(19)国家知识产权局

(12)发明专利申请

(10)申请公布号 CN 117063461 A
(43)申请公布日 2023.11.14

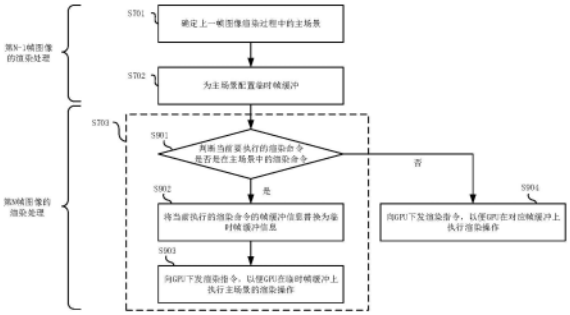
(21)申请号 202280024721.9
(22)申请日 2022.06.09
(66)本国优先权数据
202110650009.7 2021.06.10 CN
(85)PCT国际申请进入国家阶段日
2023.09.28
(86)PCT国际申请的申请数据
PCT/CN2022/097931 2022.06.09
(87)PCT国际申请的公布数据
WO2022/258024 ZH 2022.12.15
(71)申请人 荣耀终端有限公司
地址 518040 广东省深圳市福田区香蜜湖街道东海社区红荔西路8089号深业中城6号楼A单元3401
(72)发明人 陈聪儿 刘金晓
(74)专利代理机构 北京中博世达专利商标代理有限公司 11274
专利代理师 申健
(51)Int.Cl.
*H04M 1/72439*(2006.01)

(54)发明名称

一种图像处理方法和电子设备

(57)摘要

本申请实施例公开了一种图像处理方法和电子设备,涉及图像处理领域,可以使用较小的分辨率执行主场景中的大量渲染操作,从而达到降低渲染功耗的效果。具体方案为:确定对该第一图像执行渲染处理过程中的第一主场景。该第一主场景是该电子设备对该第一图像的渲染处理过程中,执行渲染操作数量最多的帧缓冲。配置临时帧缓冲,该临时帧缓冲的分辨率小于该第一主场景的分辨率。在对该第一图像进行渲染处理时,在该临时帧缓冲上,执行第一渲染操作。该第一渲染操作是该应用程序指示的在该第一主场景上执行的渲染操作。




CN 117063461 A

(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
2022 年 12 月 15 日 (15.12.2022)

WIPO | PCT

(10) 国际公布号
WO 2022/258024 A1

(51) 国际专利分类号:
*H04M 1/72439* (2021.01)

(21) 国际申请号: PCT/CN2022/097931

(22) 国际申请日: 2022 年 6 月 9 日 (09.06.2022)

(25) 申请语言: 中文

(26) 公布语言: 中文

(30) 优先权:
202110650009.7 2021年6月10日 (10.06.2021) CN

(71) 申请人: 荣耀终端有限公司(**HONOR DEVICE CO., LTD.**) [CN/CN]; 中国广东省深圳市福田区香蜜湖街道红荔西路8089号深业中城6号楼A单元3401, Guangdong 518040 (CN)。

(72) 发明人: 陈聪儿(**CHEN, Conger**); 中国广东省深圳市福田区香蜜湖街道红荔西路8089号深业中城6号楼A单元3401, Guangdong 518040 (CN)。 刘金晓(**LIU, Jinxiao**); 中国广东省深圳市福田区香蜜湖街道红荔西路8089号深业中城6号楼A单元3401, Guangdong 518040 (CN)。

(74) 代理人: 北京中博世达专利商标代理有限公司(**BEIJING ZBSD PATENT&TRADEMARK AGENT LTD.**); 中国北京市海淀区交大东路31号11号楼8层, Beijing 100044 (CN)。

(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, 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,

(54) **Title:** IMAGE PROCESSING METHOD AND ELECTRONIC DEVICE

(54) 发明名称: 一种图像处理方法和电子设备

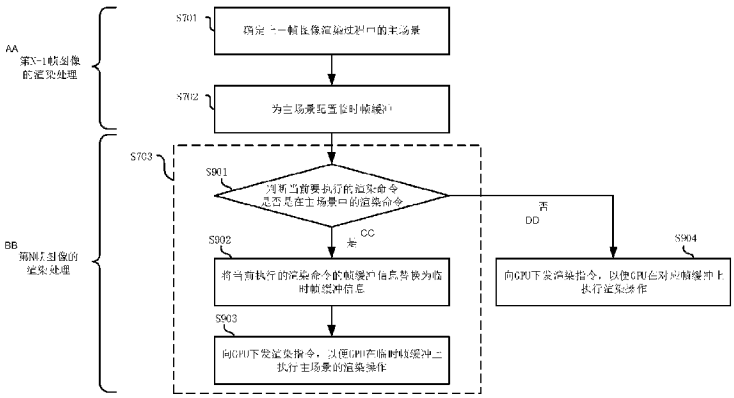



图 9

S701 Determine a primary scene in a process of performing rendering processing on a previous frame image
S702 Configure a temporary frame buffer for the primary scene
S901 Determine whether the current rendering command to be executed is a rendering command in the primary scene
S902 Replace the frame buffer information of the currently executed rendering command with temporary frame buffer information
S903 Issue a rendering instruction to the GPU, so that the GPU executes a rendering operation of the primary scene on the temporary frame buffer
S904 Issue a rendering instruction to the GPU so that the GPU executes a rendering operation on a corresponding frame buffer
AA Rendering processing of an N-1th frame image
BB Rendering processing of an Nth frame image
CC Yes
DD No

(57) **Abstract:** Embodiments of the present application disclose an image processing method and an electronic device, related to the field of image processing. A large number of rendering operations in a primary scene may be executed using a relatively small resolution, thereby achieving the effect of reducing power consumption during rendering. The specific solution is: determining a first primary scene in a process of executing rendering processing on a first image. The first primary scene is the frame buffer that executes the highest number of rendering operations in the process of rendering the first image by the electronic device. Configuring a temporary frame buffer, the resolution of the temporary frame buffer being less than the resolution of the first primary scene. When rendering processing is performed on the first image, a first rendering operation is executed on the temporary frame buffer. The first rendering operation is a rendering operation performed on the first primary scene indicated by the application.

WO 2022/258024 A1

[见续页]

LC, LK, LR, LS, LU, LY, MA, MD, ME, MG, MK, MN, 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, GH, GM, KE, LR, LS, MW, MZ, NA, RW, 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, 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**:

— 包括国际检索报告(条约第21条(3))。

— 含有关于援引加入遗漏项目或部分的信息(细则20.6)

**(57) 摘要**：本申请实施例公开了一种图像处理方法和电子设备，涉及图像处理领域，可以使用较小的分辨率执行主场景中的大量渲染操作，从而达到降低渲染功耗的效果。具体方案为：确定对该第一图像执行渲染处理过程中的第一主场景。该第一主场景是该电子设备对该第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。配置临时帧缓冲，该临时帧缓冲的分辨率小于该第一主场景的分辨率。在对该第一图像进行渲染处理时，在该临时帧缓冲上，执行第一渲染操作。该第一渲染操作是该应用程序指示的在该第一主场景上执行的渲染操作。

# 一种图像处理方法和电子设备

本申请要求于2021年6月10日提交国家知识产权局、申请号为202110650009.7、发明名称为“一种图像处理方法和电子设备”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。

## 技术领域

本申请实施例涉及图像处理领域，尤其涉及一种图像处理方法和电子设备。

## 背景技术

随着电子设备的发展，对电子设备显示图像的能力要求也越来越高。

示例性的，以电子设备为手机为例。为了能够为用户提供更加清楚的图像显示，手机可以采用更高的分辨率向用户展示图像。而手机在渲染获取具有高分辨率的图像时，产生的功耗更高。因此就会导致算力开销过大，发热严重等问题。严重时还会出现手机运行卡顿。进而严重影响用户体验。

## 发明内容

本申请实施例提供一种图像处理方法和电子设备，可以使用较小的分辨率执行主场景中的大量渲染操作，从而达到降低渲染功耗的效果。

为了达到上述目的，本申请实施例采用如下技术方案：

第一方面，提供一种图像处理方法，应用于电子设备对第一图像的渲染处理，该电子设备运行有应用程序，该电子设备对该第一图像执行渲染处理时调用一个或多个帧缓冲，该电子设备对该第一图像执行渲染处理的渲染操作由该应用程序下发。该方法包括：确定对该第一图像执行渲染处理过程中的第一主场景。该第一主场景是该电子设备对该第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。配置临时帧缓冲，该临时帧缓冲的分辨率小于该第一主场景的分辨率。在对该第一图像进行渲染处理时，在该临时帧缓冲上，执行第一渲染操作。该第一渲染操作是该应用程序指示的在该第一主场景上执行的渲染操作。

基于该方案，提供了一种降低主场景下的渲染功耗的方案。在本示例中，电子设备可以为主场景（如执行渲染处理最大的帧缓冲）配置具有较小分辨率的临时帧缓冲。电子设备还可以在临时帧缓冲上执行原先需要在主场景上执行的渲染操作。这样，就可以实现采用较小的分辨率执行渲染操作的目的，从而节省由于高分辨率的大量渲染操作导致的渲染功耗过高的问题。可以理解的是，本示例中，配置临时帧缓冲，以及在临时帧缓冲上执行渲染操作的对象，均为主场景，即渲染操作最多的帧缓冲。在本申请的另一些实现中，电子设备还可以为其他具有较多渲染操作的帧缓冲配置对应的具有较小分辨率的临时帧缓冲，并该对应的临时帧缓冲上进行较低分辨率的渲染操作，从而使得采用较小分辨率执行的渲染操作数量得到提升，进一步降低渲染操作的功耗。

在一种可能设计中，该确定对该第一图像的渲染处理过程中的第一主场景，包括：基于对第二图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲，确定该第一主场景。该第二图像的渲染处理在该第一图像的渲染处理之前。对该第二图像执行渲染



援引加入(细则20.6)

# 一种图像处理方法和电子设备

本申请要求于2021年6月10日提交国家知识产权局、申请号为202110650009.7、发明名称为“一种图像处理方法和电子设备”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。

## 技术领域

本申请实施例涉及图像处理领域，尤其涉及一种图像处理方法和电子设备。

## 背景技术

本申请实施例提供一种图像处理方法和电子设备，可以使用较小的分辨率执行主场景中的大量渲染操作，从而达到降低渲染功耗的效果。

为了达到上述目的，本申请实施例采用如下技术方案：

第一方面，提供一种图像处理方法，应用于电子设备对第一图像的渲染处理，该电子设备运行有应用程序，该电子设备对该第一图像执行渲染处理时调用一个或多个帧缓冲，该电子设备对该第一图像执行渲染处理的渲染操作由该应用程序下发。该方法包括：确定对该第一图像执行渲染处理过程中的第一主场景。该第一主场景是该电子设备对该第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。配置临时帧缓冲，该临时帧缓冲的分辨率小于该第一主场景的分辨率。在对该第一图像进行渲染处理时，在该临时帧缓冲上，执行第一渲染操作。该第一渲染操作是该应用程序指示的在该第一主场景上执行的渲染操作。

基于该方案，提供了一种降低主场景下的渲染功耗的方案。在本示例中，电子设备可以为主场景（如执行渲染处理最大的帧缓冲）配置具有较小分辨率的临时帧缓冲。电子设备还可以在临时帧缓冲上执行原先需要在主场景上执行的渲染操作。这样，就可以实现采用较小的分辨率执行渲染操作的目的，从而节省由于高分辨率的大量渲染操作导致的渲染功耗过高的问题。可以理解的是，本示例中，配置临时帧缓冲，以及在临时帧缓冲上执行渲染操作的对象，均为主场景，即渲染操作最多的帧缓冲。在本申请的另一些实现中，电子设备还可以为其他具有较多渲染操作的帧缓冲配置对应的具有较小分辨率的临时帧缓冲，并该对应的临时帧缓冲上进行较低分辨率的渲染操作，从而使得采用较小分辨率执行的渲染操作数量得到提升，进一步降低渲染操作的功耗。

在一种可能设计中，该确定对该第一图像的渲染处理过程中的第一主场景，包括：基于对第二图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲，确定该第一主场景。该第二图像的渲染处理在该第一图像的渲染处理之前。对该第二图像执行渲染处理时调用该第一主场景。基于该方案，提供了一种第一主场景的确定方案。在本示例中，电子设备可以根据在当前帧图像（如第一图像）的渲染处理之前，对其他帧图像（如第二图像）进行的渲染处理的过程中，各个帧缓冲上的渲染操作的数量确定当前帧图像的主场景。可以理解的是，第二图像的渲染处理在第一图像之前，那么在执行第一图像的渲染时，第二图像的所有渲染操作已经完成，因此电子设备就可以知晓第二图像在渲染过程中，各个帧缓冲上执行渲染操作的数量。由于图像的连续性，因此一般情况，可以认为对第一图像而言，渲染操作最多的帧缓冲与第二图像保持一致。

因此，在本示例中，可以将第二图像渲染处理过程中，执行渲染操作最多的帧缓冲，作为当前帧图像（即第一图像）的主场景。

在一种可能设计中，在该确定对该第一图像执行渲染处理过程中的第一主场景之前，该方法还包括：在对该第二图像的渲染处理过程中，确定在每个帧缓冲上执行绘制调用（drawcall）的数量，将执行 drawcall 的数量最多的帧缓冲确定为该第一主场景。基于该方案，提供了一种具体的确定第一主场景的方案示例。在本示例中，可以通过 drawcall 的数量，来表示渲染操作的数量。比如，drawcall 的数量越多，那么对应的渲染操作的数量就越多，反之亦然。由此，电子设备就可以通过确定第二图像的渲染过程中，各个帧缓冲上的 drawcall 的数量，确定具有最大 drawcall 数量的帧缓冲为第一主场景。可以理解的是，由于主场景上的 drawcall 最多，因此在主场景上的渲染处理最复杂，需要的功耗也就越高。因此，结合前述方案，在 drawcall 最多的帧缓冲（即主场景）上执行上述方案，能够达到较好的降低功耗的效果。

在一种可能设计中，该第二图像是该第一图像的上一帧图像。基于该方案，提供了一种第二图像的具体选择方案。比如，第二图像可以是第一图像的上一帧图像。这样，由于第二图像与第一图像所要显示的图像非常接近，因此根据第二图像渲染过程中的 drawcall 数量确定的第一主场景也就越准确。

在一种可能设计中，该电子设备配置有处理器和渲染处理模块。该在该临时帧缓冲上，执行第一渲染操作，包括：在该处理器接收来自该应用程序的对该第一图像的渲染命令的情况下，该处理器向该渲染处理模块下发第一渲染指令，该第一渲染指令包括该第一渲染操作，该第一渲染指令用于指示该渲染处理模块在该临时帧缓冲上执行该第一渲染操作。该渲染处理模块根据该第一渲染指令，在该临时帧缓冲上执行该第一渲染操作。基于该方案，提供了一种执行第一渲染操作的具体方案的示例。在本示例中，电子设备可以通过处理器和渲染处理模块配合，实现以较小的分辨率执行主场景的渲染操作的目的。比如，处理器可以向渲染处理模块下发渲染指令，指示渲染处理模块在具有较小分辨率的临时帧缓冲上执行本应在具有较大分辨率的主场景上执行的渲染操作。这样，渲染处理模块就可以根据渲染指令，在临时帧缓冲上执行对应的渲染操作，从而达到采用较小分辨率进行渲染的效果。

在一种可能设计中，在该处理器向该渲染处理模块下发第一渲染指令之前，该方法还包括：该处理器判断当前执行的渲染命令是对该主场景的渲染命令的情况下，将当前执行的该渲染命令绑定的帧缓冲信息由第一帧缓冲信息替换为第二帧缓冲信息，以得到该第一渲染指令，该第一帧缓冲信息用于指示在该主场景上执行该第一渲染操作，该第二帧缓冲信息用于指示在该临时帧缓冲上执行该第一渲染操作。基于该方案，提供了一种处理器向渲染处理模块发送的第一渲染指令的确定逻辑。在该示例中，处理器可以在确定当前来自应用程序的渲染命令是针对主场景的渲染命令的情况下，执行后续的操作。比如，处理器可以根据当前渲染命令所绑定的帧缓冲的帧缓冲对象，与主场景的帧缓冲对象进行对比，当结果一致时，则处理器确定当前的渲染命令是针对主场景的。此后，处理器可以向渲染处理模块下发渲染指令，以指示渲染处理模块进行渲染操作。在本示例中，处理器可以将渲染指令中，本指向主场景的帧缓冲信息，替换为指向临时帧缓冲的帧缓冲信息，并下发给渲染处理模块。这样，渲染处理模块

