

(12) 按照专利合作条约所公布的国际申请

(19) 世界知识产权组织
国际局

(43) 国际公布日
2023 年 11 月 23 日 (23.11.2023)



(10) 国际公布号
WO 2023/224547 A2

- (51) 国际专利分类号:
无分类
- (21) 国际申请号: PCT/SG2023/050306
- (22) 国际申请日: 2023 年 5 月 5 日 (05.05.2023)
- (25) 申请语言: 中文
- (26) 公布语言: 中文
- (30) 优先权:
202210530042.0 2022年5月16日 (16.05.2022) CN
- (71) 申请人: 脸萌有限公司 (LEMON INC.) [GB/SG]; 开曼群岛大开曼西湾路 802 号木槿道大楼邮政信箱 31119, Grand Cayman KY1 - 1205 (KY)。
- (72) 发明人: 张乐林 (ZHANG, Lelin); 美国加利福尼亚州洛杉矶西杰弗逊大道 12655 号六层第 137 室,

California 90066 (US)。李小奇 (LI, Xiaoqi); 中国北京市朝阳区七圣中街 12 号院融中心 B1 小邮局, Beijing 100028 (CN)。刘策龙 (LIU, Celong); 美国加利福尼亚州洛杉矶西杰弗逊大道 12655 号六层第 137 室, California 90066 (US)。王帅 (WANG, Shuai); 中国北京市朝阳区七圣中街 12 号院融中心 B1 小邮局, Beijing 100028 (CN)。黄攀 (HUANG, Pan); 中国北京市朝阳区七圣中街 12 号院融中心 B1 小邮局, Beijing 100028 (CN)。黄欣 (HUANG, Xin); 中国北京市朝阳区七圣中街 12 号院融中心 B1 小邮局, Beijing 100028 (CN)。代增 (DAI, Zeng); 美国加利福尼亚州洛杉矶西杰弗逊大道 12655 号六层第 137 室, California 90066 (US)。梅星 (MEI, Xing); 美国加利福尼亚州洛杉矶西杰弗逊大道 12655 号六层第 137 室, California 90066 (US)。

(54) Title: DATA PROCESSING METHOD AND APPARATUS, DEVICE AND STORAGE MEDIUM

(54) 发明名称: 数据处理方法、装置、设备及存储介质

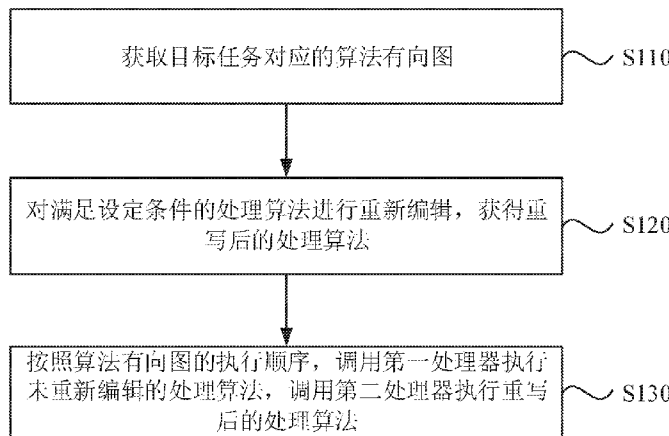


图 1

- S110 Acquire an algorithm directed graph corresponding to a target task
- S120 Re-edit a processing algorithm that meets a set condition, so as to obtain a rewritten processing algorithm
- S130 According to an execution sequence of the algorithm directed graph, call a first processor to execute a non-reedited processing algorithm, and call a second processor to execute the rewritten processing algorithm

(57) Abstract: The embodiments of the present disclosure disclose a data processing method and apparatus, a device and a storage medium. The method comprises: acquiring an algorithm directed graph corresponding to a target task, wherein the algorithm directed graph includes a plurality of algorithm nodes, and one algorithm node corresponds to one processing algorithm; re-editing the processing algorithm that meets a set condition, so as to obtain a rewritten processing algorithm; and according to an execution sequence of the algorithm directed graph, calling a first processor to execute a non-reedited processing algorithm, and calling a second processor to

(74) 代理人:傅子健(**POH, Chee Kian, Daniel**); 新加坡新加坡麥仕奇新加坡,丹戎巴葛邮局,POBox636, Singapore 910816 (SG)。

(81) 指定国(除另有指明, 要求每一种可提供的国家保护): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CV, CZ, DE, DJ, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IQ, IR, IS, IT, JM, JO, JP, KE, KG, KH, KN, KP, KR, KW, KZ, LA, LC, LK, LR, LS, LU, LY, MA, MD, MG, MK, MN, MU, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, WS, ZA, ZM, ZW。

(84) 指定国(除另有指明, 要求每一种可提供的地区保护): ARIPO (BW, CV, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SC, SD, SL, ST, SZ, TZ, UG, ZM, ZW), 欧亚 (AM, AZ, BY, KG, KZ, RU, TJ, TM), 欧洲 (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, ME, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML, MR, NE, SN, TD, TG)。

本国际公布:

— 不包括国际检索报告, 在收到该报告后将重新公布(细则48.2(g))。

execute the rewritten processing algorithm.

