检测流程,将整个检测过程放置于编译前的阶段去实现。在对初始图片集中的每张图片不执行编译打包处理操作的前提下,对iOS工程中的所有图片信息的色彩特性文件信息进行检测,查找是否存在包含P3信息的目标图片。由于整个过程并没有生成assets.car,因此可以避免后续使用assetutil工具进行解压得到assets.json,再从assets.json中搜索包含P3信息的目标图片的一系列处理操作,就能从初始图片集中成功的获取到所有的P3图片,从而避免了相关技术流程搜索范围不全,容易出现遗漏的可能。

[0053] 作为一种可选的实施例,针对每张图片,检测是否存在色彩特性文件对象包括:针对每张图片,读取图片摘要信息;基于图片摘要信息,检测是否存在色彩特性文件对象。

[0054] 作为一种可选的实施例,读取图片摘要信息包括:创建图像处理库对象;利用图像处理库对象,读取图片摘要信息。

[0055] 根据本公开的实施例,图片摘要信息包括图片的显性信息,也包含图片的隐藏信息。摘要信息是用于描述图片属性的各种信息,可以包括但不限于通用信息、更多信息和名称与扩展名信息。通用信息可以包括但不限于种类、大小、位置、创建时间以及修改时间。更多信息可以包括但不限于尺寸、颜色空间、颜色描述文件以及Alpha通道。

[0056] 通过摘要信息可以检测是否包含色彩特性文件对象。具体地,可以通过图像处理库对象来读取图片摘要信息。

[0057] 可选地,图像处理库对象可以是Python图像库(Python Image Library,PIL)的image对象。作为一种可选的实施例,基于检测到的色彩特性文件对象,检索出存在于初始

图片集中的目标图片包括:基于检测到的色彩特性文件对象,检测图片摘要信息中是否包含指定显示类型信息,其中,指定显示类型信息用于表征图片的色域大于预设范围;将图片摘要信息中包含指定显示类型信息的图片,作为检索出的存在于初始图片集中的目标图片。

[0058] 作为一种可选的实施例,在检索出存在于初始图片集中的目标图片之后,方法还包括:保存目标图片的图片摘要信息。

[0059] 根据本公开的实施例,针对检测到包含色彩特性文件对象的图片来说,可以进一步检测是否包含显示类型(DisplayType),且显示类型的值为P3。若包含,则表明该图片为P3图片,保存该目标图片的图片摘要信息。若不包含,则表明该图片不是P3图片,不需要保存。

[0060] 图3示意性示出了根据本公开另一实施例的图片检测方法的流程图。

[0061] 如图3所示,该图片检测方法包括操作S310~S360。

[0062] 在操作S310,代码提交。

[0063] 在操作S320,P3图片检测。

[0064] 在操作S330,检测是否包含P3图片。若是,则执行操作S340。若否,则执行操作S350。

[0065] 在操作S340,结束并终止打包过程。

[0066] 在操作S350,执行编译操作。

[0067] 在操作S360,执行打包操作。

[0068] 本公开彻底摒弃了相关技术中的利用assetutil工具进行图片检测的方法,使得整个图片检测过程不需要获取系统的高级权限,在编译打包之前执行,不依赖于编译打包和系统高级权限支持,研发人员可随时检测。

[0069] 图4示意性示出了根据本公开另一实施例的图片检测方法的流程图。

[0070] 如图4所示,该图片检测方法包括操作S410~S4100。

[0071] 在操作S410,工程目录下所有文件、文件夹扫描。

[0072] 在操作S420,创建子文件夹迭代器对象。

[0073] 在操作S430,遍历每张jpg和png图片。

[0074] 在操作S440,创建PIL image对象。

[0075] 在操作S450,检测是否包含icc\_profile对象。若是,则执行操作S470。若否,则执行操作S460。

[0076] 在操作S460,跳过当前图片,继续检测下一张图片。

[0077] 在操作S470,检测是否包含DisplayType,且值为P3。若是,则执行操作S480。若否,则执行操作S460。

[0078] 在操作S480,检测到一张P3图片,保存P3文件上下文。上下文可以包括摘要信息,即属性信息。

[0079] 在操作S490,检测是否扫描完毕。若是,则执行操作S4100。若否,则执行操作S460。

[0080] 在操作S4100,结束检测过程。

[0081] 本公开实施例在对图片进行编译打包之前就对工程目录进行读取所有的图片文件,从中检索出隐藏在icc\_profile中的P3信息,从而确定P3图的信息,成功避免了相关技