就可以根据该渲染指令，在该渲染指令指示的帧缓冲（如临时帧缓冲）上执行对应的渲染操作，由此实现在临时帧缓冲上执行本在主场景上执行的渲染操作。

在一种可能设计中，该第一帧缓冲信息包括第一帧缓冲对象，该第一帧缓冲对象是该主场景对应的帧缓冲对象，该第二帧缓冲信息包括第二帧缓冲对象，该第二帧缓冲对象是该临时帧缓冲对应的帧缓冲对象。基于该方案，提供了一种帧缓冲信息的具体实现。比如，帧缓冲信息可以包括帧缓冲对象。以替换 bindFrameBuffer（）函数的帧缓冲信息为例。在主场景为 FB1，临时帧缓冲为 FB3 的情况下，渲染命令中可以包括 bindFrameBuffer（1）。对应的，在不替换帧缓冲信息的情况下，处理器可以向渲染处理模块下发包括 bindFrameBuffer(1)的渲染指令，以指示渲染处理模块在 FB1 上执行渲染指令中的 drawcall。在采用本申请所述的方案时，可以将 FB1 的帧缓冲信息替换为 FB3 的帧缓冲信息，那么，渲染命令中可以包括 bindFrameBuffer（1）时，处理器可以将 bindFrameBuffer（1）替换为 bindFrameBuffer（3），并在渲染指令中携带该 bindFrameBuffer(3)，从而指示渲染处理模块在 FB3 上执行对应的渲染操作。

在一种可能设计中，该渲染命令是该应用程序下发给该处理器的，该渲染命令包括该第一渲染操作，以及该第一帧缓冲信息。基于该方案，提供了一种渲染命令的基本组成示意。在该示例中，渲染命令可以是来自应用程序的，用于指示处理器在所指示的帧缓冲上执行第一渲染操作。

在一种可能设计中，该处理器是中央处理器（CPU）。基于该方案，提供了一种处理器的具体实现。比如，该处理器可以是电子设备中的 CPU。在另一些实现中，处理器的功能还可以通过其他具有处理功能的部件或者电路实现。

在一种可能设计中，该渲染处理模块是图形处理器（GPU）。基于该方案，提供了一种渲染处理模块的具体实现。比如，该渲染处理模块可以是电子设备中的 GPU。在另一些实现中国，渲染处理模块的功能还可通过其他具有图形渲染功能的部件或者电路实现。

在一种可能设计中，该第一渲染指令还包括：对该第一渲染操作获取的图像执行多重采样。基于该方案，提供了一种在降低功耗的前提下提升图像显示质量的方案示例。在本示例中，电子设备可以对采用较低分辨率执行渲染操作获取的图像，采用多重采样技术，降低图像边缘的锯齿，提升图像的整体显示质量。

第二方面，提供一种图像处理装置，应用于电子设备对第一图像的渲染处理，该电子设备运行有应用程序，该电子设备对该第一图像执行渲染处理时调用一个或多个帧缓冲，该电子设备对该第一图像执行渲染处理的渲染操作由该应用程序下发。该装置包括：确定单元，用于确定对该第一图像执行渲染处理过程中的第一主场景。该第一主场景是该电子设备对该第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。配置单元，用于配置临时帧缓冲，该临时帧缓冲的分辨率小于该第一主场景的分辨率。执行单元，用于在对该第一图像进行渲染处理时，在该临时帧缓冲上，执行第一渲染操作。该第一渲染操作是该应用程序指示的在该第一主场景上执行的渲染操作。

在一种可能设计中，确定单元，具体用于基于对第二图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲，确定该第一主场景。该第二图像的渲染处理在该第一

图像的渲染处理之前。对该第二图像执行渲染处理时调用该第一主场景。

在一种可能设计中，确定单元，还用于在对该第二图像的渲染处理过程中，确定在每个帧缓冲上执行绘制调用（drawcall）的数量，将执行 drawcall 的数量最多的帧缓冲确定为该第一主场景。

在一种可能设计中，该第二图像是第一图像的上一帧图像。

在一种可能设计中，该执行单元的功能可以通过处理器和渲染处理模块实现。示例性的，在该处理器用于在接收到来自该应用程序的对该第一图像的渲染命令的情况下，向该渲染处理模块下发第一渲染指令，该第一渲染指令包括该第一渲染操作，该第一渲染指令用于指示该渲染处理模块在该临时帧缓冲上执行该第一渲染操作。该渲染处理模块用于根据该第一渲染指令，在该临时帧缓冲上执行该第一渲染操作。

在一种可能设计中，该处理器还用于，在该处理器向该渲染处理模块下发第一渲染指令之前，判断当前执行的渲染命令是对该主场景的渲染命令的情况下，将当前执行的该渲染命令绑定的帧缓冲信息由第一帧缓冲信息替换为第二帧缓冲信息，以得到该第一渲染指令，该第一帧缓冲信息用于指示在该主场景上执行该第一渲染操作，该第二帧缓冲信息用于指示在该临时帧缓冲上执行该第一渲染操作。

在一种可能设计中，该第一帧缓冲信息包括第一帧缓冲对象，该第一帧缓冲对象是该主场景对应的帧缓冲对象，该第二帧缓冲信息包括第二帧缓冲对象，该第二帧缓冲对象是该临时帧缓冲对应的帧缓冲对象。

在一种可能设计中，该渲染命令是该应用程序下发给该处理器的，该渲染命令包括该第一渲染操作，以及该第一帧缓冲信息。

在一种可能设计中，该处理器是中央处理器（CPU）。

在一种可能设计中，该渲染处理模块是图形处理器（GPU）。

在一种可能设计中，该第一渲染指令还包括：对该第一渲染操作获取的图像执行多重采样。

第三方面，提供一种电子设备，电子设备包括一个或多个处理器和一个或多个存储器；一个或多个存储器与一个或多个处理器耦合，一个或多个存储器存储有计算机指令；当一个或多个处理器执行计算机指令时，使得电子设备执行如上述第一方面以及各种可能的设计中任一种的图像处理方法。

第四方面，提供一种芯片系统，芯片系统包括接口电路和处理器；接口电路和处理器通过线路互联；接口电路用于从存储器接收信号，并向处理器发送信号，信号包括存储器中存储的计算机指令；当处理器执行计算机指令时，芯片系统执行如上述第一方面以及各种可能的设计中任一种的图像处理方法。

第五方面，提供一种计算机可读存储介质，计算机可读存储介质包括计算机指令，当计算机指令运行时，执行如上述第一方面以及各种可能的设计中任一种的图像处理方法。

第六方面，提供一种计算机程序产品，计算机程序产品中包括指令，当计算机程序产品在计算机上运行时，使得计算机可以根据指令执行如上述第一方面以及各种可能的设计中任一种的图像处理方法。

应当理解的是，上述第二方面，第三方面，第四方面，第五方面以及第六方面提

供的技术方案，其技术特征均可对应到第一方面及其可能的设计中提供的图像处理方法，因此能够达到的有益效果类似，此处不再赘述。

**附图说明**

图 1 为一种视频流的组成示意图；

图 2 为一种图像显示的示意图；

图 3 为又一种图像显示的示意图；

图 4 为本申请实施例提供的一种电子设备的组成示意图；

图 5 为本申请实施例提供的一种电子设备的软件组成示意图；

图 6A 为本申请实施例提供的一种渲染命令的下发示意图；

图 6B 为本申请实施例提供的一种渲染命令的下发示意图；

图 6C 为本申请实施例提供的一种图像渲染的示意图；

图 7A 为本申请实施例提供的一种图像处理方法的流程示意图；

图 7B 为本申请实施例提供的一种不同帧缓冲上 drawcall 数量的确定示意图；

图 7C 为本申请实施例提供的一种不同帧缓冲上 drawcall 数量的确定示意图；

图 8 为本申请实施例提供的一种渲染命令的下发示意图；

图 9 为本申请实施例提供的一种图像处理方法的流程示意图；

图 10 为本申请实施例提供的一种图像处理装置的组成示意图；

图 11 为本申请实施例提供的一种电子设备的组成示意图；

图 12 为本申请实施例提供的一种芯片系统的组成示意图。

**具体实施方式**

在用户使用电子设备的过程中，电子设备通过显示屏可以向用户展示多媒体流文件，以便于向用户提供丰富的视觉体验。

以多媒体流为视频流为例。该视频流可以包括多个帧图像。示例性的，结合图 1，该视频流可以包括 N 帧图像，如第 1 帧，第 2 帧，……第 N 帧。电子设备可以在展示该视频流时，分别在显示屏上显示第 1 帧，第 2 帧，……第 N 帧。在电子设备切换显示帧图像的频率高于人眼能够分辨的频率时，就可以实现用户对于不同帧图像的切换展示的不感知，从而获取观看连续的观看效果。

想要播放视频流的应用程序，可以向电子设备发出针对不同帧图像的渲染命令。电子设备可以根据这些渲染命令，进行对各个帧图像的渲染，并基于渲染结果进行显示。

在一些实现中，电子设备可以在显示当前帧图像时，根据应用程序下发的渲染命令，进行后续帧图像的渲染。以便于在需要显示后续帧图像时，可以将渲染完成的需要显示的帧图像替换当前显示的帧图像，展示给用户。

示例性的，以当前显示的帧图像为第 N-1 帧为例。如图 2 所示，电子设备可以在需要显示第 N-1 帧图像时，控制显示屏从当前帧缓冲（如图 2 所示的当前帧缓冲）中读取第 N-1 帧图像的数据进行显示。其中，当前帧缓冲可以是电子设备在内存中为当前需要显示的帧图像配置的存储空间，该当前帧缓冲可以用于存储当前需要显示的帧图像的数据（如各个像素的颜色数据，深度数据等）。也就是说，在需要显示第 N-1 帧图像时，电子设备可以根据当前帧缓冲中存储的第 N-1 帧图像的数据，控制显示屏

显示第 N-1 帧图像。

在一些实现中，在电子设备显示第 N-1 帧图像时，可以根据应用程序下发的渲染命令，对后续将要显示的图像（如第 N 帧）进行渲染，从而获取用于显示第 N 帧图像的数据。

请参考图 3，电子设备在显示第 N-1 帧图像时，可以执行对第 N 帧的渲染。可以理解的是，电子设备在渲染帧图像的过程中，需要将渲染的结果存储起来以便后续使用。因此，在本示例中，电子设备可以在对第 N 帧图像进行渲染之前，在内存中配置帧缓冲。帧缓冲可以对应到内存中的存储空间。在电子设备进行渲染时，可以将渲染结果存储在对应的帧缓冲中。

在一些实施例中，为了避免在一个帧缓冲上执行过多的操作，电子设备可以在执行 1 个帧图像的渲染时，调用多个帧缓冲。示例性的，电子设备可以在执行第 N 帧图像的渲染时，调用如图 3 所示的 3 个帧缓冲（如 FBO，FB1，以及 FB2）。这样，在执行对第 N 帧图像的渲染时，FB1 可以用于存储第 N 帧图像的一部分元素的渲染结果（如称为渲染结果 1），FB2 可以用于存储第 N 帧图像的另一部分元素的渲染结果（如称为渲染结果 2）。在分别获取渲染结果 1 和渲染结果 2 之后，电子设备可以将渲染结果 1 以及渲染结果 2 融合（或称为渲染）到 FBO 中。由此即可在 FBO 中获取第 N 帧图像的完整的渲染结果。

电子设备可以在需要显示第 N 帧图像时，控制 FBO 中的数据交换（swap）到当前帧缓冲中，进而根据交换之后的当前帧缓冲中的数据（如第 N 帧的数据），控制显示屏显示第 N 帧图像。

需要说明的是，为了能够对帧图像进行完整的渲染，电子设备可以在创建帧缓冲时，为帧缓冲配置对应的分辨率。分辨率越高，帧缓冲对应的存储空间就越大，这样就能够实现对较高分辨率的图像进行渲染。

可以理解的是，随着电子设备的发展，用户对于电子设备提供的显示分辨率要求越来越高。各个帧图像的渲染都需要进行较高分辨率的渲染，这也对电子设备的图像处理能力提出了较高的要求。由此也会引入电子设备的功耗和发热的显著增加，从而影响用户体验。

为了解决上述问题，本申请实施例提供一种图像处理方法，可以使得电子设备能够根据已经完成渲染的帧图像的渲染情况，灵活调整后续帧图像的渲染机制，比如调整后续帧图像渲染过程中使用的帧缓冲的分辨率，从而降低帧图像渲染过程中对内存和算力造成的较大压力，进而避免由此产生的电子设备的发热和功耗的上升。

以下结合附图对本申请实施例提供的方案进行详细说明。

需要说明的是，本申请实施例提供的图像处理方法，可以应用在用户的电子设备中。该电子设备可以是能够提供网络接入能够的设备。比如，该电子设备可以是手机、平板电脑、个人数字助理（personal digital assistant，PDA）、增强现实（augmented reality，AR）\虚拟现实（virtual reality，VR）设备、媒体播放器等具备拍摄功能的便携式移动设备，该电子设备也可以是智能手表等可穿戴电子设备。本申请实施例对该设备的具体形态不作特殊限制。在一些实施例中，该电子设备可以具有显示功能。比如，电子设备可以根据应用程序下发的渲染命令进行图像的渲染，并根据渲染获取

的渲染结果向用户展示对应的图像。

请参考图 4，为本申请实施例提供的一种电子设备 400 的组成示意图。如图 4 所示，该电子设备 400 可以包括处理器 410，外部存储器接口 420，内部存储器 421，通用串行总线（universal serial bus，USB）接口 430，充电管理模块 440，电源管理模块 441，电池 442，天线 1，天线 2，移动通信模块 450，无线通信模块 460，音频模块 470，传感器模块 480，按键 490，马达 491，指示器 492，摄像头 493，显示屏 494，以及用户标识模块（subscriber identification module，SIM）卡接口 495 等。其中，传感器模块 480 可以包括压力传感器，陀螺仪传感器，气压传感器，磁传感器，加速度传感器，距离传感器，接近光传感器，指纹传感器，温度传感器，触摸传感器，环境光传感器，骨传导传感器等。在一些实施例中，该电子设备 400 还可以包括扬声器，受话器，麦克风，耳机接口等器件用于实现电子设备的音频相关功能。

可以理解的是，本实施例示意的结构并不构成对电子设备 400 的具体限定。在另一些实施例中，电子设备 400 可以包括比图示更多或更少的部件，或者组合某些部件，或者拆分某些部件，或者不同的部件布置。图示的部件可以以硬件，软件或软件和硬件的组合实现。

处理器 410 可以包括一个或多个处理单元，例如：处理器 410 可以包括中央处理器（Central Processing Unit/Processor，CPU）应用处理器（application processor，AP），调制解调处理器，图形处理器（graphics processing unit，GPU），图像信号处理器（image signal processor，ISP），控制器，存储器，视频编解码器，数字信号处理器（digital signal processor，DSP），基带处理器，和/或神经网络处理器（neural-network processing unit，NPU）等。其中，不同的处理单元可以是独立的器件，也可以集成在一个或多个处理器中。

控制器可以是电子设备 400 的神经中枢和指挥中心。控制器可以根据指令操作码和时序信号，产生操作控制信号，完成取指令和执行指令的控制。

处理器 410 中还可以设置存储器，用于存储指令和数据。在一些实施例中，处理器 410 中的存储器为高速缓冲存储器。该存储器可以保存处理器 410 刚用过或循环使用的指令或数据。如果处理器 410 需要再次使用该指令或数据，可从所述存储器中直接调用。避免了重复存取，减少了处理器 410 的等待时间，因而提高了系统的效率。

在一些实施例中，处理器 410 可以包括一个或多个接口。接口可以包括集成电路（inter-integrated circuit，I2C）接口，集成电路内置音频（inter-integrated circuit sound，I2S）接口，脉冲编码调制（pulse code modulation，PCM）接口，通用异步收发传输器（universal asynchronous receiver/transmitter，UART）接口，移动产业处理器接口（mobile industry processor interface，MIPI），通用输入输出（general-purpose input/output，GPIO）接口，用户标识模块（subscriber identity module，SIM）接口，和/或通用串行总线（universal serial bus，USB）接口等。

电子设备 400 可以通过 ISP，摄像头 493，视频编解码器，GPU，显示屏 494 以及应用处理器 410 等实现拍摄功能。

ISP 用于处理摄像头 493 反馈的数据。例如，拍照时，打开快门，光线通过镜头被传递到摄像头 493 感光元件上，光信号转换为电信号，摄像头 493 感光元件将所述

电信号传递给 ISP 处理，转化为肉眼可见的图像。ISP 还可以对图像的噪点，亮度，肤色进行算法优化。ISP 还可以对拍摄场景的曝光，色温等参数优化。在一些实施例中，ISP 可以设置在摄像头 493 中。

摄像头 493 用于捕获静态图像或视频。物体通过镜头生成光学图像投射到感光元件。感光元件可以是电荷耦合器件（charge coupled device，CCD）或互补金属氧化物半导体（complementary metal-oxide-semiconductor，CMOS）光电晶体管。感光元件把光信号转换成电信号，之后将电信号传递给 ISP 转换成数字图像信号。ISP 将数字图像信号输出到 DSP 加工处理。DSP 将数字图像信号转换成标准的 RGB，YUV 等格式的图像信号。在一些实施例中，电子设备 400 可以包括 1 个或 N 个摄像头 493，N 为大于 1 的正整数。