(57) 摘要: 说明书摘要本公开实施例公开了一种数据处理方法、装置、设备及存储介质。该方法包括: 获取目标任务对应的算法有向图; 其中, 所述算法有向图包含多个算法节点, 一个算法节点对应一种处理算法; 对满足设定条件的处理算法进行重新编辑, 获得重写后的处理算法; 按照所述算法有向图的执行顺序, 调用第一处理器执行未重新编辑的处理算法, 调用第二处理器执行重写后的处理算法。

数据处理方法、装置、设备及存储介质

本公开要求在 2022 年 5 月 16 日提交中国专利局、申请号为 202210530042.0 的中国专利申请的优先权，以上申请的全部内容通过引用结合在本公开中。

技术领域

本公开实施例涉及计算机技术领域，例如涉及一种数据处理方法、装置、设备及存储介质。

背景技术

对于一个图像处理任务，通常需要调用多个算法实现。在运行这些算法时存在多次中央处理器（central processing unit, CPU）和图形处理器（graphics processing unit, GPU）之间的切换，而 CPU 和 GPU 之间的切换需要进行数据拷贝，按照相关技术的方案，需要进行多次数据拷贝，这样存在较大的计算和内存开销，在一定程度上影响整个任务的处理效率。

发明内容

本公开实施例提供一种数据处理方法、装置、设备及存储介质，可以减少 CPU 和 GPU 的切换次数，从而节省计算和内存资源，且可以提高数据处理的效率。

第一方面，本公开实施例提供了一种数据处理方法，包括：

获取目标任务对应的算法有向图；其中，所述算法有向图包含多个算法节点，一个算法节点对应一种处理算法；

对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；

按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

第二方面，本公开实施例还提供了一种数据处理装置，包括：

算法有向图获取模块，设置为获取目标任务对应的算法有向图；其中，所述算法有向图包含多个算法节点，一个算法节点对应一种处理算法；

处理算法编辑模块，设置为对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；

处理器调用模块，设置为按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

第三方面，本公开实施例还提供了一种电子设备，所述电子设备包括：

一个或多个处理装置；

存储装置，设置为存储一个或多个程序；

当所述一个或多个程序被所述一个或多个处理装置执行，使得所述一个或多个处理装置实现如本公开实施例所述的数据处理方法。

第四方面，本公开实施例还提供了一种计算机可读介质，所述计算机可读介质上存储有计算机程序，该计算机程序被处理装置执行时实现如本公开实施例所述的数据处理方法。

附图说明

图1是本公开实施例中的一种数据处理方法的流程图；

图2是本公开实施例中的一种算法有向图的示例图；

图3是本公开实施例中的一种算法有向图示例图；

图4是本公开实施例中的目标任务实例的处理过程示例图；

图5是本公开实施例中的一种数据处理装置的结构示意图；

图6是本公开实施例中的一种电子设备的结构示意图。

具体实施方式

下面将参照附图描述本公开的实施例。虽然附图中显示了一些实施例，然而应当理解的是，本公开可以通过各种形式来实现。应当理解的是，本公开的附图及实施例仅用于示例性作用。

应当理解，本公开的方法实施方式中记载的各个步骤可以按照不同的顺序执行，和/或并行执行。此外，方法实施方式可以包括附加的步骤和/或省略执行示出的步骤。

本文使用的术语“包括”及其变形是开放性包括，即“包括但不限于”。术语“基于”是“至少部分地基于”。术语“一个实施例”表示“至少一个实施例”；术语“另一实施例”表示“至少一个另外的实施例”；术语“一些实施例”表示“至少一些实施例”。其他术语的相关定义将在下文描述中给出。

需要注意，本公开中提及的“第一”、“第二”等概念仅用于对不同的装置、模块或单元进行区分，并非用于限定这些装置、模块或单元所执行的功能的顺序或者相互依存关系。

需要注意，本公开中提及的“一个”、“多个”的修饰是示意性的，本领域技术人员应当理解，除非在上下文另有明确指出，否则应该理解为“一个或多个”。

本公开实施方式中的多个装置之间所交互的消息或者信息的名称仅用于说明性的目的，而并不是用于对这些消息或信息的范围进行限制。

智能创作：特指视频类社交平台中所使用的基于计算机视觉和图形学的图像视频内容生成方法，通过人工智能（例如传统机器学习或深度学习）和虚拟现实/增强现实技术的应用，使用户提供的视频内容具备更加多元丰富的内容。

算法平台：在用户使用移动平台或其他个人计算机(Personal Computer, PC)平台进行智能创作时所使用的支持算法调度与执行的软件系统，该系统的输入为来自相机的图片或视频信息以及所需要运行的算法，这些算法的执行顺序与依赖关系通过有向图来描述和连接，该系统的输出为算法运行的结果，包括图像分类信息、目标物体检测包围框及置信度、物体分割信息、生成图像、人体或物体关键点信息等。

本方案的目标是使得算法有向图中的设定类别的算法可以由设定处理器（如GPU）执行，使得数据在算法有向图的传输过程中可以完全（或者大部分）在设定处理器上执行，通过减少数据在不同处理器间的复制次数，达到节省计算和内存资源的效果。本实施例中，在智能创作中，使算法有向图中各个算法节点在支持CPU功能的同时，实现在GPU上的功能扩展，使得设定类别的算法可以在GPU上运行。

图1为本公开实施例提供的一种数据处理方法的流程图，本实施例可适用于调用处理器对数据进行处理的情况，该方法可以由数据处理装置来执行，该装置可由硬件和/或软件组成，并一般可集成在具有数据处理功能的设备中，该设备可以是服务器、移动终端或服务器集群等电子设备。如图1所示，该方法可包括如下步骤：

S110，获取目标任务对应的算法有向图。

其中，算法有向图包含多个算法节点，一个算法节点对应一种处理算法。多个算法节点之间通过有向边连接，有向边两端的算法节点具有依赖关系，有向边结束端的算法节点依赖起始端的算法节点。示例性的，图2是本实施例中的一种算法有向图的示例图，如图2所示，该算法有向图包含5个算法节点，其中，

算法节点2和算法节点3均依赖算法节点1，算法节点4依赖算法节点2和算法节点3，算法节点5依赖算法节点4。

其中，目标任务可以是需要调用多种算法完成的数据处理任务，可以是图像处理任务或者音频处理任务等。本实施例中，主要是针对图像处理任务的方案。

示例性的，获取目标任务对应的算法有向图的方式可以是：获取目标任务所需的多个处理算法；确定多个处理算法的依赖关系；基于依赖关系建立算法有向图。

其中，获取目标任务所需的多个处理算法的过程可以是：首先确定目标任务所需处理的图像或者音频的初始状态以及目标状态，然后基于初始状态和目标状态将目标任务划分为多个阶段，然后确定出每个阶段所需调用的处理算法。