术提供的图片检测方法存在的种种技术问题。

[0082] 经过实测,对亿级日活的某购物网站应用程序工程来说,三千多张jpg、png图片,利用本公开提供的图片检测方法扫描时长仅需10秒,相对于传统的图片检测方法大概耗时为分钟级别来说,大大降低了检测时间,通过多次试验对比,本公开提供的图片检测方法在准确性上和扫描Assets.json完全一致,但是本公开提供的图片检测方法整个检测过程能彻底覆盖所有的工程图片,具有良好的实用性。

[0083] 作为一种可选的实施例,本公开实施例可以通过脚本实现图片检测方法。例如,可以利用P3check4py2.py作为检测脚本,结合需要检测的路径为参数,通过本公开的实施例最终可以输出该路径及其子路径下所检测到的P3图。该方案已应用于某购物网站的主站应用程序的日常打包流程中,在连续10个版本迭代过程中,成功规避了2例由于P3图片导入而可能引发的线上的问题。

[0084] 本公开提供的图片检测方法,在对图片的隐藏信息进行研究时,提供了一套完整的解决方案,在不编译打包的前提下,对工程中所有图片信息的icc\_profile信息进行检测,查找P3信息的包含情况。由于此过程并还没Assets.car的生成,能成功的获取到所有的图片,从而避免了传统流程漏掉的可能。此外,本公开可以完全脱离传统检测流程,将检测过程放置于编译前的阶段去执行,由于不依赖于编译打包,能全面逐一的对所有iOS工程中使用到的图片进行排查,彻底解放人工检测。

[0085] 图5示意性示出了根据本公开实施例的图片检测系统的框图。

[0086] 如图5,该系统500可以包括获取模块510,检测模块520,检索模块530以及处理模块540。

[0087] 获取模块510,用于获取初始图片集,初始图片集中的每张图片为执行编译打包处理操作之前的图片。

[0088] 检测模块520,用于针对每张图片,检测是否存在色彩特性文件对象。

[0089] 检索模块530,用于基于检测到的色彩特性文件对象,检索出存在于初始图片集中的目标图片,目标图片的色域大于预设范围。

[0090] 处理模块540,用于对初始图片集中除目标图片之外的目标图片集执行编译打包处理操作。

[0091] 通过本公开的实施例,针对iOS工程中的P3这一目标图片,在对初始图片集中的每张图片执行编译打包处理操作之前,对每张图片的色彩特性文件对象进行检测,以检索出存在于初始图片集中的目标图片,最终对初始图片集中除目标图片之外的目标图片集执行编译打包处理操作,使得目标图片的检测做到全自动化,在不依赖于编译打包的情况下,能全面逐一的对所有iOS工程中使用到的图片进行排查,彻底解放人工检测,有效提高图片检测效率,降低系统安全隐患。

[0092] 作为一种可选的实施例,检测模块包括:读取子模块,用于针对每张图片,读取图片摘要信息;第一检测子模块,用于基于图片摘要信息,检测是否存在色彩特性文件对象。

[0093] 作为一种可选的实施例,读取子模块包括:创建单元,用于创建图像处理库对象;读取单元,用于利用图像处理库对象,读取图片摘要信息。

[0094] 作为一种可选的实施例,检索模块包括:第二检测子模块,用于基于检测到的色彩特性文件对象,检测图片摘要信息中是否包含指定显示类型信息,其中,指定显示类型信息

用于表征图片的色域大于预设范围;处理子模块,用于将图片摘要信息中包含指定显示类型信息的图片,作为检索出的存在于初始图片集中的目标图片。

[0095] 作为一种可选的实施例,在检索出存在于初始图片集中的目标图片之后,系统还包括:保存模块,用于保存目标图片的图片摘要信息。

[0096] 本公开彻底摒弃了相关技术中的利用assetutil工具进行图片检测的方法,使得整个图片检测过程不需要获取系统的高级权限,在编译打包之前执行,不依赖于编译打包和系统高级权限支持,研发人员可随时检测。