数字信号处理器 410 用于处理数字信号，除了可以处理数字图像信号，还可以处理其他数字信号。例如，当电子设备 400 在频点选择时，数字信号处理器 410 用于对频点能量进行傅里叶变换等。

视频编解码器用于对数字视频压缩或解压缩。电子设备 400 可以支持一种或多种视频编解码器。这样，电子设备 400 可以播放或录制多种编码格式的视频，例如：动态图像专家组（moving picture experts group，MPEG）1，MPEG2，MPEG3，MPEG4 等。

NPU 为神经网络（neural-network，NN）计算处理器 410，通过借鉴生物神经网络结构，例如借鉴人脑神经元之间传递模式，对输入信息快速处理，还可以不断的自学习。通过 NPU 可以实现电子设备 400 的智能认知等应用，例如：图像识别，人脸识别，语音识别，文本理解等。

充电管理模块 440 用于从充电器接收充电输入。其中，充电器可以是无线充电器，也可以是有线充电器。在一些有线充电的实施例中，充电管理模块 440 可以通过 USB 接口 430 接收有线充电器的充电输入。在一些无线充电的实施例中，充电管理模块 440 可以通过电子设备 400 的无线充电线圈接收无线充电输入。充电管理模块 440 为电池 442 充电的同时，还可以通过电源管理模块 441 为电子设备 400 供电。电源管理模块 441 用于连接电池 442，充电管理模块 440 与处理器 410。电源管理模块 441 接收电池 442 和/或充电管理模块 440 的输入，为处理器 410，内部存储器 421，外部存储器，显示屏 494，摄像头 493，和无线通信模块 460 等供电。电源管理模块 441 还可以用于监测电池 442 容量，电池 442 循环次数，电池 442 健康状态（漏电，阻抗）等参数。在其他一些实施例中，电源管理模块 441 也可以设置于处理器 410 中。在另一些实施例中，电源管理模块 441 和充电管理模块 440 也可以设置于同一个器件中。

电子设备 400 的无线通信功能可以通过天线 1，天线 2，移动通信模块 450，无线通信模块 460，调制解调处理器以及基带处理器等实现。

天线 1 和天线 2 用于发射和接收电磁波信号。电子设备 400 中的每个天线可用于覆盖单个或多个通信频带。不同的天线还可以复用，以提高天线的利用率。例如：可以将天线 1 复用为无线局域网的分集天线。在另外一些实施例中，天线可以和调谐开关结合使用。

移动通信模块 450 可以提供应用在电子设备 400 上的包括 2G/3G/4G/5G 等无线通信的解决方案。移动通信模块 450 可以包括至少一个滤波器，开关，功率放大器，低

噪声放大器（low noise amplifier，LNA）等。移动通信模块 450 可以由天线 1 接收电磁波，并对接收的电磁波进行滤波，放大等处理，传送至调制解调处理器进行解调。移动通信模块 450 还可以对经调制解调处理器调制后的信号放大，经天线 1 转为电磁波辐射出去。在一些实施例中，移动通信模块 450 的至少部分功能模块可以被设置于处理器 410 中。在一些实施例中，移动通信模块 450 的至少部分功能模块可以与处理器 410 的至少部分模块被设置在同一个器件中。

调制解调处理器可以包括调制器和解调器。其中，调制器用于将待发送的低频基带信号调制成中高频信号。解调器用于将接收的电磁波信号解调为低频基带信号。随后解调器将解调得到的低频基带信号传送至基带处理器处理。低频基带信号经基带处理器处理后，被传递给应用处理器。应用处理器通过音频设备（不限于扬声器，受话器等）输出声音信号，或通过显示屏 494 显示图像或视频。在一些实施例中，调制解调处理器可以是独立的器件。在另一些实施例中，调制解调处理器可以独立于处理器 410，与移动通信模块 450 或其他功能模块设置在同一个器件中。

无线通信模块 460 可以提供应用在电子设备 400 上的包括无线局域网（wireless local area networks，WLAN）（如无线保真（wireless fidelity，Wi-Fi）网络），蓝牙（bluetooth，BT），全球导航卫星系统（global navigation satellite system，GNSS），调频（frequency modulation，FM），近距离无线通信技术（near field communication，NFC），红外技术（infrared，IR）等无线通信的解决方案。无线通信模块 460 可以是集成至少一个通信处理模块的一个或多个器件。无线通信模块 460 经由天线 2 接收电磁波，将电磁波信号调频以及滤波处理，将处理后的信号发送到处理器 410。无线通信模块 460 还可以从处理器 410 接收待发送的信号，对其进行调频，放大，经天线 2 转为电磁波辐射出去。

在一些实施例中，电子设备 400 的天线 1 和移动通信模块 450 耦合，天线 2 和无线通信模块 460 耦合，使得电子设备 400 可以通过无线通信技术与网络以及其他设备通信。所述无线通信技术可以包括全球移动通讯系统（global system for mobile communications，GSM），通用分组无线服务（general packet radio service，GPRS），码分多址接入（code division multiple access，CDMA），宽带码分多址（wideband code division multiple access，WCDMA），时分码分多址（time-division code division multiple access，TD-SCDMA），长期演进（long term evolution，LTE），BT，GNSS，WLAN，NFC，FM，和/或 IR 技术等。所述 GNSS 可以包括全球卫星定位系统（global positioning system，GPS），全球导航卫星系统（global navigation satellite system，GLONASS），北斗卫星导航系统（beidou navigation satellite system，BDS），准天顶卫星系统（quasi-zenith satellite system，QZSS）和/或星基增强系统（satellite based augmentation systems，SBAS）。

电子设备 400 通过 GPU，显示屏 494，以及应用处理器等实现显示功能。GPU 为图像处理的微处理器，连接显示屏 494 和应用处理器。GPU 用于执行数学和几何计算，用于图形渲染。处理器 410 可包括一个或多个 GPU，其执行程序指令以生成或改变显示信息。

显示屏 494 用于显示图像，视频等。显示屏 494 包括显示面板。显示面板可以采

用液晶显示屏 494（liquid crystal display，LCD），有机发光二极管（organic light-emitting diode，OLED），有源矩阵有机发光二极体或主动矩阵有机发光二极体（active-matrix organic light emitting diode，AMOLED），柔性发光二极管（flex light-emitting diode，FLED），Miniled，MicroLed，Micro-oLed，量子点发光二极管（quantum dot light emitting diodes，QLED）等。在一些实施例中，电子设备 400 可以包括 1 个或 N 个显示屏 494，N 为大于 1 的正整数。

外部存储器接口 420 可以用于连接外部存储卡，例如 Micro SD 卡，实现扩展电子设备 400 的存储能力。外部存储卡通过外部存储器接口 420 与处理器 410 通信，实现数据存储功能。例如将音乐，视频等文件保存在外部存储卡中。

内部存储器 421 可以用于存储计算机可执行程序代码，所述可执行程序代码包括指令。处理器 410 通过运行存储在内部存储器 421 的指令，从而执行电子设备 400 的各种功能应用以及数据处理。内部存储器 421 可以包括存储程序区和存储数据区。其中，存储程序区可存储操作系统，至少一个功能所需的应用程序（比如声音播放功能，图像播放功能等）等。存储数据区可存储电子设备 400 使用过程中所创建的数据（比如音频数据，电话本等）等。此外，内部存储器 421 可以包括高速随机存取存储器，还可以包括非易失性存储器，例如至少一个磁盘存储器件，闪存器件，通用闪存存储器（universal flash storage，UFS）等。

在本申请实施例中，内部存储器 421 也可以称为内存。在一些实施例中，处理器（如 CPU）可以在内存中为不同的帧图像的渲染处理创建对应的帧缓冲。比如，结合图 3，CPU 可以在应用程序的命令的控制下，在内存中创建 FBO，FB1 以及 FB2 用于对第 N 帧图像的渲染操作。

电子设备 400 可以通过音频模块 470，扬声器，受话器，麦克风，耳机接口，以及应用处理器 410 等实现音频功能。例如音乐播放，录音等。

音频模块 470 用于将数字音频信息转换成模拟音频信号输出，也用于将模拟音频输入转换为数字音频信号。音频模块 470 还可以用于对音频信号编码和解码。在一些实施例中，音频模块 470 可以设置于处理器 410 中，或将音频模块 470 的部分功能模块设置于处理器 410 中。扬声器，也称“喇叭”，用于将音频电信号转换为声音信号。电子设备 400 可以通过扬声器收听音乐，或收听免提通话。受话器，也称“听筒”，用于将音频电信号转换成声音信号。当电子设备 400 接听电话或语音信息时，可以通过将受话器靠近人耳接听语音。麦克风，也称“话筒”，“传声器”，用于将声音信号转换为电信号。当拨打电话或发送语音信息或需要通过语音助手触发电子设备 400 执行某些功能时，用户可以通过人嘴靠近麦克风发声，将声音信号输入到麦克风。电子设备 400 可以设置至少一个麦克风。在另一些实施例中，电子设备 400 可以设置两个麦克风，除了采集声音信号，还可以实现降噪功能。在另一些实施例中，电子设备 400 还可以设置三个，四个或更多麦克风，实现采集声音信号，降噪，还可以识别声音来源，实现定向录音功能等。耳机接口用于连接有线耳机。耳机接口可以是 USB 接口 430，也可以是 3.5mm 的开放移动电子设备 400 平台（open mobile terminal platform，OMTP）标准接口，美国蜂窝电信工业协会（cellular telecommunications industry association of the USA，CTIA）标准接口。

触摸传感器，也称“触控面板”。触摸传感器可以设置于显示屏 494，由触摸传感器与显示屏 494 组成触摸屏，也称“触控屏”。触摸传感器用于检测作用于其上或附近的触摸操作。触摸传感器可以将检测到的触摸操作传递给应用处理器 410，以确定触摸事件类型。在一些实施例中，可以通过显示屏 494 提供与触摸操作相关的视觉输出。在另一些实施例中，触摸传感器也可以设置于电子设备 400 的表面，与显示屏 494 所处的位置不同。

压力传感器用于感受压力信号，可以将压力信号转换成电信号。在一些实施例中，压力传感器可以设置于显示屏 494。压力传感器的种类很多，如电阻式压力传感器，电感式压力传感器，电容式压力传感器等。电容式压力传感器可以是包括至少两个具有导电材料的平行板。当有力作用于压力传感器，电极之间的电容改变。电子设备 400 根据电容的变化确定压力的强度。当有触摸操作作用于显示屏 494，电子设备 400 根据压力传感器检测所述触摸操作强度。电子设备 400 也可以根据压力传感器的检测信号计算触摸的位置。在一些实施例中，作用于相同触摸位置，但不同触摸操作强度的触摸操作，可以对应不同的操作指令。例如：当有触摸操作强度小于第一压力阈值的触摸操作作用于短消息应用图标时，执行查看短消息的指令。当有触摸操作强度大于或等于第一压力阈值的触摸操作作用于短消息应用图标时，执行新建短消息的指令。

陀螺仪传感器可以用于确定电子设备 400 的运动姿态。在一些实施例中，可以通过陀螺仪传感器确定电子设备 400 围绕三个轴（即，x，y 和 z 轴）的角速度。陀螺仪传感器可以用于拍摄防抖。示例性的，当按下快门，陀螺仪传感器检测电子设备 400 抖动的角度，根据角度计算出镜头模组需要补偿的距离，让镜头通过反向运动抵消电子设备 400 的抖动，实现防抖。陀螺仪传感器还可以用于导航，体感游戏场景。

气压传感器用于测量气压。在一些实施例中，电子设备 400 通过气压传感器测得的气压值计算海拔高度，辅助定位和导航。

磁传感器包括霍尔传感器。电子设备 400 可以利用磁传感器检测翻盖皮套的开合。在一些实施例中，当电子设备 400 是翻盖机时，电子设备 400 可以根据磁传感器检测翻盖的开合。进而根据检测到的皮套的开合状态或翻盖的开合状态，设置翻盖自动解锁等特性。

加速度传感器可检测电子设备 400 在各个方向上（一般为三轴）加速度的大小。当电子设备 400 静止时可检测出重力的大小及方向。还可以用于识别电子设备 400 姿态，应用于横竖屏切换，计步器等应用。

距离传感器，用于测量距离。电子设备 400 可以通过红外或激光测量距离。在一些实施例中，拍摄场景，电子设备 400 可以利用距离传感器测距以实现快速对焦。

接近光传感器可以包括例如发光二极管（LED）和光检测器，例如光电二极管。发光二极管可以是红外发光二极管。电子设备 400 通过发光二极管向外发射红外光。电子设备 400 使用光电二极管检测来自附近物体的红外反射光。当检测到充分的反射光时，可以确定电子设备 400 附近有物体。当检测到不充分的反射光时，电子设备 400 可以确定电子设备 400 附近没有物体。电子设备 400 可以利用接近光传感器检测用户手持电子设备 400 贴近耳朵通话，以便自动熄灭屏幕达到省电的目的。接近光传感器也可用于皮套模式，口袋模式自动解锁与锁屏。

环境光传感器用于感知环境光亮度。电子设备 400 可以根据感知的环境光亮度自适应调节显示屏 494 亮度。环境光传感器也可用于拍照时自动调节白平衡。环境光传感器还可以与接近光传感器配合，检测电子设备 400 是否在口袋里，以防误触。

指纹传感器用于采集指纹。电子设备 400 可以利用采集的指纹特性实现指纹解锁，访问应用锁，指纹拍照，指纹接听来电等。

温度传感器用于检测温度。在一些实施例中，电子设备 400 利用温度传感器检测的温度，执行温度处理策略。例如，当温度传感器上报的温度超过阈值，电子设备 400 执行降低位于温度传感器附近的处理器 410 的性能，以便降低功耗实施热保护。在另一些实施例中，当温度低于另一阈值时，电子设备 400 对电池 442 加热，以避免低温导致电子设备 400 异常关机。在其他一些实施例中，当温度低于又一阈值时，电子设备 400 对电池 442 的输出电压执行升压，以避免低温导致的异常关机。

骨传导传感器可以获取振动信号。在一些实施例中，骨传导传感器可以获取人体声部振动骨块的振动信号。骨传导传感器也可以接触人体脉搏，接收血压跳动信号。在一些实施例中，骨传导传感器也可以设置于耳机中，结合成骨传导耳机。音频模块 470 可以基于所述骨传导传感器获取的声部振动骨块的振动信号，解析出语音信号，实现语音功能。应用处理器 410 可以基于所述骨传导传感器获取的血压跳动信号解析心率信息，实现心率检测功能。

按键 490 包括开机键，音量键等。按键 490 可以是机械按键 490。也可以是触摸式按键 490。电子设备 400 可以接收按键 490 输入，产生与电子设备 400 的用户设置以及功能控制有关的键信号输入。

马达 491 可以产生振动提示。马达 491 可以用于来电振动提示，也可以用于触摸振动反馈。例如，作用于不同应用（例如拍照，音频播放等）的触摸操作，可以对应不同的振动反馈效果。作用于显示屏 494 不同区域的触摸操作，马达 491 也可对应不同的振动反馈效果。不同的应用场景（例如：时间提醒，接收信息，闹钟，游戏等）也可以对应不同的振动反馈效果。触摸振动反馈效果还可以支持自定义。

指示器 492 可以是指示灯，可以用于指示充电状态，电量变化，也可以用于指示消息，未接来电，通知等。

SIM 卡接口 495 用于连接 SIM 卡。SIM 卡可以通过插入 SIM 卡接口 495，或从 SIM 卡接口 495 拔出，实现和电子设备 400 的接触和分离。电子设备 400 可以支持 1 个或 N 个 SIM 卡接口 495,N 为大于 1 的正整数。SIM 卡接口 495 可以支持 Nano SIM 卡,Micro SIM 卡，SIM 卡等。同一个 SIM 卡接口 495 可以同时插入多张卡。所述多张卡的类型可以相同，也可以不同。SIM 卡接口 495 也可以兼容不同类型的 SIM 卡。SIM 卡接口 495 也可以兼容外部存储卡。电子设备 400 通过 SIM 卡和网络交互，实现通话以及数据通信等功能。在一些实施例中，电子设备 400 采用 eSIM，即：嵌入式 SIM 卡。eSIM 卡可以嵌在电子设备 400 中，不能和电子设备 400 分离。

应当理解的是，上述图 4 示出了电子设备中的一种硬件结构组成。本申请中，还可以从另一个角度对电子设备 400 进行划分。比如，参考图 5，示出了电子设备 400 的另一种逻辑组成。