确定多个处理算法的依赖关系的方式可以是：首先确定目标任务所需的多个处理算法的执行顺序，然后根据执行顺序确定出处理算法间的依赖关系。

基于依赖关系建立算法有向图的方式可以是：在处理算法对应的算法节点之间添加有向边。其中，有向边的起始端设置被依赖的算法节点，有向边的结束端设置依赖的算法节点。示例性的，一目标任务是对图像进行特效处理，其经历的阶段为：首先对图像中的人体面部进行检测，然后对面部中的关键点进行检测，再然后根据关键点确定眼睛位置和嘴巴位置，再然后将眼睛和嘴巴截取出来，最后对眼睛和嘴巴进行形变处理。从上述可以看出，该目标任务需要调用的算法包括：面部检测算法、关键点检测算法、眼睛检测算法、嘴巴检测算法、图像分割算法、图像形变算法。且各算法的依赖关系为：关键点检测算法依赖面部检测算法的处理结果、眼睛检测算法和嘴巴检测算法依赖关键点检测算法的处理结果，图像分割算法依赖眼睛检测算法和嘴巴检测算法的处理结果，图像形变算法依赖图像分割算法的处理结果。因此确定出的算法有向图参见图3。本实施例的技术方案，基于处理算法间的依赖关系建立算法有向图，可以提高算法有向图的准确性。

S120，对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法。

其中，设定条件可以是处理算法为设定类别。设定类别可以是关键点检测类（Landmark Detection）、图像分割类（Segmentation）及图像变换类（如生成对抗网络（Generative adversarial network, GAN））。设定类别的处理算法具有的特性是输出的数据可以以图像（即可渲染的纹理）的形式表征。例如：GAN类可以输出红色、绿色、蓝色（Red、Green、Blue, RGB）或者红色、绿色、蓝色、不透明度（Red、Green、Blue, Alpha, RGBA）类型的数据，Segmentation

类可以输出多维归一化的灰度图（即掩膜），Landmark Detection类可以输出每个关键点的多维归一化的灰度图。

本实施例中，对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法的过程可以是：将满足设定条件的处理算法按照第二处理器支持的格式进行重新编辑，并将处理算法的输入接口和输出接口改写为第二处理器支持的接口，获得重写后的处理算法。

其中，第二处理器可以是GPU。示例性的，首先获取满足设定条件的处理算法的源代码，逐行读取源代码，提取源代码中的算法参数，最后将提取出的算法参数按照GPU支持的格式进行重新编辑。其中，第二处理器支持的接口可以是纹理接口。示例性的，表1是改写后的接口格式：

表1

类型	纹理标识（ID）	数据尺寸/数据顺序	数据格式
输入	int	NHWC	u8, int8, int16, uint16, float, fp16, double; 支持的图像通道类型：RGB, RGBA, R
输出	int	NHWC	

从表1可以看出，改写后的接口支持的数据类别可以是RGB、RGBA或者R。数据尺寸可以表示为NHWC，其中，N表示GPU的并行度，H表示图像的高，W表示图像的宽，C表示图像的通道数。本实施例中，将满足设定条件的处理算法改写为GPU支持的格式，使得处理算法由GPU执行，使得数据在算法有向图的传输过程中可以完全（或者大部分）在GPU上执行，减少数据在CPU和GPU间的复制次数。

S130，按照算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

例如，第一处理器可以是CPU，第二处理器可以是GPU。对于不满足设定条件的处理算法，无法对其编写格式进行编写，仍然需要调用CPU执行。例如，对于图像分类算法（Classification）以及目标检测类（Object Detection），由于其输出数据为向量格式而非图像格式，因此继续沿用原有CPU接口。

本实施例中，通过创建的线程调用第一处理器和第二处理器执行对应的处理算法。可以创建一个线程，按照算法有向图的执行顺序，通过创建的线程调用第一处理器执行未重新编辑的处理算法，以及通过创建的线程调用第二处理

器执行重写后的处理算法。为了提高处理效率，也可以创建不同的线程，对不同的线程分配任务，以调用第一处理器或者第二处理器执行对应的处理算法。

示例性的，按照算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法的方式可以是：将算法有向图中的处理算法划分为前后两组，获得前算法组和后算法组；对前算法组创建第一线程，通过第一线程调用第一处理器和/或第二处理器执行前算法组内的处理算法；对后算法组创建第二线程，通过第二线程调用第一处理器和/或第二处理器执行后算法组内的处理算法。

其中，将算法有向图中的处理算法划分为前后两组可以理解为：将算法有向图中靠前的处理算法划分为一组，将靠后的处理算法划分为一组。

其中，将算法有向图中的处理算法划分为前后两组的方式可以是：获取算法有向图中排序靠后且连续为重写后的处理算法的第一数量；若第一数量小于或等于算法有向图中总节点数量的一半，则将第一数量的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组；若第一数量大于算法有向图中总节点数量的一半，则将总节点数量一半的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组。

本实施例，在将满足设定条件的处理算法重新编辑后，算法有向图中的大部分或者全部处理算法可能被重新编辑，排在算法有向图中靠前的处理算法（即处于前处理阶段的处理算法）可能不满足设定条件，即未被重新编辑。因此，可以在算法有向图中获取到排序靠后且连续为重写后的处理算法。

若第一数量小于或等于算法有向图中总节点数量的一半，则将第一数量的重写后的处理算法直接作为后算法组，剩余的处理算法划分为前算法组。这样划分出的前算法组包含未重新编辑的处理算法和/或重写后的处理算法，后算法组只包含重写后的处理算法。对前算法组创建第一线程，通过第一线程调用第一处理器和/或第二处理器执行前算法组内的处理算法。对后算法组创建第二线程，通过第二线程调用第二处理器执行前算法组内的处理算法。将连续的重写后的处理算法划分为一组，可以避免第二线程一直调用第二处理器，无需切换调用其他处理器。