[0097] 可以理解的是,获取模块510,检测模块520,检索模块530、处理模块540、读取子模块、第一检测子模块、创建单元、读取单元、第二检测子模块、处理子模块以及保存模块可以合并在一个模块中实现,或者其中的任意一个模块可以被拆分成多个模块。或者,这些模块中的一个或多个模块的至少部分功能可以与其他模块的至少部分功能相结合,并在一个模块中实现。根据本发明的实施例,获取模块510,检测模块520,检索模块530、处理模块540、读取子模块、第一检测子模块、创建单元、读取单元、第二检测子模块、处理子模块以及保存模块中的至少一个可以至少被部分地实现为硬件电路,例如现场可编程门阵列(FPGA)、可编程逻辑阵列(PLA)、片上系统、基板上的系统、封装上的系统、专用集成电路(ASIC),或可以对电路进行集成或封装的任何其他的合理方式等硬件或固件来实现,或以软件、硬件以及固件三种实现方式的适当组合来实现。或者,获取模块510,检测模块520,检索模块530、处理模块540、读取子模块、第一检测子模块、创建单元、读取单元、第二检测子模块、处理子模块以及保存模块中的至少一个可以至少被部分地实现为计算机程序模块,当该程序被计算机运行时,可以执行相应模块的功能。

[0098] 图6示意性示出了根据本公开实施例的适于实现图片检测方法及其系统的计算机系统的框图。图6示出的计算机系统仅仅是一个示例,不应对本公开实施例的功能和使用范围带来任何限制。

[0099] 如图6所示,根据本公开实施例的计算机系统600包括处理器601,其可以根据存储在只读存储器(ROM)602中的程序或者从存储部分608加载到随机访问存储器(RAM)603中的程序而执行各种适当的动作和处理。处理器601例如可以包括通用微处理器(例如CPU)、指令集处理器和/或相关芯片组和/或专用微处理器(例如,专用集成电路(ASIC)),等等。处理器601还可以包括用于缓存用途的板载存储器。处理器601可以包括用于执行如上描述的根据本公开实施例的方法流程的不同动作的单一处理单元或者是多个处理单元。

[0100] 在RAM 603中,存储有系统600操作所需的各种程序和数据。处理器601、ROM 602以及RAM 603通过总线604彼此相连。处理器601通过执行ROM 602和/或RAM 603中的程序来执行以上描述的图片检测方法的各种操作。需要注意,程序也可以存储在除ROM 602和RAM 603以外的一个或多个存储器中。处理器601也可以通过执行存储在一个或多个存储器中的程序来执行以上描述的图片检测方法的各种操作。

[0101] 根据本公开的实施例,系统600还可以包括输入/输出(I/O)接口605,输入/输出(I/O)接口605也连接至总线604。系统600还可以包括连接至I/O接口605的以下部件中的一项或多项:包括键盘、鼠标等的输入部分601;包括诸如阴极射线管(CRT)、液晶显示器(LCD)等以及扬声器等的输出部分607;包括硬盘等的存储部分608;以及包括诸如LAN卡、调制解调器等的网络接口卡的通信部分609。通信部分609经由诸如因特网的网络执行通信处理。

驱动器610也根据需要连接至I/O接口605。可拆卸介质611,诸如磁盘、光盘、磁光盘、半导体存储器等等,根据需要安装在驱动器610上,以便于从其上读出的计算机程序根据需要被安装入存储部分608。

[0102] 根据本公开的实施例,上文参考流程图描述的方法可以被实现为计算机软件程序。例如,本公开的实施例包括一种计算机程序产品,其包括承载在计算机可读介质上的计算机程序,该计算机程序包含用于执行流程图所示的方法的程序代码。在这样的实施例中,该计算机程序可以通过通信部分609从网络上被下载和安装,和/或从可拆卸介质611被安装。在该计算机程序被处理器601执行时,执行本公开实施例的系统中限定的上述功能。根据本公开的实施例,上文描述的系统、设备、装置、模块、单元等可以通过计算机程序模块来实现。