示例性的，在如图 5 所示的示例中，电子设备 400 可以就有分层架构。在该示例

中，分层架构将软件分成若干个层，每一层都有清晰的角色和分工。层与层之间通过软件接口通信。在一些实施例中，以电子设备运行有安卓®（Android）操作系统为例。系统可以分为五层，从上至下分别为应用程序层，应用程序框架层，安卓运行时(Android runtime，ART)和原生 C/C++库，硬件抽象层(Hardware Abstract Layer，HAL)以及内核层。

其中，应用程序层可以包括一系列应用程序包。如图 5 所示，应用程序包可以包括相机，图库，日历，通话，地图，导航，WLAN，蓝牙，音乐，视频，短信息等应用程序。

在本申请的一些实施例中，应用程序层可以包括向用户提供多媒体流展示功能的应用程序。比如，应用程序层中可以包括各类游戏类应用（如和平精英®，王者荣耀®等）。又如，应用程序层中还可以包括各类视频类应用（如爱奇艺®等）。在这些应用程序运行时，可以下发出渲染命令，由此使得 CPU 可以根据该渲染命令，控制 GPU 进行对应的渲染，以获取各个帧图像的数据。

应用程序框架层为应用程序层的应用程序提供应用编程接口(application programming interface，API)和编程框架。应用程序框架层包括一些预先定义的函数。如图 5 所示，应用程序框架层可以包括窗口管理器，内容提供器，视图系统，资源管理器，通知管理器，活动管理器，输入管理器等。窗口管理器提供窗口管理服务（Window Manager Service，WMS），WMS 可以用于窗口管理、窗口动画管理、surface 管理以及作为输入系统的中转站。内容提供器用来存放和获取数据，并使这些数据可以被应用程序访问。该数据可以包括视频，图像，音频，拨打和接听的电话，浏览历史和书签，电话簿等。视图系统包括可视控件，例如显示文字的控件，显示图片的控件等。视图系统可用于构建应用程序。显示界面可以由一个或多个视图组成的。例如，包括短信通知图标的显示界面，可以包括显示文字的视图以及显示图片的视图。资源管理器为应用程序提供各种资源，比如本地化字符串，图标，图片，布局文件，视频文件等等。通知管理器使应用程序可以在状态栏中显示通知信息，可以用于传达告知类型的消息，可以短暂停留后自动消失，无需用户交互。比如通知管理器被用于告知下载完成，消息提醒等。通知管理器还可以是以图表或者滚动条文本形式出现在系统顶部状态栏的通知，例如后台运行的应用程序的通知，还可以是以对话窗口形式出现在屏幕上的通知。例如在状态栏提示文本信息，发出提示音，电子设备振动，指示灯闪烁等。活动管理器可以提供活动管理服务（Activity Manager Service，AMS），AMS 可以用于系统组件（例如活动、服务、内容提供者、广播接收器）的启动、切换、调度以及应用进程的管理和调度工作。输入管理器可以提供输入管理服务（Input Manager Service，IMS），IMS 可以用于管理系统的输入，例如触摸屏输入、按键输入、传感器输入等。IMS 从输入设备节点取出事件，通过和 WMS 的交互，将事件分配至合适的窗口。

安卓运行时包括核心库和安卓运行时。安卓运行时负责将源代码转换为机器码。安卓运行时主要包括采用提前（ahead or time，AOT）编译技术和及时（just in time，JIT）编译技术。核心库主要用于提供基本的 Java 类库的功能，例如基础数据结构、数学、输入输出（Input Output，IO）、工具、数据库、网络等库。核心库为用户进行安卓应用开发提供了 API。

原生C/C++库可以包括多个功能模块。例如：表面管理器(surface manager)，媒体框架(Media Framework)，标准C库（Standard C library，libc），嵌入式系统的开放图形库（OpenGL for Embedded Systems，OpenGL ES）、Vulkan、SQLite、Webkit等。

其中，表面管理器用于对显示子系统进行管理，并且为多个应用程序提供了2D和3D图层的融合。媒体框架支持多种常用的音频，视频格式回放和录制，以及静态图像文件等。媒体库可以支持多种音视频编码格式，例如：动态图象专家组4（Moving Pictures Experts Group MPEG4），H.264，动态影像专家压缩标准音频层面3（Moving Picture Experts Group Audio Layer3，MP3），高级音频编码（Advanced Audio Coding，AAC），自适应多码解码（Adaptive Multi-Rate，AMR），联合图像专家组（Joint Photographic Experts Group，JPEG，或称为JPG），便携式网络图形（Portable Network Graphics，PNG）等。OpenGL ES和/或Vulkan提供应用程序中2D图形和3D图形的绘制和操作。SQLite为电子设备400的应用程序提供轻量级关系型数据库。

硬件抽象层运行于用户空间（user space），对内核层驱动进行封装，向上层提供调用接口。内核层是硬件和软件之间的层。内核层至少包含显示驱动，摄像头驱动，音频驱动，传感器驱动。

以下结合图4和图5所示的组成，对图像渲染过程中，软件和硬件的处理机制进行示例性说明。其中以处理器410中包括CPU和GPU为例。在本申请实施例中，CPU可以用于接收来自应用程序的渲染命令，并根据该渲染命令，向GPU下发对应的渲染指令，以便于GPU根据渲染指令执行对应的渲染。作为一种示例，渲染命令中可以包括bindFrameBuffer（）函数，以及一个或多个glDrawElement。对应的，渲染指令中也可以包括bindFrameBuffer（）函数，以及一个或多个glDrawElement。其中，bindFrameBuffer（）函数可以用于指示当前绑定的帧缓冲。如，bindFrameBuffer（1）可以指示当前绑定的帧缓冲为FB1，即在FB1上执行后续的glDrawElement。为了便于说明，以下示例中，将渲染命令/渲染指令中，glDrawElement的集合称为渲染操作。

在应用程序层中的应用程序需要渲染某个图像（如第N帧图像）时，可以向下发起渲染命令。在该渲染命令中，可以包括渲染该第N帧图像所需要调用的帧缓冲，以及在各个帧缓冲中需要执行的渲染操作。在一些示例中，渲染命令可以通过bindFrameBuffer（）函数，实现对应渲染操作与帧缓冲的绑定。比如，第N帧图像需要使用FB0，FB1以及FB2作为帧缓冲为例。结合图6A，为FB1对应的渲染命令的一种示例。应用程序下发的渲染命令中，可以包括bindFrameBuffer（1）函数，由此实现当前渲染操作与FB1的绑定。在绑定FB1之后，可以通过glDrawElement指令，实现在该FB1上的渲染操作指示。在本示例中，1个glDrawElement指令可以对应1个drawcall。在不同实现中，在FB1上执行的glDrawElement指令可以为多个，则对应的在FB1上执行的drawcall就可以是多个。例如，结合图6A，在该示例中，渲染命令中可以包括在FB1上执行的drawcall可以为A个，如glDrawElement 1-glDrawElement A。

类似的，对于需要使用FB2的渲染操作，在渲染指令中，可以通过bindFrameBuffer（2）函数绑定FB2。此后可以通过不同的glDrawElement指令，实现在该FB2上的渲

染操作指示。

在接收到上述命令之后，CPU 可以根据 bindFrameBuffer（）函数所绑定的帧缓冲信息，以及对应的 glDrawElement 指令，向 GPU 下发对应的渲染指令。从而使得 GPU 可以进行对应的渲染操作，并将该渲染操作的结果存储在当前绑定的帧缓冲中。

示例性的，CPU 可以在接收到 bindFrameBuffer（1）的指令时，确定激活（active）的帧缓冲为 FB1。CPU 可以向 GPU 下发在 FB1 上进行对应 glDrawElement 的渲染指令。比如，结合图 6A，CPU 可以向 GPU 下发包括执行 glDrawElement 1-1 到 glDrawElement 1-A 的渲染指令，以便于 GPU 执行 glDrawElement 1-1 到 glDrawElement 1-A。在一些实施例中，CPU 可以向 GPU 下发 bindFrameBuffer（1），以便于 GPU 可以确定在 FB1 上执行上述渲染指令，并将结果保存在 FB1 中。

类似的，结合图 6B，CPU 可以在接收到 bindFrameBuffer（2）的命令时，确定激活（active）的帧缓冲为 FB2。CPU 可以向 GPU 下发在 FB2 上进行对应 glDrawElement 渲染指令，由此完成对 FB2 上相关数据的渲染。

如此往复，即可完成对第 N 帧图像对应的各个帧缓冲的渲染。

这样，电子设备可以将这些帧缓冲中的渲染结果渲染到 FBO 中，由此即可在 FBO 中获取第 N 帧图像对应的数据。比如，以第 N 帧图像的默认帧缓冲为 FBO，第 N 帧图像在进行渲染处理过程中还调用 FB1 和 FB2 为例。结合图 6A 以及图 6B 的示例，参考图 6C，在 GPU 完成 FB1 和 FB2 上的渲染处理之后，即可在 FB1 和 FB2 中分别存储对应的数据（如各个像素点的颜色数据和/或深度数据等）。这些数据（如 FB1 和 FB2 中的数据）可以分别对应第 N 帧图像的部分元素。那么，结合图 6C，电子设备可以将 FB1 和 FB2 中的数据渲染到 FBO 上。这样，在 FBO 上就可以获取第 N 帧图像所有的数据。在电子设备需要显示第 N 帧图像时，结合图 3 所示的方案，CPU 可以将第 N 帧图像对应的数据，通过 swapbuffer 的指令，从 FBO 中交换到当前帧缓冲中。进而使得显示屏可以根据当前帧缓冲中的第 N 帧图像的数据进行显示。

需要说明的是，上述示例中，是以在 FBO 上只进行基于 FB1 和 FB2 的数据的渲染结果进行渲染获取第 N 帧图像的全量数据为例进行说明的。比如，以第 N 帧图像是根据游戏应用下发的渲染指令进行渲染的图像。在该第 N 帧图像中可以包括人物，树木等元素，还可以包括特效等元素，还可以包括用户界面（User Interface，UI）控件等元素。那么在一些实施例中，CPU 可以根据游戏应用下发的渲染命令，控制 GPU 在 FB1 上执行人物，树木，UI 控件等元素的渲染，并将结果存储在 FB1。CPU 还可以根据游戏应用下发的渲染命令，控制 GPU 在 FB2 上执行特效等元素的渲染，并将结果存储在 FB2。接着就可以通过将 FB1 存储的数据和 FB2 存储的数据渲染到 FBO 上，即可获取第 N 帧图像的全量数据。

在本申请的另一些实施例中，在 FBO 上执行的渲染操作，还可以包括基于 FB1，FB2 的数据之外的渲染指令执行的渲染操作。比如，继续结合上述示例，CPU 可以根据游戏应用下发的渲染命令，控制 GPU 在 FB1 上执行人物，树木等元素的渲染，并将结果存储在 FB1。CPU 还可以根据游戏应用下发的渲染命令，控制 GPU 在 FB2 上执行特效等元素的渲染，并将结果存储在 FB2。接着，CPU 还可以根据游戏应用下发的渲染命令，控制 GPU 在 FBO 上执行 UI 控件的渲染，并结合 FB1 和 FB2 存储的数据，将所有渲染结

果进行渲染融合，从而在 FBO 上获取第 N 帧图像的全量数据。

可以理解的是，在对第 N 帧图像的渲染过程中，在不同的帧缓冲上执行的 drawcall 数量是不同。执行的 drawcall 数量越多，则表明在该帧缓冲上执行的渲染操作越多，那么在该帧缓冲的渲染操作过程中就会消耗更多的算力资源，也就会产生更高的功耗以及发热。

本申请实施例提供的方案，能够使得电子设备可以自动识别当前需要渲染的帧图像的渲染过程中，消耗资源较高的帧缓冲（如称为主场景）。通过采用较低的分辨率，执行该主场景中的渲染操作，达到降低在主场景中的渲染操作消耗的算力资源的效果。

示例性的，基于连续帧图像的相关性，电子设备可以通过第 N-1 帧图像的渲染过程，确定主场景。在一些实施例中，CPU 可以根据已经完成渲染的帧图像（如第 N-1 帧）在处理过程中，在各个帧缓冲上执行的 drawcall 个数，确定在执行该第 N-1 帧图像在渲染过程中，需要执行渲染操作最多的帧缓冲（即主场景）。也就是说，在对当前帧图像（如第 N 帧）的渲染时，该主场景的渲染操作的数量也可能是最多的。在该示例中，CPU 可以在对第 N 帧图像进行渲染处理时，为该主场景在内存中配置具有较小分辨率的临时帧缓冲。以便于在收到对主场景的渲染命令的情况下，CPU 可以控制 GPU 在该临时帧缓冲中，以较小的分辨率执行主场景对应的渲染操作。由此就可以使得在对第 N 帧图像的渲染过程中，大量的渲染操作可以以较小的分辨率执行，从而减少渲染负载，降低渲染所需功耗。

以下结合示例，对本申请实施例提供的方案进行详细说明。示例性的，请参考图 7A，为本申请实施例提供的一种图像处理方法的流程示意图。如图所示，该方法可以包括：

S701、确定上一帧图像渲染过程中的主场景。

其中，主场景可以是渲染过程中，执行渲染操作数量最多的帧缓冲。

可以理解的是，以电子设备播放视频流为例。组成视频流的连续帧图像的场景具有关联性。因此，电子设备的处理器（如 CPU）可以根据上一帧图像的主场景确定当前帧图像的主场景。

示例性的，以当前渲染的帧图像为第 N 帧图像为例。CPU 可以根据上一帧（如第 N-1 帧图像）图像渲染过程中，在不同帧缓冲上执行 drawcall 的个数，确定第 N-1 帧图像的主场景。在一些实施例中，该第 N 帧图像也可以称为第一图像，第 N 帧图像的主场景也可以称为第一主场景。对应的，第 N-1 帧图像可以称为第二图像，第二图像的主场景可以与第一图像的主场景相同（如都是第一主场景）。

作为一种可能的实现方式，以执行渲染帧图像的默认帧缓冲为 FBO，第 N-1 帧图像的渲染过程中调用了 FB1，FB2 为例。为了便于说明，以下将应用程序下发的与 FB1 对应的渲染命令称为渲染命令 1，将应用程序下发的与 FB2 对应的渲染命令称为渲染命令 2。那么，渲染命令 1 中就可以包括 glBindFrameBuffer(1)用于绑定 FB1。渲染命令 1 中还可以包括一个或多个需要在 FB1 上执行的 glDrawElement。比如，渲染命令 1 中可以包括 A 个 glDrawElement，即 A 个 drawcall。类似的，渲染命令 2 中就可以包括 glBindFrameBuffer(2)用于绑定 FB2。渲染命令 2 中还可以包括一个或多个需要在 FB2 上执行的 glDrawElement。比如，渲染命令 2 中可以包括 B 个 glDrawElement，

即 B 个 drawcall。

在接收到对 FB1 和 FB2 的渲染命令时，CPU 可以对对应的渲染命令中包括的 drawcall 数量进行计数，从而获取各个 FB 上执行的 drawcall 数量。

结合图 7B，以 FB1 为例。CPU 可以在根据应用程序下发的渲染命令，执行 glBindFrameBuffer(1)时，初始化计数器 1。比如在内存中为 FB1 配置对应的计数帧位。通过初始化该计数器 1，既可以将该帧位的值初始化为 0。后续每在 FB1 上执行 glDrawElement 时，计数器 1 计数加 1，如执行 count1++。比如，在执行 glDrawElement1-1 后，CPU 可以对计数器 1 执行 count1++，从而使得存储在 FB1 Drawcall 数量的帧位的数值由 0 变为 1。也就是说，此时在 FB1 上执行的 drawcall 数量为 1。以此类推。那么，CPU 就可以确定在进行第 N-1 帧图像的渲染过程中，FB1 上执行 drawcall 的数量为计数器 1 的当前计数（比如，该计数可以为 A）。

与 FB1 类似的，CPU 还可以在根据应用程序下发的渲染命令，调用 glBindFrameBuffer(2)时，初始化计数器 2。如初始化 count2=0。后续每在 FB2 上执行 glDrawElement 时，计数器 2 计数加 1，如执行 count2++。在完成 FB2 上图像的渲染处理之后，CPU 就可以确定在进行第 N-1 帧图像的渲染过程中，FB2 上执行 drawcall 的数量为计数器 2 的当前计数（比如，该计数可以为 B）。如参考图 7C。在完成 FB1 和 FB2 的渲染处理之后，内存中存储 FB1 Drawcall 数量的帧位的数值可以为 A，存储 FB2 Drawcall 数量的帧位可以为 B。

在本示例中，CPU 可以选取 A 和 B 中较大的计数对应的帧缓冲作为主场景。比如，在 A 大于 B 时，则 CPU 可以确定 FB1 上执行的 drawcall 个数更多，由此确定 FB1 为第 N-1 帧图像的主场景。相对的，在 A 小于 B 时，则 CPU 可以确定 FB2 上执行的 drawcall 个数更多，由此确定 FB2 为第 N-1 帧图像的主场景。