若第一数量大于算法有向图中总节点数量的一半，则将总节点数量一半的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组。这样划分出的前算法组包含未重新编辑的处理算法和/或重写后的处理算法，后算法组只包含重写后的处理算法，且前后两组包含的节点数量相近。对前算法组创建第一线程，通过第一线程调用第一处理器和/或第二处理

器执行前算法组内的处理算法。对后算法组创建第二线程，通过第二线程调用第二处理器执行前算法组内的处理算法。本实施例中，前后两组包含的节点数量相近，可以保证资源的分配均衡。

示例性的，通过第一线程调用第一处理器和/或第二处理器执行前算法组内的处理算法的过程可以是：按照前算法组的执行顺序，通过第一线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法；通过第二线程调用第一处理器和/或第二处理器执行后算法组内的处理算法的过程可以是：按照后算法组的执行顺序，通过第二线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

本实施例中，当执行前算法组内的处理算法时，启动第一线程，按照前算法组的执行顺序，若执行到未重新编辑的处理算法，则调用第一处理器执行；若执行到重写后的处理算法，则调用第二处理器执行。本应用场景下，由于后算法组只包含重写后的处理算法，因此当执行到后算法组的处理算法时，启动第二线程，按照前算法组的执行顺序，调用第二处理器依次执行。本实施例，按照算法有向图的顺序依次执行处理算法，可以保证顺利准确地完成目标任务。

可选的，按照算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法的方式可以是：对未重新编辑的处理算法创建第三线程，对重写后的处理算法创建第四线程；按照算法有向图的执行顺序，通过第三线程调用第一处理器执行未重新编辑的处理算法，通过第四线程调用第二处理器执行重写后的处理算法。

本实施例中，将未重新编辑的处理算法划分为一组，将重写后的处理算法划分为另一组。对未重新编辑的处理算法创建一个线程，即第三线程，对重写后的处理算法创建一个线程，即第四线程。示例性的，按照算法有向图的执行顺序，当执行到未重新编辑的处理算法时，启动第三线程（此时第四线程休眠），以调用第一处理器执行该处理算法；当执行到重写后的处理算法，启动第四线程（此时第三线程休眠），以调用第二处理器执行该处理算法。本实施例中，为同类型的处理算法创建一个线程，使得线程在整个任务执行过程中，调用同一个处理器，可以避免线程调用处理器时的切换频率。

示例性的，以“泡沫特效渲染”为例，所需要的处理算法依次为：纹理输入（texture blit）、面部检测（face detect）、面部校准（face align）、纹理渲染（render）、图像转换（nh-image transform）、图像推理（nh-inference）及图像后处理（nh-post process）。图4是目标任务实例的处理过程示例图。如图4所示，在该算法有向图中，其中，前三个处理算法：texture blit、face detect及face align调用CPU执行，后四个处理算法：render、nh-image transform、nh-inference及

nh-post process调用GPU执行。从图4中可以看出，该有向图的链路中只进行了一次CPU和GPU间的切换，从而有效的减少了CPU和GPU之间的数据交换，极大的增加了数据处理效率。

本公开实施例的技术方案，获取目标任务对应的算法有向图；其中，算法有向图包含多个算法节点，一个算法节点对应一种处理算法；对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；按照算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。本公开实施例提供的数据处理方法，对满足设定条件的处理算法重新编辑，使得重写后的处理算法可以被第二处理器执行，可以在执行算法有向图中的处理算法时减少第一处理器和第二处理器的切换次数，从而节省计算和内存资源，且可以提高数据处理的效率。

图5是本公开实施例公开的一种数据处理装置的结构示意图，如图5所示，该装置包括：

算法有向图获取模块510，设置为获取目标任务对应的算法有向图；其中，算法有向图包含多个算法节点，一个算法节点对应一种处理算法；

处理算法编辑模块520，设置为对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；

处理器调用模块530，设置为按照算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

可选的，设定条件为处理算法为设定类别，处理算法编辑模块520，设置为通过以下方式对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法：

将满足设定条件的处理算法按照第二处理器支持的格式进行重新编辑，并将处理算法的输入接口和输出接口改写为第二处理器支持的接口，获得重写后的处理算法。

可选的，设定类别包括如下任意一种：关键点检测类、图像分割类及图像变换类。

可选的，处理器调用模块530，设置为通过以下方式调用处理器执行处理算法：

将算法有向图中的处理算法划分为前后两组，获得前算法组和后算法组；

对前算法组创建第一线程，通过第一线程调用第一处理器和/或第二处理器执行前算法组内的处理算法；

对后算法组创建第二线程，通过第二线程调用第一处理器和/或第二处理器执行后算法组内的处理算法。

可选的，处理器调用模块530，设置为通过以下方式获得前算法组和后算法组：

获取算法有向图中排序靠后且连续为重写后的处理算法的第一数量；

若第一数量小于或等于算法有向图中总节点数量的一半，则将第一数量的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组；

若第一数量大于算法有向图中总节点数量的一半，则将总节点数量一半的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组。

可选的，处理器调用模块530，设置为通过以下方式按照前算法组的执行顺序调用处理器执行处理算法：

按照前算法组的执行顺序，通过第一线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

可选的，处理器调用模块530，设置为通过以下方式按照后算法组的执行顺序调用处理器执行处理算法：

按照后算法组的执行顺序，通过第二线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

可选的，处理器调用模块530，设置为通过以下方式创建线程并通过过程调用处理器执行处理算法：

对未重新编辑的处理算法创建第三线程，对重写后的处理算法创建第四线程；

按照算法有向图的执行顺序，通过第三线程调用第一处理器执行未重新编辑的处理算法，通过第四线程调用第二处理器执行重写后的处理算法。

上述装置可执行本公开前述所有实施例所提供的方法，具备执行上述方法相应的功能模块和有益效果。未在本实施例中详尽描述的技术细节，可参见本公开前述所有实施例所提供的方法。