[0103] 需要说明的是,本公开所示的计算机可读介质可以是计算机可读信号介质或者计算机可读存储介质或者是上述两者的任意组合。计算机可读存储介质例如可以是——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。计算机可读存储介质的更具体的例子可以包括但不限于:具有一个或多个导线的电连接、便携式计算机磁盘、硬盘、随机访问存储器(RAM)、只读存储器(ROM)、可擦式可编程只读存储器(EPROM或闪存)、光纤、便携式紧凑磁盘只读存储器(CD-ROM)、光存储器件、磁存储器件、或者上述的任意合适的组合。在本公开中,计算机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。而在本公开中,计算机可读的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,其中承载了计算机可读的程序代码。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。计算机可读的信号介质还可以是计算机可读存储介质以外的任何计算机可读介质,该计算机可读介质可以发送、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。计算机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、电线、光缆、RF等等,或者上述的任意合适的组合。根据本公开的实施例,计算机可读介质可以包括上文描述的ROM 602和/或RAM 603和/或ROM 602和RAM 603以外的一个或多个存储器。

[0104] 附图中的流程图和框图,图示了按照本公开各种实施例的系统、方法和计算机程序产品的可能实现的体系架构、功能和操作。在这点上,流程图或框图中的每个方框可以代表一个模块、程序段、或代码的一部分,上述模块、程序段、或代码的一部分包含一个或多个用于实现规定的逻辑功能的可执行指令。也应当注意,在有些作为替换的实现中,方框中所标注的功能也可以以不同于附图中所标注的顺序发生。例如,两个接连地表示的方框实际上可以基本并行地执行,它们有时也可以按相反的顺序执行,这依所涉及的功能而定。也要注意,框图或流程图中的每个方框、以及框图或流程图中的方框的组合,可以用执行规定的功能或操作的专用的基于硬件的系统来实现,或者可以用专用硬件与计算机指令的组合来实现。

[0105] 作为另一方面,本公开还提供了一种计算机可读介质,该计算机可读介质可以是上述实施例中描述的设备中所包含的;也可以是单独存在,而未装配入该设备中。上述计算机可读介质承载有一个或者多个程序,当上述一个或者多个程序被一个该设备执行时,使得该设备执行图片检测方法。

[0106] 以上对本公开的实施例进行了描述。但是,这些实施例仅仅是为了说明的目的,而并非为了限制本公开的范围。尽管在以上分别描述了各实施例,但是这并不意味着各个实施例中的措施不能有利地结合使用。本公开的范围由所附权利要求及其等同物限定。不脱离本公开的范围,本领域技术人员可以做出多种替代和修改,这些替代和修改都应落在本公开的范围之内。