在本申请的一些实施例中，该 S701 的执行过程，可以是 CPU 在进行第 N-1 帧图像的渲染处理的过程中执行的。

S702、为主场景配置临时帧缓冲。

根据上述 S701 中的说明，主场景是执行 drawcall 数量最多的帧缓冲。因此，在需要降低渲染处理对电子设备造成的压力时，可以对主场景下的渲染处理进行调整，从而获取相较于调整其他帧缓冲中的渲染处理机制更加显著的效果。

在本申请实施例中，电子设备的 CPU 可以为主场景配置临时帧缓冲，该临时帧缓冲的分辨率可以小于主场景的分辨率。由此使得电子设备可以采用较小的分辨率，对主场景中的大量 drawcall 进行处理，从而降低高分辨率的渲染对电子设备造成的影响。

示例性的，在一些实施例中，CPU 可以在内存中为主场景配置临时帧缓冲之前，确定主场景的分辨率。

作为一种可能的实现，CPU 可以根据第 N-1 帧图像渲染时，使用的画布大小，确定该主场景的分辨率。可以理解的是，在应用程序向 CPU 下发渲染命令时，在渲染命令中可以通过 bindFrameBuffer()函数绑定帧缓冲。以渲染命令通过 bindFrameBuffer（1）绑定 FB1 为例。在该渲染命令中还可以通过 glViewPort（x，y）函数指定在 FB1 中执行渲染操作时的绘制大小。这样，CPU 就可以控制 GPU 在该 x*y 的像素区域中执行后续的 glDrawElement 对应的渲染处理。在本示例中，CPU 可以在获取主场景的分

辨率时，根据该主场景在第 N-1 帧图像渲染过程中，glViewPort（x，y）函数指定的像素区域，确定主场景的像素大小。比如，以 FB1 为主场景为例。CPU 可以确定在执行第 N-1 帧图像的渲染时，绑定 FB1（如接收到 bindFrameBuffer（1）函数）后，接收到 glViewPort（2218，978）。那么 CPU 就可以确定在第 N-1 帧图像的渲染过程中，使用了 FB1 上 2218×978 的像素区域。由此，CPU 就可以确定主场景（即 FB1）的分辨率为 2218×978。

作为又一种可能的实现，CPU 可以在接收到对第 N 帧图像的渲染命令后，通过对主场景执行 glGetIntegerv 指令，确定主场景的分辨率。例如，以主场景为 FB1，执行 glGetIntegerv(GL_VIEWPORT)指令获取分辨率为例。CPU 可以在接收到第 N 帧图像中，绑定 FB1（如接收到 bindFrameBuffer（1）函数）后，执行 glGetIntegerv(GL_VIEWPORT)指令。从而根据获取的结果（如 tex1（2118×978）），确定主场景（即 FB1）的分辨率为 2218×978。

在本申请实施例中，CPU 可以在确定主场景的分辨率的情况下，为主场景配置具有较小分辨率的临时帧缓冲。其中，该配置临时帧缓冲的操作，可以是 CPU 在执行第 N-1 帧图像的渲染过程中执行的。在另一些实现中，该配置临时帧缓冲的操作，还可以是 CPU 在执行完成第 N-1 帧图像的渲染，开始执行第 N 帧图像的渲染之前进行的。

需要说明的是，在本申请的一些实施例中，在电子设备中可以配置有缩放参数。该缩放参数可以用于确定临时帧缓冲的分辨率。例如，CPU 可以根据如下公式（1）确定临时帧缓冲的分辨率。

临时帧缓冲的分辨率=主场景的分辨率×缩放参数……公式（1）。

其中，缩放参数可以是小于 1 的正数。该缩放参数可以是用户配置的，也可以是电子设备中预置的，也可以是在需要时电子设备从云端获取的。

在确定临时帧缓冲的分辨率之后，CPU 就可以在内存中为该临时帧缓冲配置对应大小的存储空间。

在本申请的一些实施例中，CPU 可以在执行完成第 N-1 帧图像的渲染处理之后，确定主场景的分辨率。进而根据上述公式(1)即可确定临时帧缓冲的分辨率。基于此，CPU 就可以在内存中完成对临时帧缓冲的配置。

在一些实现中，CPU 可以通过创建临时帧缓冲对象，并将该临时帧缓冲对象绑定到内存中配置的存储空间上，实现临时帧缓冲的配置。其中，临时帧缓冲对象可以是该临时帧缓冲的名称。

作为一种示例，以下给出一种创建帧缓冲的典型指令流的示例：

```
unsigned int fbo; // 定义一个 FB 的 ID，变量名定义为 fbo
glGenFramebuffers(1, &fbo); // 创建一个 FB，ID 值为 fbo,驱动会为该 FB 分配 ID，比如 fbo 可以被赋值为 3
glBindFramebuffer(GL_FRAMEBUFFER, fbo); //绑定帧缓冲 3
unsigned int texture; //定义一个纹理 ID，变量名定义为 texture
glGenTextures(1, &texture); //创建一个纹理，ID 值为 texture,驱动会为该 FB 分配 texture，比如 texture 可以被赋值为 11
glBindTexture(GL_TEXTURE_2D, texture); // 绑定纹理 11,其中 GL_TEXTURE_2D
```

用于指示纹理目标为 2D 纹理

glTexImage2D(GL_TEXTURE_2D, 0, GL_RGB, 800, 600, 0, GL_RGB, GL_UNSIGNED_BYTE, NULL); // 给纹理分配一个 800 * 600 大小的内存空间；其中函数中的第 2 个因子（即 0）可以用于指示纹理级别，函数中的第 3 个因子（即 GL_RGB）可以用于指示目标纹理格式，函数中的第 7 个因子（即 GL_RGB）可以用于指示入参的纹理格式，函数中的第 8 个因子（即 GL_UNSIGNED_BYTE）可以用于指示入参纹理数据类型

glFramebufferTexture2D(GL_FRAMEBUFFER,GL_COLOR_ATTACHMENT0,GL_TEXTURE_2D,texture, 0); // 将创建出来的纹理 texture 附加到当前绑定的帧缓冲对象

可以理解的是，以临时帧缓冲为 FB3 为例。由于该 FB3 的分辨率小于 FB1 的分辨率，因此 FB3 在内存中占用的存储空间小于 FB1 在内存中占用的存储空间。对应的，由于在 FB3 上执行的渲染操作的分辨率更低，也就是说在 FB3 上执行相同渲染处理时，需要处理的像素数量小于 FB1，因此在 FB3 上执行渲染操作时相较于 FB1 上执行渲染操作所需功耗更低，发热更小。

这样，在完成第 N-1 帧图像的渲染处理之后，执行第 N 帧图像的渲染处理之前，CPU 就可以完成确定主场景，并为主场景配置对应的临时帧缓冲的动作。

S703、在临时帧缓冲上执行主场景的渲染操作。

在本申请实施例中，电子设备可以通过处理器，使用具有较小分辨率的帧缓冲进行主场景下的大量渲染处理的操作。

示例性的，CPU 可以在接收到来自应用程序的渲染命令之后，根据该渲染命令控制 GPU 在对应的帧缓冲上执行渲染操作。

比如，请参考图 8。以主场景为 FB1，对应的临时帧缓冲为 FB3，在执行第 N 帧图像的渲染的过程中，在 FB1 中包括的 drawcall 数量为 2（如 glDrawElement1，以及 glDrawElement2）为例。

CPU 接收到来自应用程序的渲染命令包括如下指令为例：

bindFrameBuffer（1）；

glDrawElement1;

glDrawElement2。

在本示例中，CPU 可以将第 N 帧图像的渲染命令中，用于绑定主场景的标识（如 FB1）替换为临时帧缓冲的标识（如 FB3）。在一些实施例中，该 FB1 的帧缓冲对象也可以称为第一帧缓冲信息，该 FB3 的帧缓冲对象也可以称为第二帧缓冲信息。示例性的，CPU 可以将 bindFrameBuffer（1）替换为 bindFrameBuffer（3）。那么，替换后的渲染命令可以为：

bindFrameBuffer（3）；

glDrawElement1;

glDrawElement2。

也就是说，CPU 可以将原先绑定主场景（如 FB1）的渲染操作替换为绑定临时帧缓冲的渲染操作。进而使得 CPU 可以将该渲染指令（如称为第一渲染指令）下发给 GPU，这样，GPU 就可以在 FB3 上执行 glDrawElement1 以及 glDrawElement2 的渲染操作，

并将结果存储在 FB3 上。在一些实施例中，在第 N 帧图像的渲染过程中，在主场景（或临时帧缓冲）上执行的渲染操作也可以称为第一渲染操作。

可以理解的是，由于 FB3 的分辨率小于 FB1，因此，在 FB3 上执行相同的 drawcall（如 glDrawElement1 以及 glDrawElement2），需要消耗的功耗功效，发热更少。

需要说明的是，在一些实施例中，电子设备可以通过 CPU 对所有主场景的渲染操作执行上述方案，如采用更小分辨率的临时帧缓冲执行主场景的渲染操作。在另一些实施例中，电子设备还可以在需要进行功耗控制时，指向如图 7A 所示的方案。比如，作为一种可能的实现，电子设备可以在负载大于预设阈值的情况下，确定需要进行功耗控制。由此即可触发如图 7A 所示方案降低图像处理过程中的功耗。作为另一种可能的实现，电子设备可以结合主场景的分辨率执行上述方案。比如，电子设备可以在确定主场景的分辨率大于预设分辨率，和/或在主场景中执行的 drawcall 数量大于预设的渲染阈值的情况下，执行 S702，即为主场景配置临时帧缓冲，进而在具有较小分辨率的临时帧缓冲中执行主场景的渲染操作。

作为一种实现方式，结合图 7A 所示的方案，图 9 示出了本申请实施例提供的又一种图像处理方法的流程示意图。在一些实施例中，该方案可以有电子设备中的 CPU 执行，实现动态调整渲染操作的主场景对应的分辨率，进而降低功耗开销的效果。

如图 9 所示，在如图 7A 所示方案的基础上，S703 具体的可以通过 S901-S903 实现。

S901、判断当前要执行的渲染命令是否是在主场景中的渲染命令。

示例性的，CPU 可以在根据接收到的渲染命令中，bindFrameBuffer（）所指示的需要绑定的帧缓冲对象与 S701 中确定的主场景的帧缓冲对象是否相同，确定当前渲染命令是否需要在主场景中执行。

例如，在当前执行的渲染命令为 bindFrameBuffer（1），在 S701 中确定的主场景为 FB1 的情况下，电子设备就可以确定当前执行的渲染命令需要在主场景上执行。

又如在当前执行的渲染命令为 bindFrameBuffer（2），在 S701 中确定的主场景为 FB1 的情况下，电子设备就可以确定当前执行的渲染命令不需要在主场景上执行。

在确定当前执行的渲染命令是在主场景中执行的情况下，电子设备的 CPU 可以继续执行以下 S902。

S902、将当前执行的渲染命令的帧缓冲信息替换为临时帧缓冲信息。

其中，帧缓冲信息可以包括当前执行的命令需要绑定的帧缓冲对象。

示例性的，结合前述对 S703 的说明，CPU 可以将 bindFrameBuffer（1）对应的 FB1，替换为临时帧缓冲的帧缓冲对象（如 FB3）。比如，替换之后的命令可以为：bindFrameBuffer（3）。

S903、向 GPU 下发渲染指令，以便 GPU 在临时帧缓冲上执行主场景的渲染操作。

在本示例中，GPU 可以根据接收到的渲染指令执行对应的渲染操作。比如，GPU 在接收到 bindFrameBuffer（3），以及后续的 drawcall 时，就可以在 FB3 上执行这些 drewcall 的渲染操作，并在 FB3 上存储渲染结果。

可以理解的是，由于 FB3 的分辨率小于 FB1，因此 GPU 可以使用更少的算力资源完成相同数量的 drawcall。而 FB1 中的 drawcall 可以是第 N 帧图像的渲染过程中，

drawcall 数量最多的。因此通过该方案（如图 7A 或图 9）能够显著降低第 N 帧图像的渲染过程对算力的消耗。

在一些实施例中，如图 9 所示，CPU 还可以在执行 S901 的判断之后，根据判断结果执行 S904。

示例性的，在 CPU 确定当前要执行的渲染命令并非在主场景中的渲染命令的情况下，可以执行 S904。其中，参考 S901 中的说明，CPU 可以在当前要执行的渲染命令需要绑定的帧缓冲的帧缓冲对象不同于主场景的帧缓冲对象的情况下，确定当前要执行的渲染命令并非在主场景中的渲染命令。

可以理解的是，由于当前要执行的渲染命令并非在主场景中进行，因此该渲染命令对算力的消耗并不是第 N 帧图像中的渲染操作过程中消耗最大的。那么 CPU 就可以直接控制 GPU 在该渲染命令所指定的帧缓冲上执行对应的 drawcall。如执行 S904：向 GPU 下发渲染指令，以便 GPU 在对应帧缓冲上执行渲染操作。

如图 9 所示，以 CPU 当前执行对第 N 帧图像的渲染为例。在一些实施例中，CPU 可以在第 N-1 帧图像的渲染处理完成之后，即完成 S701-S702 的处理。即确定的第 N 帧图像的主场景，并为该主场景配置了对应的临时帧缓存。比如，以下发渲染命令的应用程序为游戏应用为例。CPU 可以在游戏刚开始时，执行 S701-S702。比如，CPU 可以被配置为从第 6 帧图像开始执行如图 7A 或图 9 所示的方案，那么，CPU 就可以在游戏开始后，第 5 帧图像完成渲染处理后，根据第 5 帧图像的渲染情况，确定主场景，并为该主场景配置对应的临时帧缓存。从而使得 CPU 可以在第 6 帧图像的渲染处理过程中，使用该临时帧缓存执行主场景的 drawcall，由此达到显著的降低渲染过程中功耗开销的效果。

需要说明的是，上述示例中，是以应用程序下发的渲染命令中通过 bindFrameBuffer（）函数指向主场景为例，对主场景和临时帧缓冲的替换方法进行说明的。在本申请的另一些实现中，在应用程序下发的渲染命令中还包括其他函数指向主场景时，CPU 还可以通过类似的方案，替换对应函数中指向的帧缓冲对象，比如由主场景替换为临时帧缓冲，从而使得对应的渲染操作能够在临时帧缓冲上执行。

示例性的，CPU 可以对当前帧缓冲的纹理中包括的附件的标识进行替换，从而使得 GPU 可以使用正确的附件在临时帧缓冲进行对应的渲染处理。比如，以主场景为 FB1，临时帧缓冲为 FB3 为例。FB1 的附件（如颜色附件，深度附件等）的标识可以是 11。那么，在将 FB1 的标识替换为 FB3 之后，CPU 可以将 FB1 的附件的标识也进行对应的替换。比如，将附件 11 替换为附件 13（即 FB3 的附件标识），例如，将 glBindTexture(GL_TEXTURE_2D, 11)替换为 glBindTexture(GL_TEXTURE_2D, 13)。这样，在 FB3 上执行渲染处理时，就可以根据附件 13 上的信息，进行正确的渲染处理。

通过上述方案说明，即可实现降低主场景渲染分辨率的效果。

在本申请实施例的另一些实现中，电子设备可以在临时帧缓冲上执行主场景的渲染操作时，采用多重采样的技术，提升降低分辨率渲染获取的渲染结果的显示效果，进一步提升向用户展示的图像的质量。

示例性的，CPU 可以在向 GPU 下发在临时帧缓冲中执行渲染操作的渲染指令时，指示 GPU 使用更多的颜色，深度和/或模板信息对图元（如点，直线，多边形等元素）

进行处理，从而达到虚化图像边缘的效果。这样，在该临时帧缓冲中进行渲染操作后获取的结果在显示时，其图像边缘就不会由于分辨率较低产生锯齿的观感。由此即可实现提升图像质量的目的。

上述主要从电子设备的角度对本申请实施例提供的方案进行了介绍。为了实现上述功能，其包含了执行各个功能相应的硬件结构和/或软件模块。本领域技术人员应该很容易意识到，结合本文中所公开的实施例描述的各示例的单元及算法步骤，本申请能够以硬件或硬件和计算机软件的结合形式来实现。某个功能究竟以硬件还是计算机软件驱动硬件的方式来执行，取决于技术方案的特定应用和设计约束条件。专业技术人员可以对每个特定的应用来使用不同方法来实现所描述的功能，但是这种实现不应认为超出本申请的范围。

本申请实施例可以根据上述方法示例对其中涉及的设备进行功能模块的划分，例如，可以对应各个功能划分各个功能模块，也可以将两个或两个以上的功能集成在一个处理模块中。上述集成的模块既可以采用硬件的形式实现，也可以采用软件功能模块的形式实现。需要说明的是，本申请实施例中对模块的划分是示意性的，仅仅为一种逻辑功能划分，实际实现时可以有另外的划分方式。