下面参考图6，其示出了适于用来实现本公开实施例的电子设备300的结构示意图。本公开实施例中的电子设备可以包括诸如移动电话、笔记本电脑、数字广播接收器、个人数字助理（Personal Digital Assistant, PDA）、平板电脑（Portable Android Device, PAD）、便携式多媒体播放器（Portable Media Player,

PMP)、车载终端(例如车载导航终端)等的移动终端以及诸如数字TV、台式计算机等的固定终端,或者各种形式的服务器,如独立服务器或者服务器集群。图6示出的电子设备仅仅是一个示例。

如图6所示,电子设备300可以包括处理装置(例如中央处理器、图形处理器等)301,处理装置301可以根据存储在只读存储装置(Read Only Memory, ROM)302中的程序或者从存储装置308加载到随机访问存储装置(Random Access Memory, RAM)303中的程序而执行各种适当的动作和处理。在RAM 303中,还存储有电子设备300操作所需的各种程序和数据。处理装置301、ROM 302以及RAM 303通过总线304彼此相连。输入/输出(Input/Output, I/O)接口305也连接至总线304。

通常,以下装置可以连接至I/O接口305:包括例如触摸屏、触摸板、键盘、鼠标、摄像头、麦克风、加速度计、陀螺仪等的输入装置306;包括例如液晶显示器(Liquid Crystal Display, LCD)、扬声器、振动器等输出装置307;包括例如磁带、硬盘等的存储装置308;以及通信装置309。通信装置309可以允许电子设备300与其他设备进行无线或有线通信以交换数据。虽然图6示出了具有各种装置的电子设备300,但是应理解的是,并不要求实施或具备所有示出的装置。可以替代地实施或具备更多或更少的装置。

在一实施例中,根据本公开的实施例,上文参考流程图描述的过程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数据处理方法的程序代码。在这样的实施例中,该计算机程序可以通过通信装置309从网络上被下载和安装,或者从存储装置308被安装,或者从ROM 302被安装。在该计算机程序被处理装置301执行时,执行本公开实施例的方法中限定的上述功能。

需要说明的是,本公开上述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组合。计算机可读存储介质例如可以是电、磁、光、电磁、红外线、或半导体的系统、装置或器件,或者以上的组合。计算机可读存储介质的示例可以包括: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(如电子可编程只读存储器(Electronic Programable Read Only Memory, EPROM)或闪存)、光纤、便携式紧凑磁盘只读存储器(Compact Disc-Read Only Memory, CD-ROM)、光存储器件、磁存储器件、或者上述的合适的组合。在本公开中,计算机可读存储介质可以是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，包括电磁信号、光信号或上述的合适的组合。计算机可读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，该计算机可读信号介质可以发送、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：电线、光缆、射频（Radio Frequency, RF）等，或者上述的合适的组合。

在一些实施方式中，客户端、服务器可以利用诸如超文本传输协议（HyperText Transfer Protocol, HTTP）之类的任何当前已知或未来研发的网络协议进行通信，并且可以与任意形式或介质的数字数据通信（例如，通信网络）互连。通信网络的示例包括局域网（Local Area Network, LAN），广域网（Wide Area Network, WAN），网际网（例如，互联网）以及端对端网络（例如，ad hoc 端对端网络），以及当前已知或未来研发的网络。

上述计算机可读介质可以是上述电子设备中所包含的；也可以是单独存在，而未装配入该电子设备中。

上述计算机可读介质承载有至少一个程序，当上述至少一个程序被该电子设备执行时，使得该电子设备：获取目标任务对应的算法有向图；其中，所述算法有向图包含多个算法节点，一个算法节点对应一种处理算法；对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

可以以一种或多种程序设计语言或其组合来编写用于执行本公开的操作的计算机程序代码，上述程序设计语言包括面向对象的程序设计语言诸如Java、Smalltalk、C++，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机上执行、部分地在用户计算机上执行、作为一个独立的软件包执行、部分在用户计算机上部分在远程计算机上执行、或者完全在远程计算机或服务器上执行。在涉及远程计算机的情形中，远程计算机可以通过任意种类的网络包括局域网(LAN)或广域网(WAN)连接到用户计算机，或者，可以连接到外部计算机（例如利用因特网服务提供商来通过因特网连接）。

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，该模块、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。也要注意的，框图和/或流程图中的每个方框、以及框图和/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。

描述于本公开实施例中所涉及到的单元可以通过软件的方式实现，也可以通过硬件的方式来实现。其中，单元的名称在某种情况下并不构成对该单元本身的限定。

本文中以上描述的功能可以至少部分地由至少一个硬件逻辑部件来执行。例如，可以使用的示范类型的硬件逻辑部件包括：现场可编程门阵列（Field-Programmable Gate Array, FPGA）、专用集成电路（Application Specific Integrated Circuit, ASIC）、专用标准产品（Application Specific Standard Parts, ASSP）、片上系统（System on Chip, SOC）、复杂可编程逻辑设备（Complex Programmable Logic Device, CPLD）等。

在本公开的上下文中，机器可读介质可以是有形的介质，其可以包含或存储以供指令执行系统、装置或设备使用或与指令执行系统、装置或设备结合地使用的程序。机器可读介质可以是机器可读信号介质或机器可读储存介质。机器可读介质可以包括电子的、磁性的、光学的、电磁的、红外的、或半导体系统、装置或设备，或者上述内容的合适组合。机器可读存储介质的示例可以包括基于一个或多个线的电气连接、便携式计算机盘、硬盘、随机存取存储器（RAM）、只读存储器（ROM）、可擦除可编程只读存储器（EPROM或快闪存储器）、光纤、便捷式紧凑盘只读存储器（CD-ROM）、光学储存设备、磁储存设备、或上述内容的合适组合。