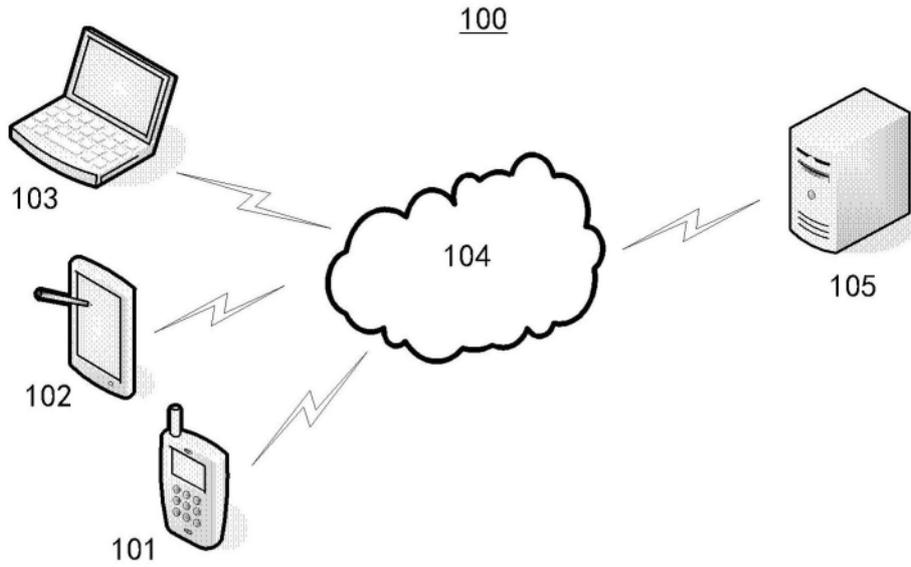


图1

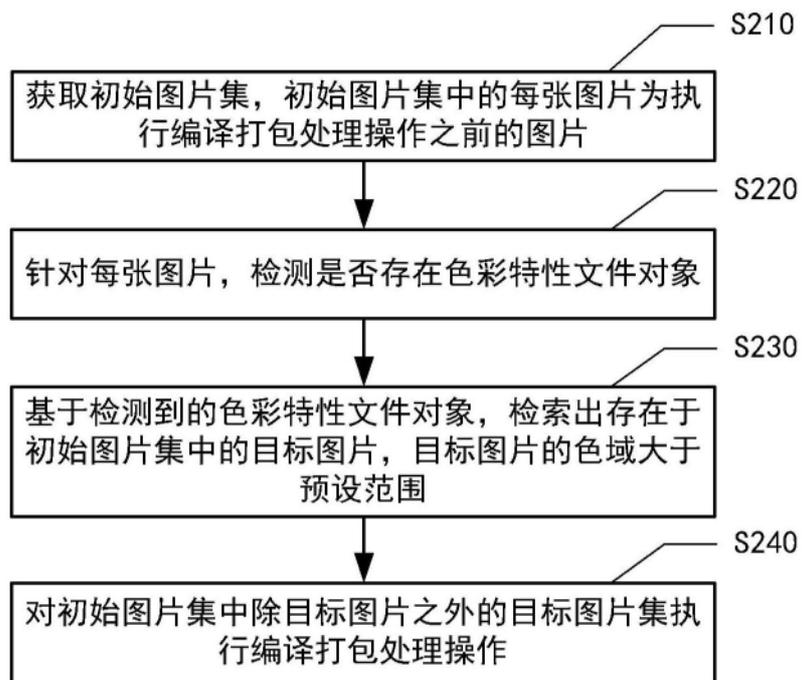


图2