请参考图 10，为本申请实施例提供的一种图像处理装置 1000 的组成示意图。该装置可以设置在电子设备中，用于实现本申请实施例提供的任一种可能的图像处理方法。

示例性的，该装置可以应用于电子设备对第一图像的渲染处理，该电子设备运行有应用程序，该电子设备对该第一图像执行渲染处理时调用一个或多个帧缓冲，该电子设备对该第一图像执行渲染处理的渲染操作由该应用程序下发。

如图 10 所示，该装置包括：确定单元 1001，用于确定对该第一图像执行渲染处理过程中的第一主场景。该第一主场景是该电子设备对该第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。配置单元 1002，用于配置临时帧缓冲，该临时帧缓冲的分辨率小于该第一主场景的分辨率。执行单元 1003，用于在对该第一图像进行渲染处理时，在该临时帧缓冲上，执行第一渲染操作。该第一渲染操作是该应用程序指示的在该第一主场景上执行的渲染操作。

在一种可能设计中，确定单元 1001，具体用于基于对第二图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲，确定该第一主场景。该第二图像的渲染处理在该第一图像的渲染处理之前。对该第二图像执行渲染处理时调用该第一主场景。

在一种可能设计中，确定单元 1001，还用于在对该第二图像的渲染处理过程中，确定在每个帧缓冲上执行绘制调用（drawcall）的数量，将执行 drawcall 的数量最多的帧缓冲确定为该第一主场景。

在一种可能设计中，该第二图像是第一图像的上一帧图像。

需要说明的是，上述方法实施例涉及的各步骤的所有相关内容均可以援引到对应功能模块的功能描述，在此不再赘述。

在一种可能的实现中，如图 10 所示的各个单元的功能，可以通过如图 4 所示的硬件模块实现。比如，以如图 10 所示的执行单元 1003 为例。该执行单元 1003 的功能可以通过如图 4 所示的处理器以及渲染处理模块实现。其中，渲染处理模块可以是具有

图形渲染功能的模块。在一些实施例中，该渲染处理模块可以是如图 4 所示的 GPU。在一些实施例中，上述处理器可以是如图 4 所示的 CPU。

作为一种示例，对处理器配合渲染处理模块实现执行单元 1003 的功能进行说明。

示例性的，在接收到来自该应用程序的对该第一图像的渲染命令的情况下，该处理器用于向该渲染处理模块下发第一渲染指令，该第一渲染指令包括该第一渲染操作，该第一渲染指令用于指示该渲染处理模块在该临时帧缓冲上执行该第一渲染操作。该渲染处理模块用于根据该第一渲染指令，在该临时帧缓冲上执行该第一渲染操作。

在一种可能设计中，该处理器还用于在该处理器向该渲染处理模块下发第一渲染指令之前，判断当前执行的渲染命令是对该主场景的渲染命令的情况下，将当前执行的该渲染命令绑定的帧缓冲信息由第一帧缓冲信息替换为第二帧缓冲信息，以得到该第一渲染指令，该第一帧缓冲信息用于指示在该主场景上执行该第一渲染操作，该第二帧缓冲信息用于指示在该临时帧缓冲上执行该第一渲染操作。

在一种可能设计中，该第一帧缓冲信息包括第一帧缓冲对象，该第一帧缓冲对象是该主场景对应的帧缓冲对象，该第二帧缓冲信息包括第二帧缓冲对象，该第二帧缓冲对象是该临时帧缓冲对应的帧缓冲对象。

在一种可能设计中，该渲染命令是该应用程序下发给该处理器的，该渲染命令包括该第一渲染操作，以及该第一帧缓冲信息。

在一种可能设计中，该第一渲染指令还包括：对该第一渲染操作获取的图像执行多重采样。

图 11 示出了的一种电子设备 1100 的组成示意图。如图 11 所示，该电子设备 1100 可以包括：处理器 1101 和存储器 1102。该存储器 1102 用于存储计算机执行指令。示例性的，在一些实施例中，当该处理器 1101 执行该存储器 1102 存储的指令时，可以使得该电子设备 1100 执行上述实施例中任一种所示的图像处理方法。

需要说明的是，上述方法实施例涉及的各步骤的所有相关内容均可以援引到对应功能模块的功能描述，在此不再赘述。

图 12 示出了的一种芯片系统 1200 的组成示意图。该芯片系统 1200 可以包括：处理器 1201 和通信接口 1202，用于支持相关设备实现上述实施例中所涉及的功能。在一种可能的设计中，芯片系统还包括存储器，用于保存终端必要的程序指令和数据。该芯片系统，可以由芯片构成，也可以包含芯片和其他分立器件。需要说明的是，在本申请的一些实现方式中，该通信接口 1202 也可称为接口电路。

需要说明的是，上述方法实施例涉及的各步骤的所有相关内容均可以援引到对应功能模块的功能描述，在此不再赘述。

在上述实施例中的功能或动作或操作或步骤等，可以全部或部分地通过软件、硬件、固件或者其任意组合来实现。当使用软件程序实现时，可以全部或部分地以计算机程序产品的形式来实现。该计算机程序产品包括一个或多个计算机指令。在计算机上加载和执行计算机程序指令时，全部或部分地产生按照本申请实施例所述的流程或功能。所述计算机可以是通用计算机、专用计算机、计算机网络、或者其他可编程装置。所述计算机指令可以存储在计算机可读存储介质中，或者从一个计算机可读存储介质向另一个计算机可读存储介质传输，例如，所述计算机指令可以从一个网站站点、

计算机、服务器或者数据中心通过有线（例如同轴电缆、光纤、数字用户线（digital subscriber line，DSL））或无线（例如红外、无线、微波等）方式向另一个网站站点、计算机、服务器或数据中心进行传输。所述计算机可读存储介质可以是计算机能够存取的任何可用介质或者是包括一个或多个可以用介质集成的服务器、数据中心等数据存储设备。所述可用介质可以是磁性介质（例如，软盘、硬盘、磁带），光介质（例如，DVD）、或者半导体介质（例如固态硬盘（solid state disk，SSD））等。

尽管结合具体特征及其实施例对本申请进行了描述，显而易见的，在不脱离本申请的精神和范围的情况下，可对其进行各种修改和组合。相应地，本说明书和附图仅仅是所附权利要求所界定的本申请的示例性说明，且视为已覆盖本申请范围内的任意和所有修改、变化、组合或等同物。显然，本领域的技术人员可以对本申请进行各种改动和变型而不脱离本申请的精神和范围。这样，倘若本申请的这些修改和变型属于本申请权利要求及其等同技术的范围之内，则本申请也意图包括这些改动和变型在内。

# 权 利 要 求 书

1.一种图像处理方法，其特征在于，应用于电子设备，所述电子设备运行有应用程序，所述电子设备执行所述应用程序下发的渲染操作，所述方法包括：

在第一帧缓冲和第二帧缓冲上执行第一图像的渲染操作，所述第一帧缓冲和所述第二帧缓冲不同；

在第三帧缓冲和所述第二帧缓冲上执行第二图像的渲染操作，所述第一图像和所述第二图像为连续的两帧图像，不在所述第三帧缓冲上执行所述第一图像的渲染操作，不在所述第一帧缓冲上执行所述第二图像的渲染操作，所述第三帧缓冲的分辨率小于所述第一帧缓冲的分辨率，所述第三帧缓冲和所述第二帧缓冲不同，所述第三帧缓冲和所述第一帧缓冲不同。

2.根据权利要求1所述的方法，其特征在于，绘制所述第一图像所使用的帧缓冲和绘制所述第二图像所使用的帧缓冲数量相同。

3.根据权利要求1或2所述的方法，其特征在于，所述第一帧缓冲和所述第三帧缓冲上执行的所述第一图像的渲染操作数量相同。

4.根据权利要求1-3中任一项所述的方法，其特征在于，所述第一帧缓冲和所述第三帧缓冲上执行的所述第一图像的渲染操作相同。

5.根据权利要求1-4中任一项所述的方法，其特征在于，所述方法还包括：创建所述第三帧缓冲。

6.根据权利要求1-4中任一项所述的方法，其特征在于，所述第一帧缓冲是所述电子设备对所述第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。

7.根据权利要求5所述的方法，其特征在于，所述方法还包括：确定所述电子设备对所述第一图像的渲染处理过程中，执行渲染操作数量最多的帧缓冲。

8.根据权利要求6所述的方法，其特征在于，所述方法还包括：

在对所述第一图像的渲染处理过程中，确定在每个帧缓冲上执行绘制调用drawcall的数量，

将执行drawcall的数量最多的帧缓冲确定为所述第一帧缓冲。

9.根据权利要求1-7中任一项所述的方法，其特征在于，所述渲染操作为glDrawElement。

10.根据权利要求1所述的方法，其特征在于，所述第二图像的渲染操作在所述第一图像的渲染操作之后，所述第一帧缓冲用于执行所述第一图像的主场景的渲染，所述第三帧缓冲用于执行所述第二图像的主场景的渲染。

11.根据权利要求1-10中任一项所述的方法，其特征在于，所述电子设备配置有处理器和渲染处理模块；

所述在第一帧缓冲和第二帧缓冲上执行第一图像的渲染操作，包括：

在所述处理器接收来自所述应用程序的对所述第一图像的渲染操作的情况下，所述处理器向所述渲染处理模块下发第一渲染指令流，所述第一渲染指令流用于指示所述渲染处理模块在所述第一帧缓冲和所述第二帧缓冲上执行所述第一图像的渲染操作；

所述渲染处理模块根据所述第一渲染指令流，在所述第一帧缓冲和所述第二帧缓冲上执行所述第一图像的渲染操作。

12. 根据权利要求 11 所述的方法，其特征在于，

所述在第三帧缓冲和所述第二帧缓冲上执行第二图像的渲染操作，包括：

在所述处理器接收来自所述应用程序的对所述第二图像的渲染操作的情况下，所述处理器向所述渲染处理模块下发第二渲染指令流；所述应用程序的对所述第二图像的渲染操作指示在所述第一帧缓冲和所述第二帧缓冲上执行所述第二图像的渲染操作；所述第二渲染指令流用于指示所述渲染处理模块在所述第三帧缓冲和所述第二帧缓冲上执行所述第二图像的渲染操作；

所述渲染处理模块根据所述第二渲染指令流，在所述第三帧缓冲和所述第二帧缓冲上执行所述第二图像的渲染操作。

13. 根据权利要求 12 所述的方法，其特征在于，

在所述处理器向所述渲染处理模块下发第二渲染指令流之前，所述方法还包括：

所述处理器根据所述第一图像的渲染操作，确定执行渲染操作最多的帧缓冲为所述第一帧缓冲；

所述处理器创建所述第三帧缓冲，所述第三帧缓冲用于执行所述第一帧缓冲上的渲染操作。

14. 根据权利要求 13 所述的方法，其特征在于，在所述处理器创建所述第三帧缓冲之后，所述方法还包括：

所述处理器将所述应用程序的对所述第二图像的渲染操作中，在所述第一帧缓冲上执行的渲染操作所指向的帧缓冲替换为所述第三帧缓冲，获得所述第三渲染指令流。

15. 根据权利要求 11-14 中任一项所述的方法，其特征在于，所述处理器是中央处理器 CPU。

16. 根据权利要求 11-15 中任一项所述的方法，其特征在于，所述渲染处理模块是图形处理器 GPU。

17. 根据权利要求 11-16 中任一项所述的方法，其特征在于，所述第三渲染指令流还包括：对所述第二图像执行多重采样。

18. 一种电子设备，其特征在于，所述电子设备包括一个或多个处理器和一个或多个存储器；所述一个或多个存储器与所述一个或多个处理器耦合，所述一个或多个存储器存储有计算机指令；

当所述一个或多个处理器执行所述计算机指令时，使得所述电子设备执行如权利要求 1-17 中任一项所述的图像处理方法。

19. 一种计算机可读存储介质，其特征在于，包括计算机指令，当所述计算机指令在电子设备上运行时，使得所述电子设备执行如权利要求 1-17 中任一项所述的图像处理方法。

20. 一种芯片系统，其特征在于，所述芯片系统包括接口电路和处理器；所述接口电路和所述处理器通过线路互联；所述接口电路用于从存储器接收信号，并向所述处理器发送信号，所述信号包括所述存储器中存储的计算机指令；当所述处理器执行所述计算机指令时，所述芯片系统执行如权利要求 1-17 中任一项所述的图像处理方法。