根据本公开实施例的一个或多个实施例，本公开实施例公开了一种数据处理方法，包括：

获取目标任务对应的算法有向图；其中，所述算法有向图包含多个算法节点，一个算法节点对应一种处理算法；

对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；

按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

进一步地，所述设定条件为处理算法为设定类别，对满足设定条件的处理

算法进行重新编辑，获得重写后的处理算法，包括：

将满足设定条件的处理算法按照所述第二处理器支持的格式进行重新编辑，并将所述处理算法的输入接口和输出接口改写为所述第二处理器支持的接口，获得重写后的处理算法。

进一步地，所述设定类别包括如下任意一种：关键点检测类、图像分割类及图像变换类。

进一步地，按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法，包括：

将所述算法有向图中的处理算法划分为前后两组，获得前算法组和后算法组；

对所述前算法组创建第一线程，通过所述第一线程调用第一处理器和/或第二处理器执行所述前算法组内的处理算法；

对所述后算法组创建第二线程，通过所述第二线程调用第一处理器和/或第二处理器执行所述后算法组内的处理算法。

进一步地，将所述算法有向图中的处理算法划分为前后两组，包括：

获取所述算法有向图中排序靠后且连续为重写后的处理算法的第一数量；

若所述第一数量小于或等于所述算法有向图中总节点数量的一半，则将所述第一数量的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组；

若所述第一数量大于所述算法有向图中总节点数量的一半，则将所述总节点数量一半的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组。

进一步地，通过所述第一线程调用第一处理器和/或第二处理器执行所述前算法组内的处理算法，包括：

按照所述前算法组的执行顺序，通过所述第一线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法；

通过所述第二线程调用第一处理器和/或第二处理器执行所述后算法组内的处理算法，包括：

按照所述后算法组的执行顺序，通过所述第二线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

进一步地，按照所述算法有向图的执行顺序，调用第一处理器执行未重新

编辑的处理算法，调用第二处理器执行重写后的处理算法，包括：

对未重新编辑的处理算法创建第三线程，对重写后的处理算法创建第四线程；

按照所述算法有向图的执行顺序，通过所述第三线程调用第一处理器执行未重新编辑的处理算法，通过所述第四线程调用第二处理器执行重写后的处理算法。

应该理解，可以使用上面所示的各种形式的流程，重新排序、增加或删除步骤。例如，本公开中记载的各步骤可以并行地执行也可以顺序地执行也可以不同的次序执行，只要能够实现本公开的技术方案所期望的结果。

权利要求书

1. 一种数据处理方法，包括：

获取目标任务对应的算法有向图；其中，所述算法有向图包含多个算法节点，一个算法节点对应一种处理算法；

对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；

按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

2. 根据权利要求1所述的方法，其中，所述设定条件为处理算法为设定类别，所述对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法，包括：

将满足设定条件的处理算法按照所述第二处理器支持的格式进行重新编辑，并将所述处理算法的输入接口和输出接口改写为所述第二处理器支持的接口，获得重写后的处理算法。

3. 根据权利要求2所述的方法，其中，所述设定类别包括如下任意一种：关键点检测类、图像分割类及图像变换类。

4. 根据权利要求1所述的方法，其中，按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法，包括：

将所述算法有向图中的处理算法划分为前后两组，获得前算法组和后算法组；

对所述前算法组创建第一线程，通过所述第一线程调用第一处理器和第二处理器中的至少之一执行所述前算法组内的处理算法；

对所述后算法组创建第二线程，通过所述第二线程调用第一处理器和第二处理器中的至少之一执行所述后算法组内的处理算法。

5. 根据权利要求4所述的方法，其中，所述将所述算法有向图中的处理算法划分为前后两组，包括：

获取所述算法有向图中排序靠后且连续为重写后的处理算法的第一数量；

响应于所述第一数量小于或等于所述算法有向图中总节点数量的一半，将所述第一数量的重写后的处理算法划分为一组，作为后算法组，剩余的处理算法划分为一组，作为前算法组；

响应于所述第一数量大于所述算法有向图中总节点数量的一半，将所述总节点数量一半的重写后的处理算法划分为一组，作为后算法组，剩余的处理算

法划分为一组，作为前算法组。

6. 根据权利要求 4 所述的方法，其中，所述通过所述第一线程调用第一处理器和第二处理器中的至少之一执行所述前算法组内的处理算法，包括：

按照所述前算法组的执行顺序，通过所述第一线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法；

所述通过所述第二线程调用第一处理器和第二处理器中的至少之一执行所述后算法组内的处理算法，包括：

按照所述后算法组的执行顺序，通过所述第二线程调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

7. 根据权利要求 1 所述的方法，其中，所述按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法，包括：