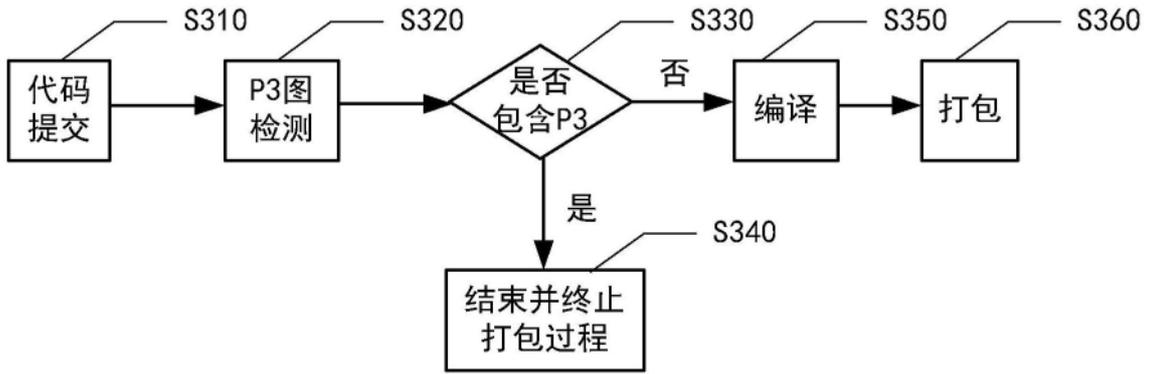


图3

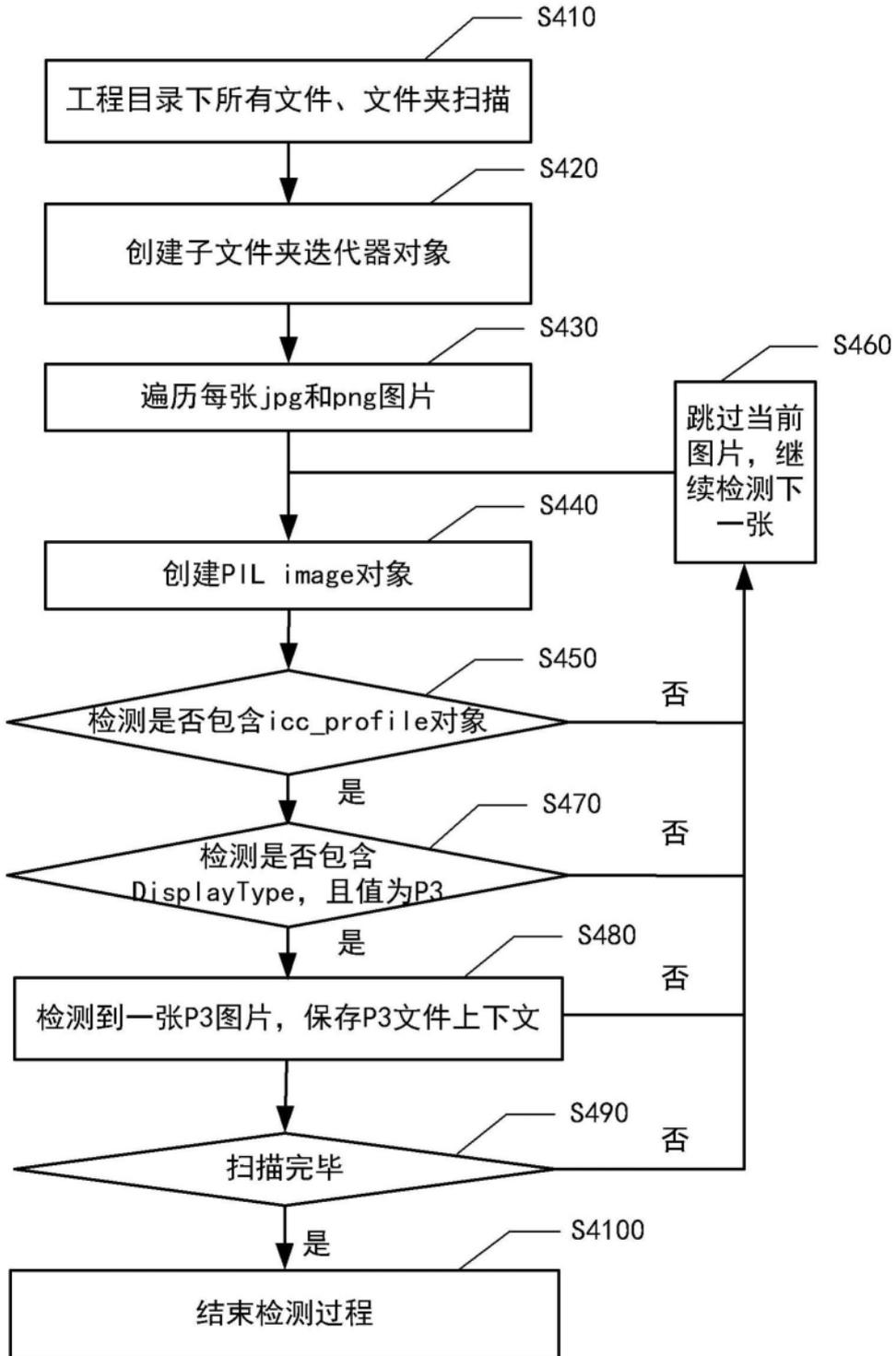


图4