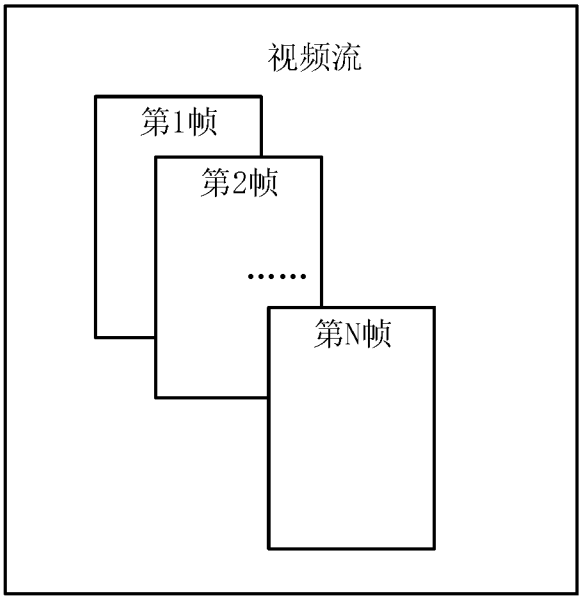



图 1

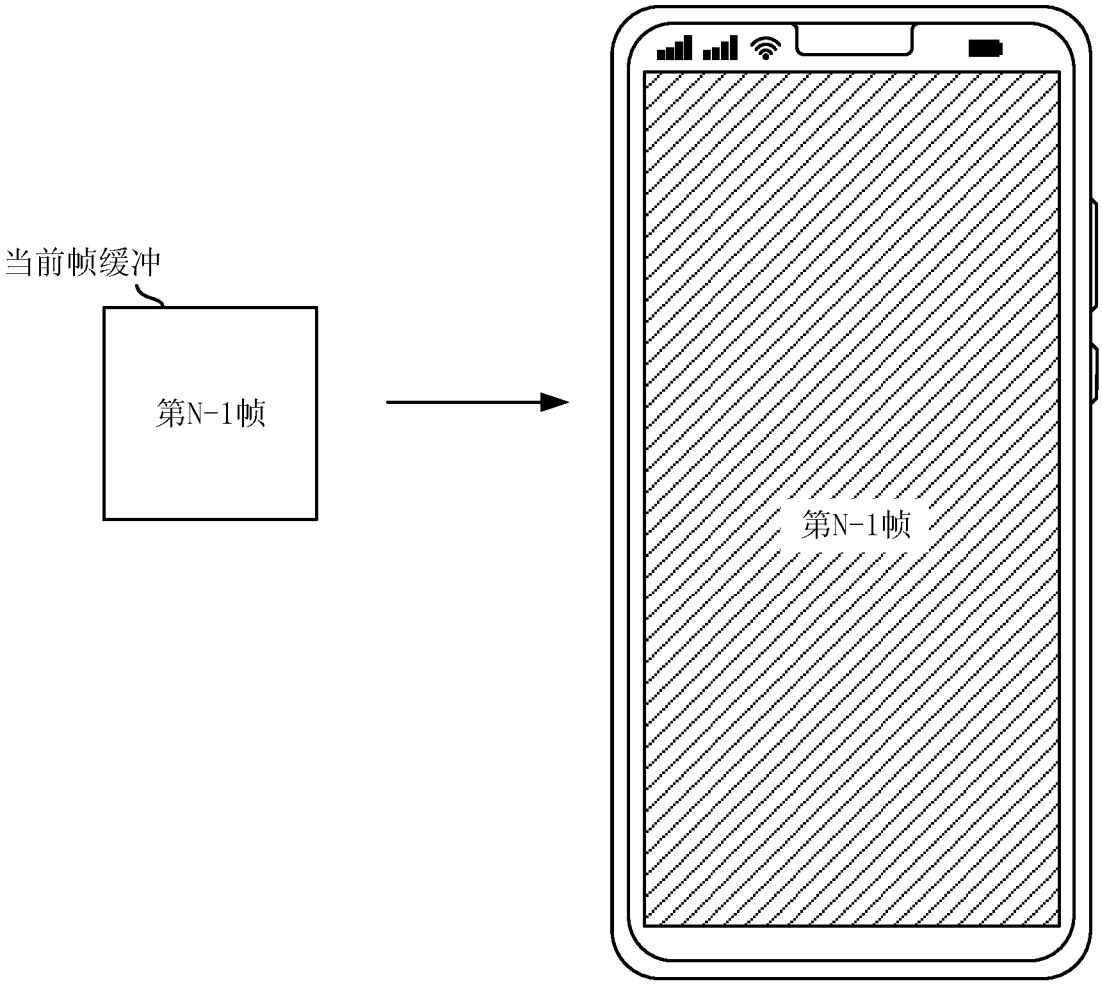



图 2

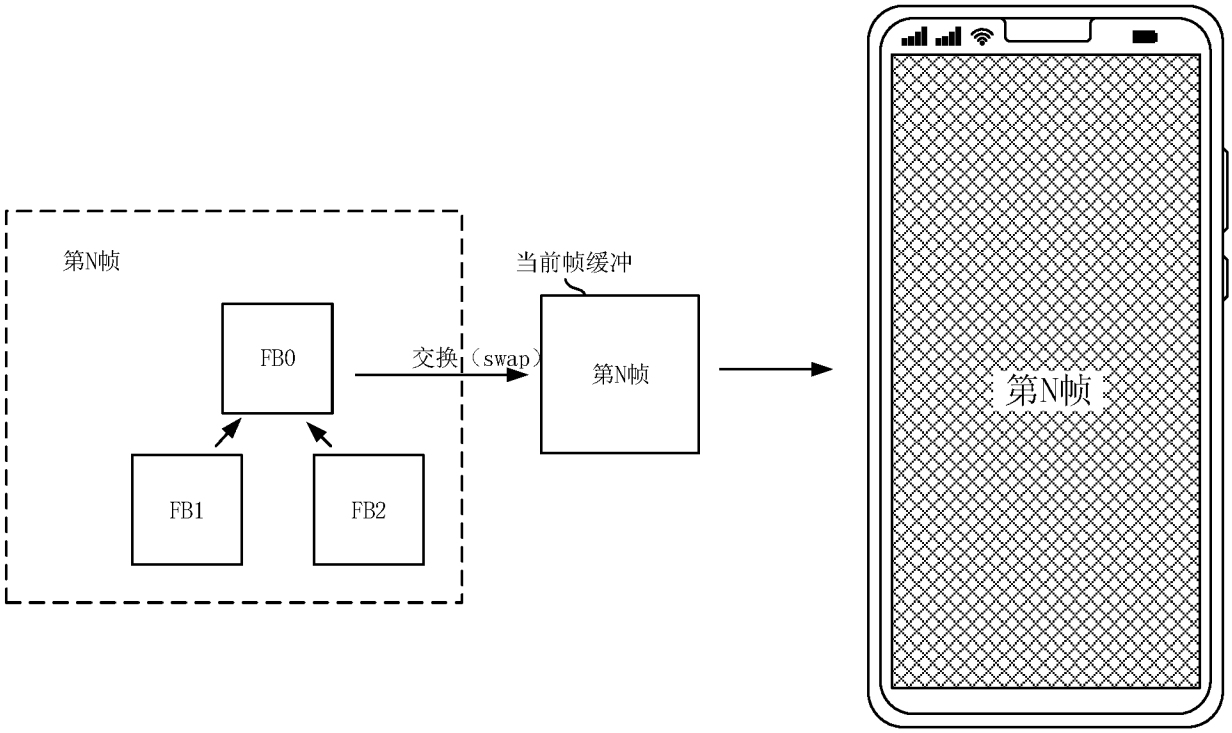



图 3

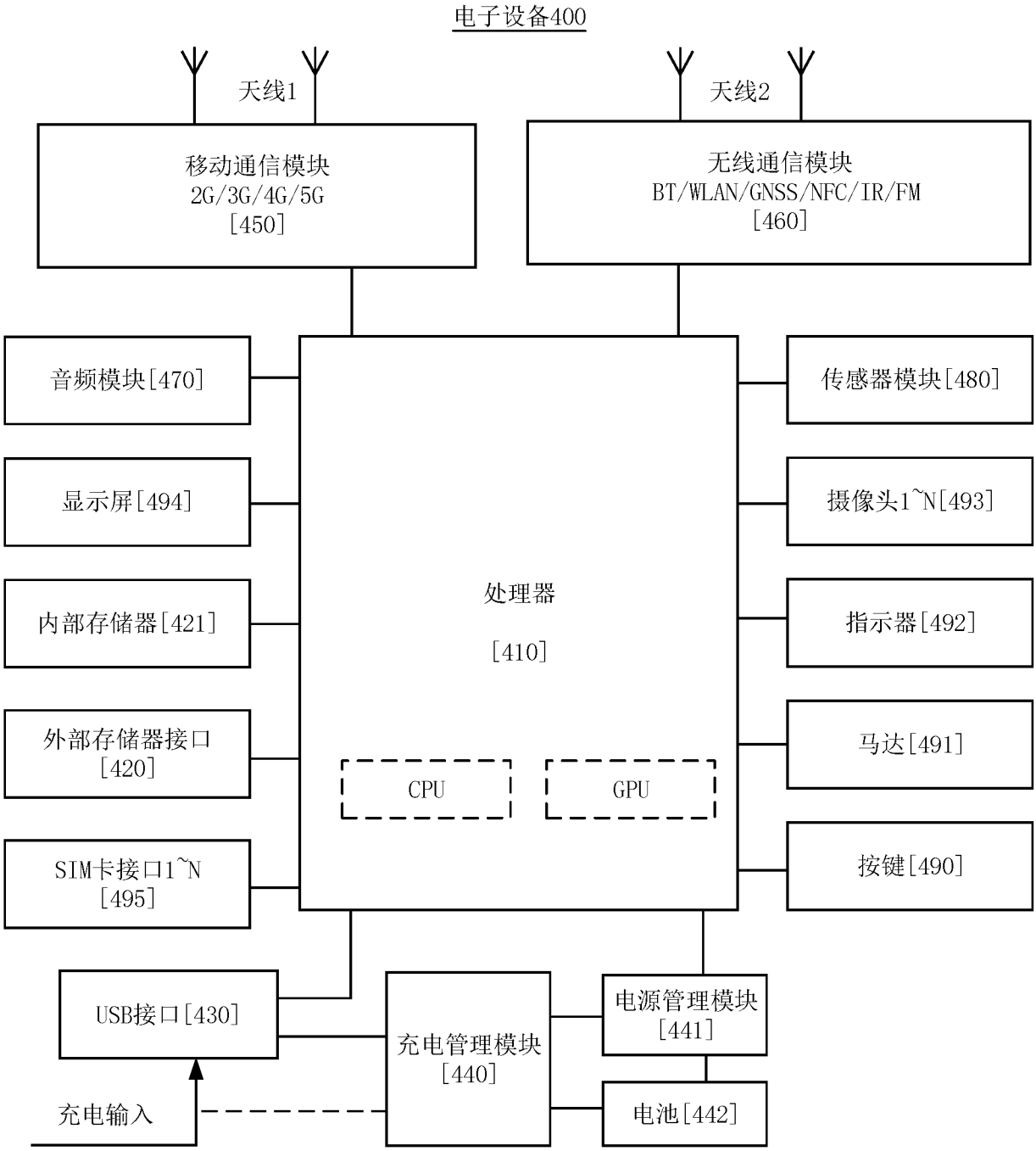

电子设备400
天线1
天线2
移动通信模块
2G/3G/4G/5G
[450]
无线通信模块
BT/WLAN/GNSS/NFC/IR/FM
[460]
音频模块[470]
传感器模块[480]
显示屏[494]
摄像头1~N[493]
处理器
[410]
内部存储器[421]
指示器[492]
外部存储器接口
[420]
马达[491]
CPU
GPU
SIM卡接口1~N
[495]
按键[490]
USB接口[430]
充电管理模块
[440]
电源管理模块
[441]
充电输入
电池[442]

图 4

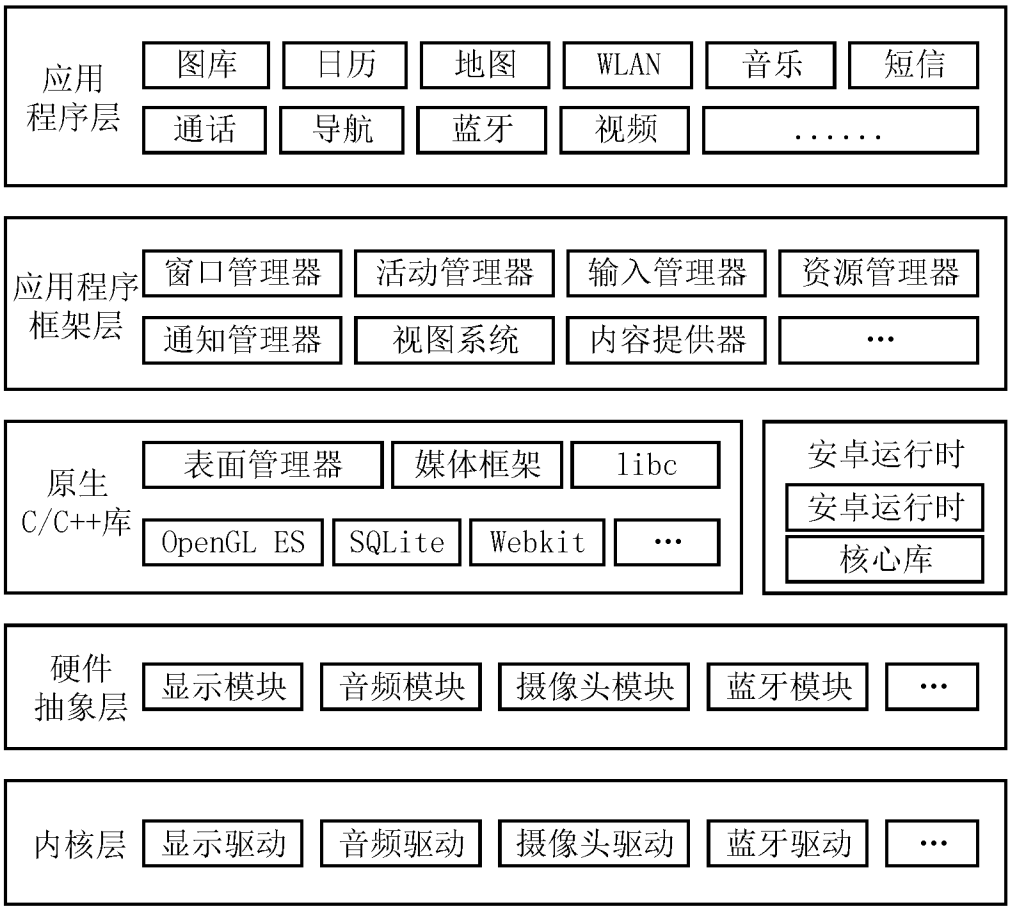

应用程序层
图库
日历
地图
WLAN
音乐
短信
通话
导航
蓝牙
视频
.....
应用程序框架层
窗口管理器
活动管理器
输入管理器
资源管理器
通知管理器
视图系统
内容提供器
…
原生C/C++库
表面管理器
媒体框架
libc
OpenGL ES
SQLite
Webkit
…
安卓运行时
安卓运行时
核心库
硬件抽象层
显示模块
音频模块
摄像头模块
蓝牙模块
…
内核层
显示驱动
音频驱动
摄像头驱动
蓝牙驱动
…

图 5

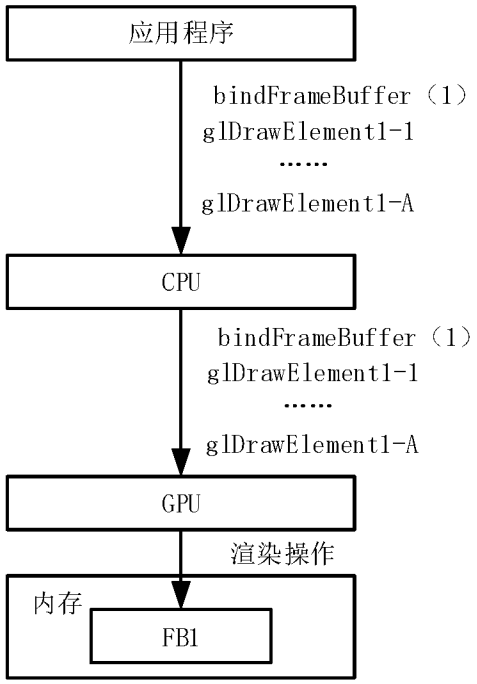

应用程序
bindFrameBuffer（1）
glDrawElement1-1
……
glDrawElement1-A
CPU
bindFrameBuffer（1）
glDrawElement1-1
……
glDrawElement1-A
GPU
渲染操作
内存
FB1

图 6A

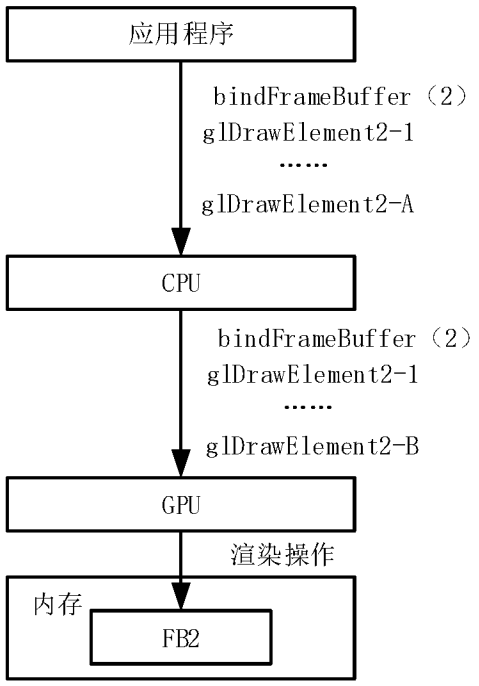

应用程序
bindFrameBuffer（2）
glDrawElement2-1
……
glDrawElement2-A
CPU
bindFrameBuffer（2）
glDrawElement2-1
……
glDrawElement2-B
GPU
渲染操作
内存
FB2

图 6B

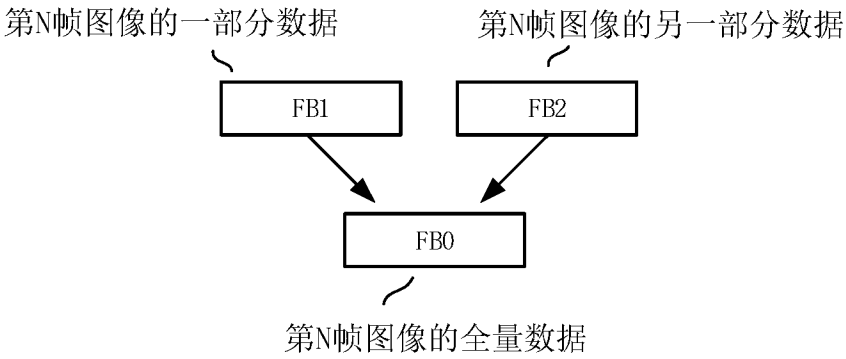

第N帧图像的一部分数据
第N帧图像的另一部分数据
FB1
FB2
FB0
第N帧图像的全量数据

图 6C

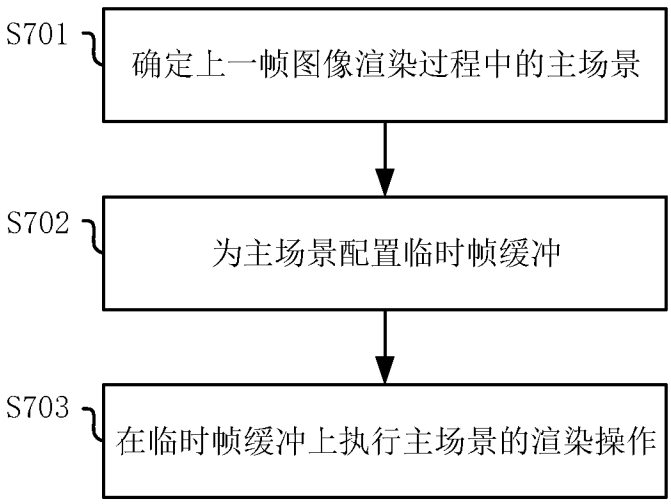

S701
确定上一帧图像渲染过程中的主场景
S702
为主场景配置临时帧缓冲
S703
在临时帧缓冲上执行主场景的渲染操作

图 7A

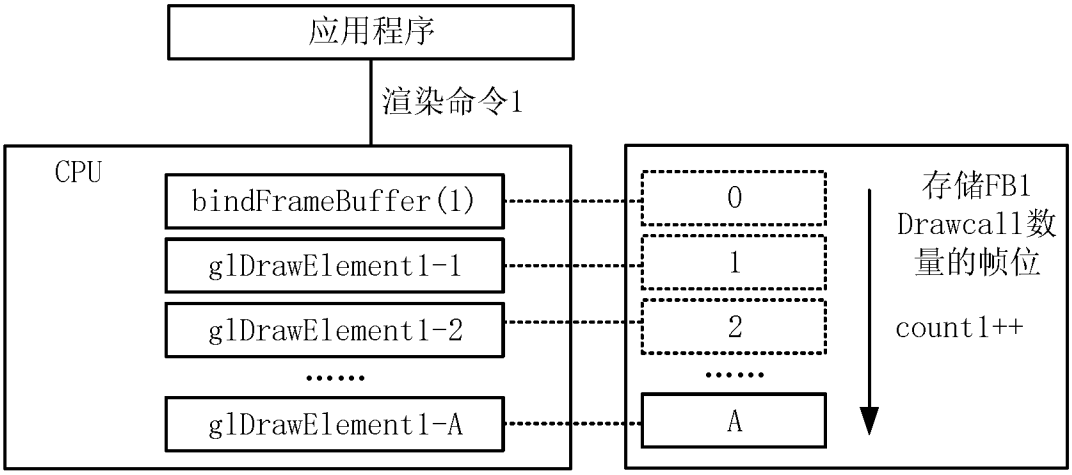

应用程序
渲染命令1
CPU
bindFrameBuffer(1)
glDrawElement1-1
glDrawElement1-2
……
glDrawElement1-A
0
1
2
……
A
存储FB1
Drawcall数
量的帧位
count1++