对未重新编辑的处理算法创建第三线程，对重写后的处理算法创建第四线程；

按照所述算法有向图的执行顺序，通过所述第三线程调用第一处理器执行未重新编辑的处理算法，通过所述第四线程调用第二处理器执行重写后的处理算法。

8. 一种数据处理装置，包括：

算法有向图获取模块，设置为获取目标任务对应的算法有向图；其中，所述算法有向图包含多个算法节点，一个算法节点对应一种处理算法；

处理算法编辑模块，设置为对满足设定条件的处理算法进行重新编辑，获得重写后的处理算法；

处理器调用模块，设置为按照所述算法有向图的执行顺序，调用第一处理器执行未重新编辑的处理算法，调用第二处理器执行重写后的处理算法。

9. 一种电子设备，所述电子设备包括：

一个或多个处理装置；

存储装置，设置为存储一个或多个程序；

当所述一个或多个程序被所述一个或多个处理装置执行，使得所述一个或多个处理装置实现如权利要求 1-7 中任一所述的数据处理方法。

10. 一种计算机可读介质，所述计算机可读介质上存储有计算机程序，所述计算机程序被处理装置执行时实现如权利要求 1-7 中任一所述的数据处理方法。

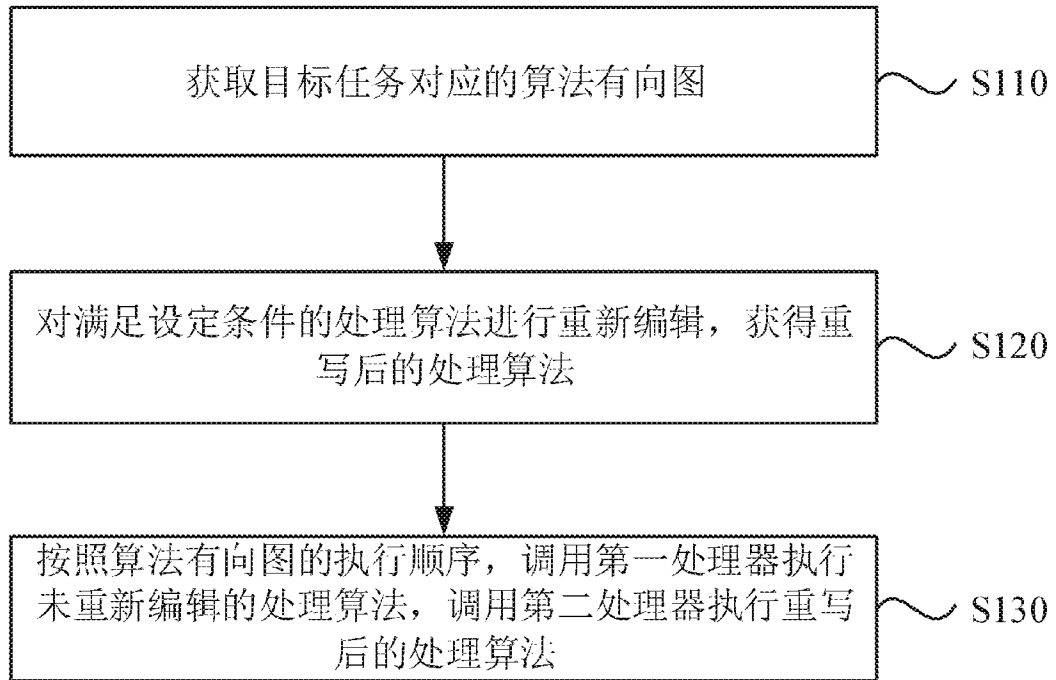


图 1

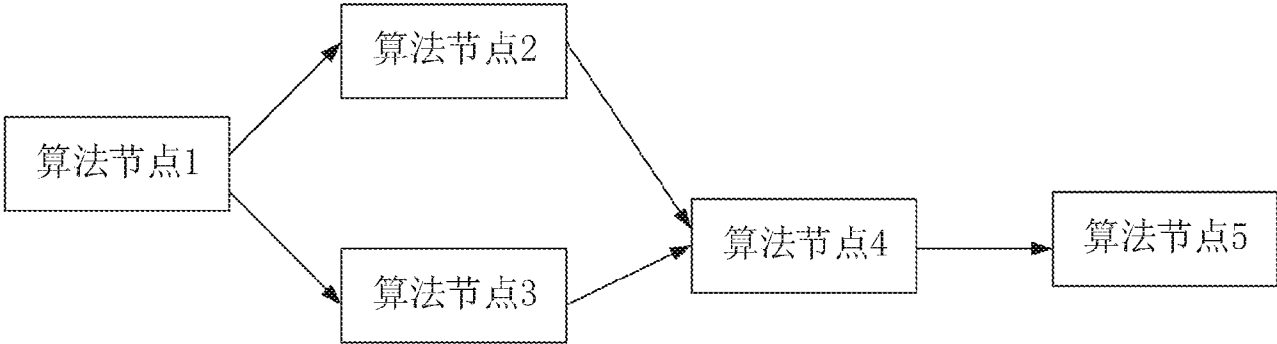


图 2

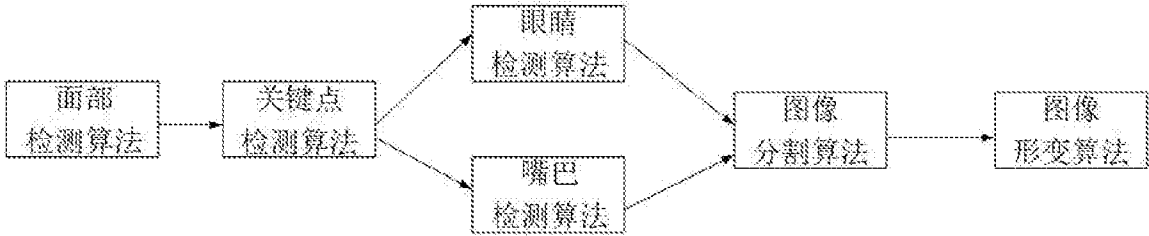


图 3

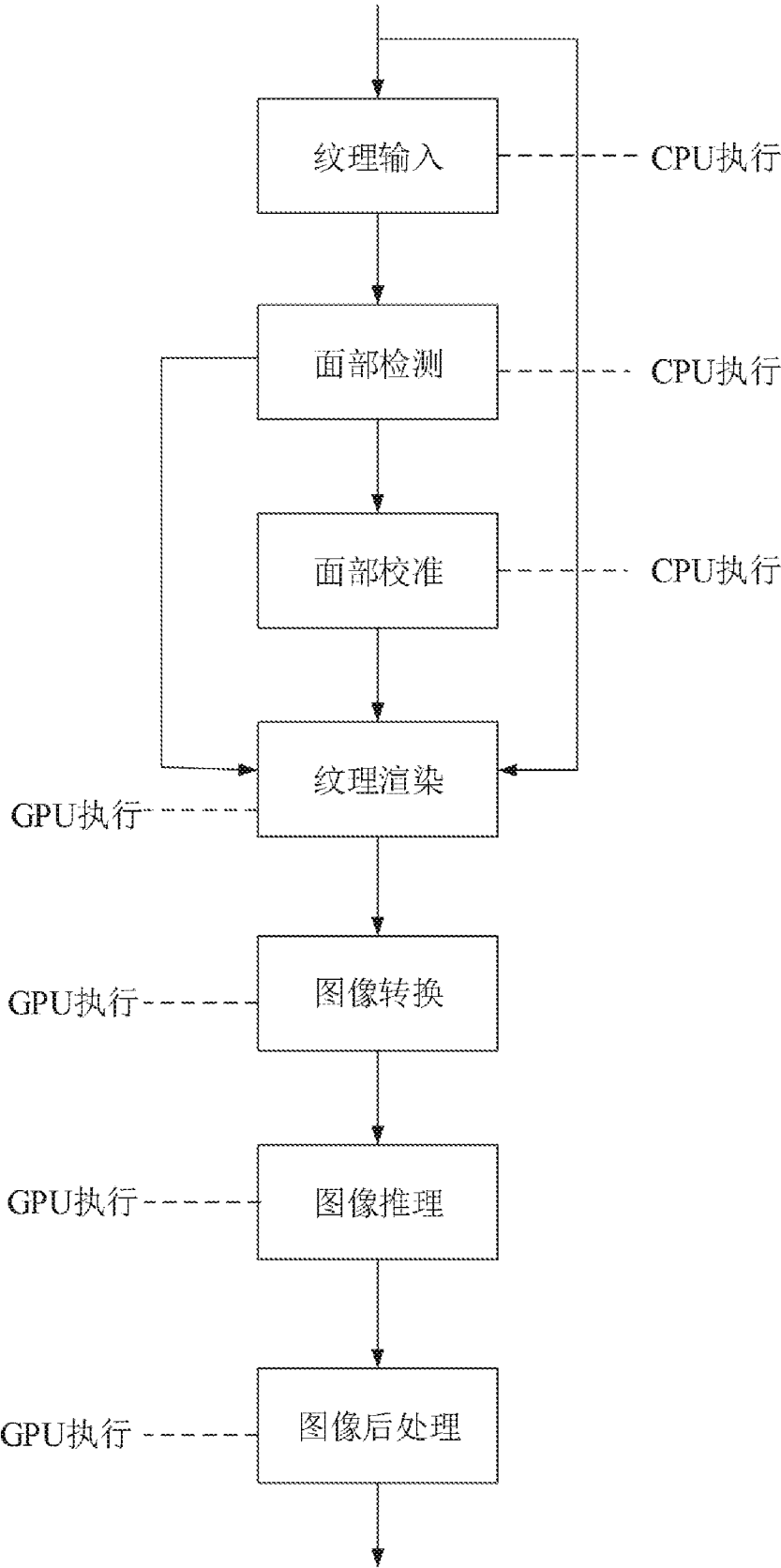


图 4

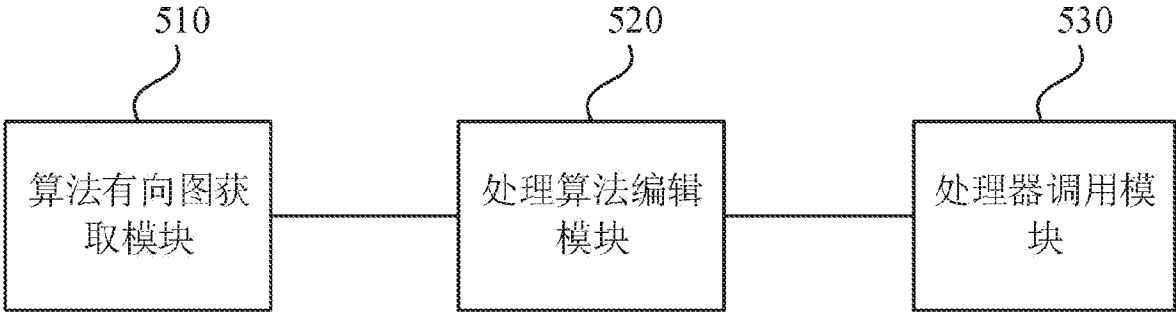


图 5

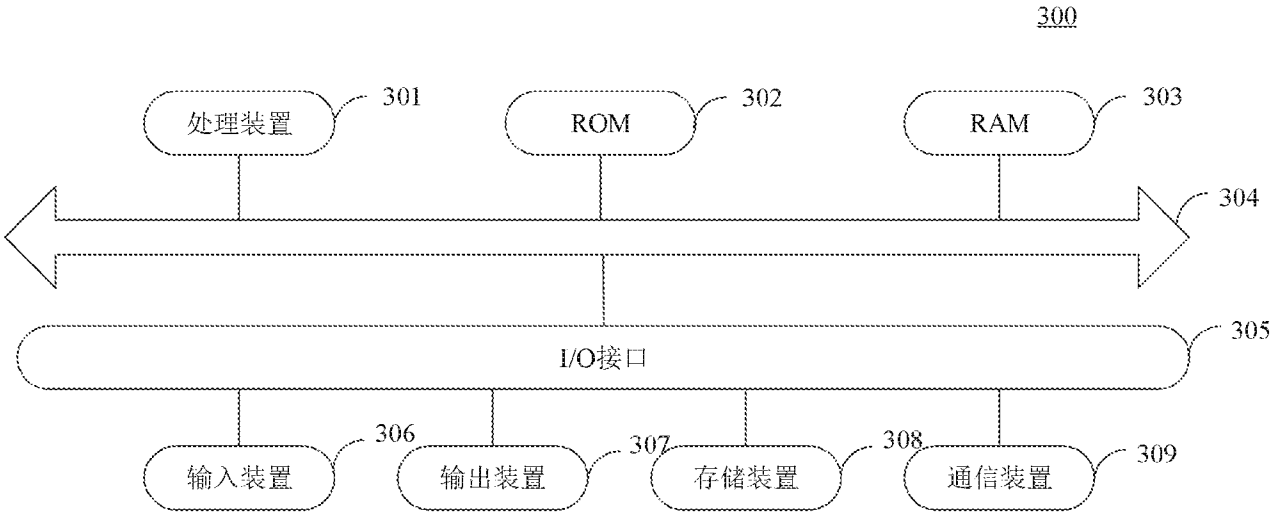


图 6