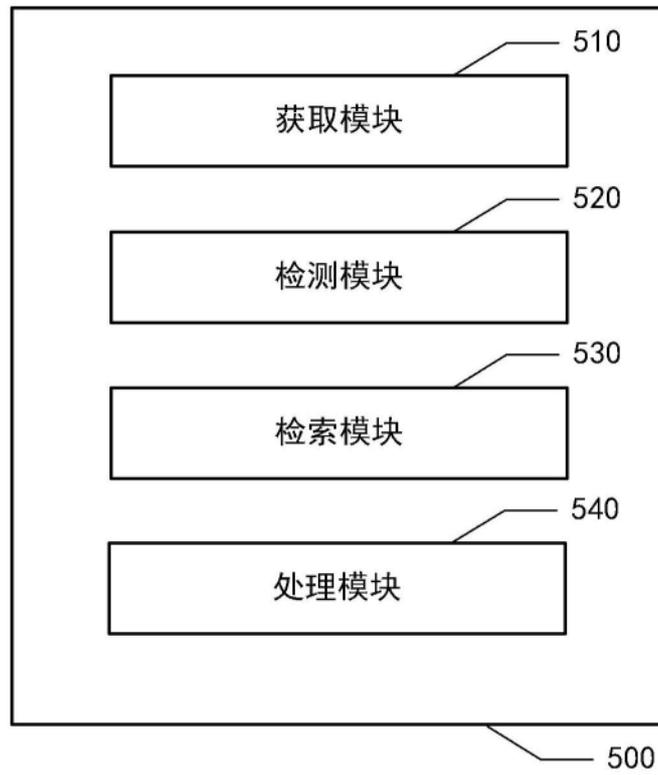


图5

600

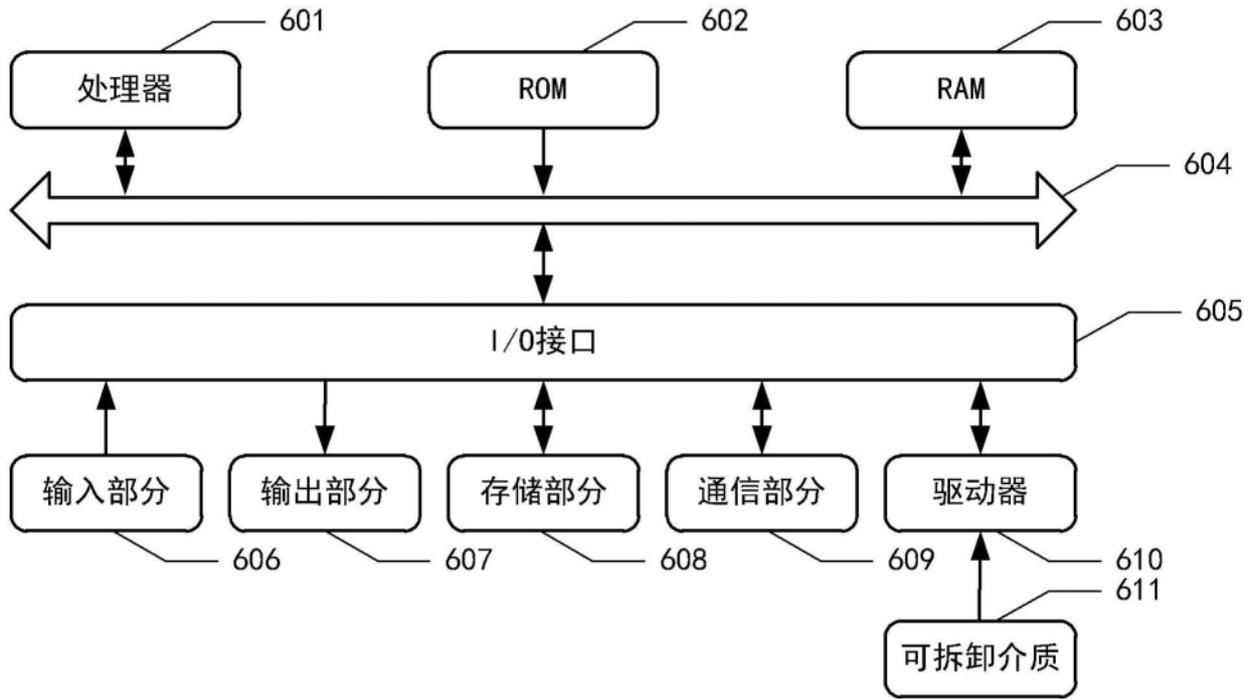


图6