图 7B

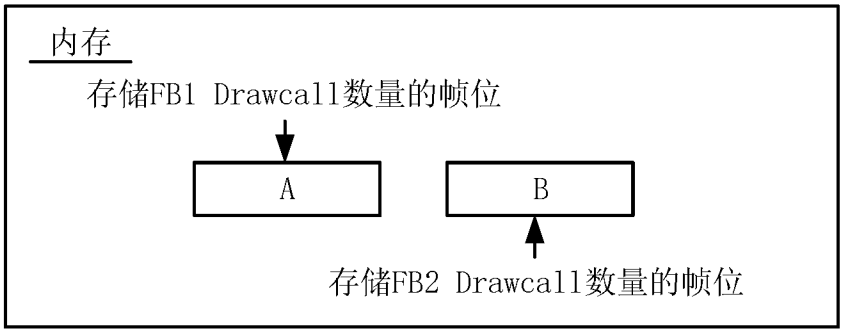

内存
存储FB1 Drawcall数量的帧位
A
B
存储FB2 Drawcall数量的帧位

图 7C

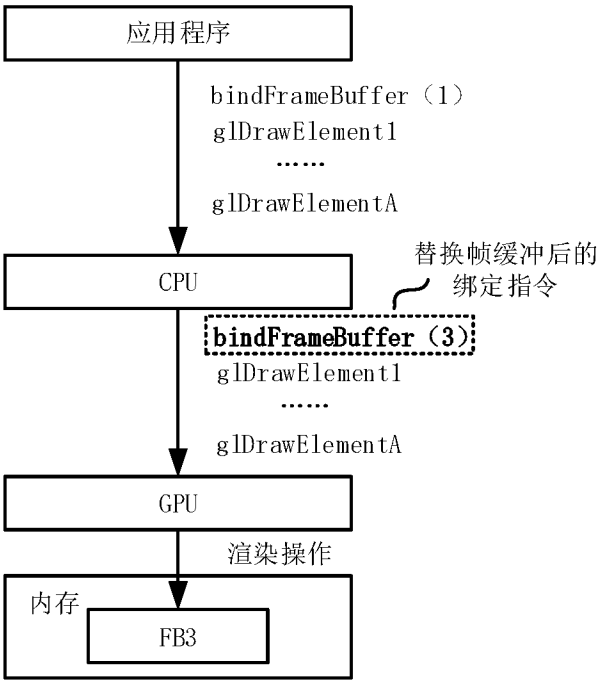

应用程序
bindFrameBuffer（1）
glDrawElement1
……
glDrawElementA
CPU
替换帧缓冲后的
绑定指令
bindFrameBuffer（3）
glDrawElement1
……
glDrawElementA
GPU
渲染操作
内存
FB3

图 8

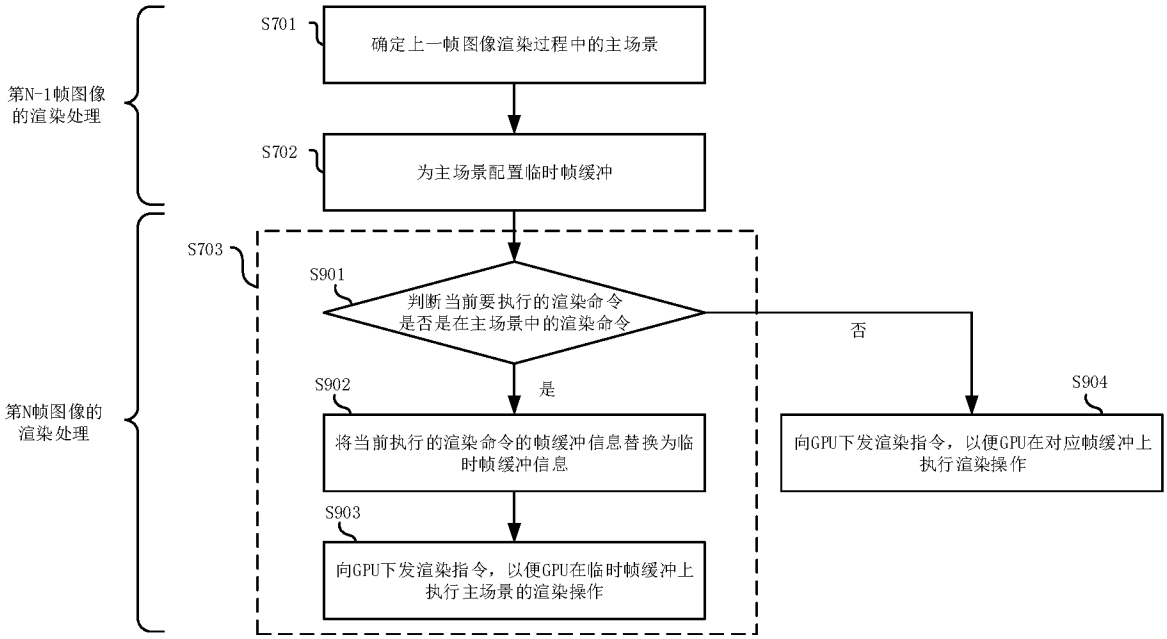

第N-1帧图像的渲染处理
第N帧图像的渲染处理
S701
确定上一帧图像渲染过程中的主场景
S702
为主场景配置临时帧缓冲
S703
S901
判断当前要执行的渲染命令是否是在主场景中的渲染命令
否
是
S902
将当前执行的渲染命令的帧缓冲信息替换为临时帧缓冲信息
S903
向GPU下发渲染指令，以便GPU在临时帧缓冲上执行主场景的渲染操作
S904
向GPU下发渲染指令，以便GPU在对应帧缓冲上执行渲染操作

图 9

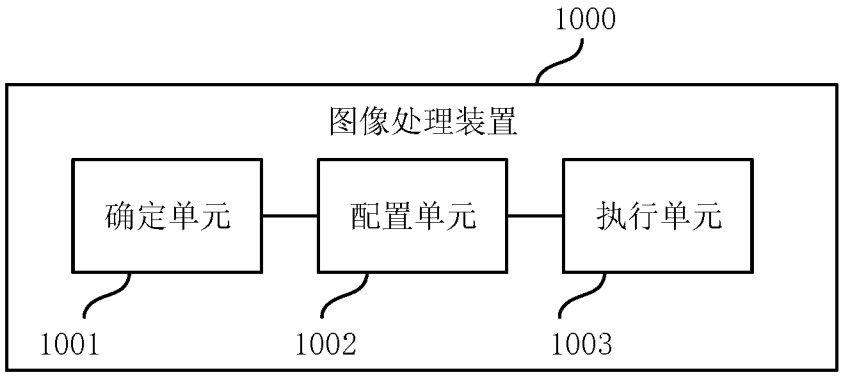

1000
图像处理装置
确定单元
配置单元
执行单元
1001
1002
1003

图 10

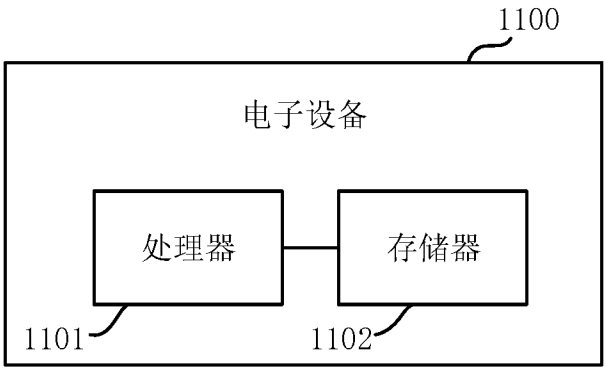

1100
电子设备
处理器
存储器
1101
1102

图 11

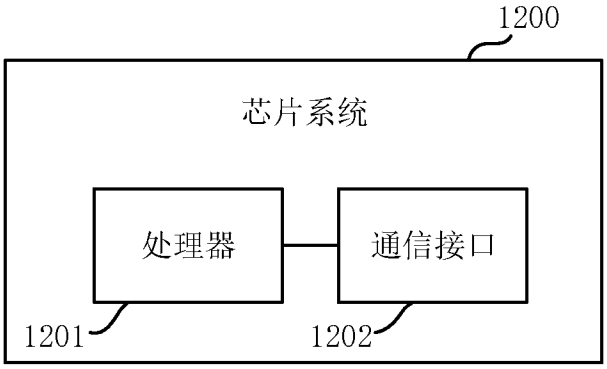



图 12

# INTERNATIONAL SEARCH REPORT

International application No.
**PCT/CN2022/097931**

## A. CLASSIFICATION OF SUBJECT MATTER

H04M 1/72439(2021.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

## B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

H04M

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

EPTXT; USTXT; VEN; WOTXT; CNABS; CNTXT; CNKI; IEEE: 分辨率, 分辨力, 像素, 象素, 精度, 渲染, 帧缓冲, 降低, 减少, 减轻, 减小, 消耗, 功耗, 算力, 资源, resolution, render+, power, resource, reduc+, consumption, frame, buffer

## C. DOCUMENTS CONSIDERED TO BE RELEVANT

| Category* | Citation of document, with indication, where appropriate, of the relevant passages | Relevant to claim No. |
|---|---|---|
| PX | CN 113726950 A (HONOR TERMINAL CO., LTD.) 30 November 2021 (2021-11-30)<br>description, paragraphs [0002]-[0221], and figures 1-12 | 1-20 |
| X | WO 2017092019 A1 (HUAWEI TECHNOLOGIES CO., LTD.) 08 June 2017 (2017-06-08)<br>description, page 11, line 6 to page 27, line 27, and figures 1-9 | 1-5, 9-12, 15-20 |
| A | CN 106204700 A (SAMSUNG ELECTRONICS CO., LTD.) 07 December 2016 (2016-12-07)<br>entire document | 1-20 |
| A | CN 109064538 A (OPPO GUANGDONG MOBILE TELECOMMUNICATIONS CO., LTD.) 21 December 2018 (2018-12-21)<br>entire document | 1-20 |

☐ Further documents are listed in the continuation of Box C. ☑ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

| Date of the actual completion of the international search | Date of mailing of the international search report |
|---|---|
| **13 August 2022** | **30 August 2022** |
| Name and mailing address of the ISA/CN<br>**China National Intellectual Property Administration (ISA/CN)**<br>**No. 6, Xitucheng Road, Jimenqiao, Haidian District, Beijing 100088, China**<br>Facsimile No. **(86-10)62019451** | Authorized officer<br><br>Telephone No. |

**INTERNATIONAL SEARCH REPORT**
**Information on patent family members**

International application No.
**PCT/CN2022/097931**

| Patent document cited in search report | | | Publication date (day/month/year) | Patent family member(s) | | | Publication date (day/month/year) |
|---|---|---|---|---|---|---|---|
| CN | 113726950 | A | 30 November 2021 | None | | | |
| WO | 2017092019 | A1 | 08 June 2017 | CN | 108369726 | A | 03 August 2018 |
| | | | | EP | 3373250 | A1 | 12 September 2018 |
| | | | | US | 2018357811 | A1 | 13 December 2018 |
| | | | | JP | 2019505037 | A | 21 February 2019 |
| | | | | US | 2020211254 | A1 | 02 July 2020 |
| | | | | KR | 20180088859 | A | 07 August 2018 |
| | | | | EP | 3373250 | A4 | 07 November 2018 |
| | | | | US | 10510175 | B2 | 17 December 2019 |
| | | | | JP | 6641484 | B2 | 05 February 2020 |
| | | | | EP | 3373250 | B1 | 08 April 2020 |
| | | | | KR | 102130361 | B1 | 05 August 2020 |
| | | | | US | 10872455 | B2 | 22 December 2020 |
| | | | | CN | 108369726 | B | 18 May 2021 |
| CN | 106204700 | A | 07 December 2016 | JP | 2016051469 | A | 11 April 2016 |
| | | | | KR | 20160026569 | A | 09 March 2016 |
| | | | | US | 2016063752 | A1 | 03 March 2016 |
| | | | | EP | 2991038 | A1 | 02 March 2016 |
| | | | | US | 9898853 | B2 | 20 February 2018 |
| | | | | CN | 106204700 | B | 01 September 2020 |
| | | | | EP | 2991038 | B1 | 02 December 2020 |
| | | | | KR | 102315280 | B1 | 20 October 2021 |
| CN | 109064538 | A | 21 December 2018 | None | | | |

# 国际检索报告

国际申请号
PCT/CN2022/097931

A. 主题的分类

H04M 1/72439(2021.01)i

按照国际专利分类(IPC)或者同时按照国家分类和IPC两种分类

B. 检索领域

检索的最低限度文献(标明分类系统和分类号)

H04M

包含在检索领域中的除最低限度文献以外的检索文献

在国际检索时查阅的电子数据库(数据库的名称，和使用的检索词（如使用）)

EPTXT;USTXT;VEN;WOTXT;CNABS;CNTXT;CNKI;IEEE：分辨率，分辨力，像素，象素，精度，渲染，帧缓冲，降低，减少，减轻，减小，消耗，功耗，算力，资源，resolution，render+，power，resource，reduc+，consumption，frame，buffer

C. 相关文件

| 类 型* | 引用文件，必要时，指明相关段落 | 相关的权利要求 |
|---|---|---|
| PX | CN 113726950 A (荣耀终端有限公司) 2021年11月30日 (2021-11-30)<br>说明书第[0002]-[0221]段，图1-12 | 1-20 |
| X | WO 2017092019 A1 (华为技术有限公司) 2017年6月8日 (2017-06-08)<br>说明书第11页第6行至第27页第27行，图1-9 | 1-5、9-12、15-20 |
| A | CN 106204700 A (三星电子株式会社) 2016年12月7日 (2016-12-07)<br>全文 | 1-20 |
| A | CN 109064538 A (OPPO广东移动通信有限公司) 2018年12月21日 (2018-12-21)<br>全文 | 1-20 |

☐ 其余文件在C栏的续页中列出。

☑ 见同族专利附件。

* 引用文件的具体类型：

"A" 认为不特别相关的表示了现有技术一般状态的文件

"E" 在国际申请日的当天或之后公布的在先申请或专利

"L" 可能对优先权要求构成怀疑的文件，或为确定另一篇引用文件的公布日而引用的或者因其他特殊理由而引用的文件（如具体说明的）

"O" 涉及口头公开、使用、展览或其他方式公开的文件

"P" 公布日先于国际申请日但迟于所要求的优先权日的文件

"T" 在申请日或优先权日之后公布，与申请不相抵触，但为了理解发明之理论或原理的在后文件

"X" 特别相关的文件，单独考虑该文件，认定要求保护的发明不是新颖的或不具有创造性

"Y" 特别相关的文件，当该文件与另一篇或者多篇该类文件结合并且这种结合对于本领域技术人员为显而易见时，要求保护的发明不具有创造性

"&" 同族专利的文件

| 国际检索实际完成的日期 | 国际检索报告邮寄日期 |
|---|---|
| 2022年8月13日 | 2022年8月30日 |
| ISA/CN的名称和邮寄地址<br>中国国家知识产权局(ISA/CN)<br>中国北京市海淀区蓟门桥西土城路6号 100088<br>传真号 (86-10)62019451 | 受权官员<br>张筱蓉<br>电话号码 (86-512)88996084 |

# 国际检索报告
## 关于同族专利的信息

国际申请号
PCT/CN2022/097931

| 检索报告引用的专利文件 | | | 公布日（年/月/日） | 同族专利 | | | 公布日（年/月/日） |
|---|---|---|---|---|---|---|---|
| CN | 113726950 | A | 2021年11月30日 | 无 | | | |
| WO | 2017092019 | A1 | 2017年6月8日 | CN | 108369726 | A | 2018年8月3日 |
| | | | | EP | 3373250 | A1 | 2018年9月12日 |
| | | | | US | 2018357811 | A1 | 2018年12月13日 |
| | | | | JP | 2019505037 | A | 2019年2月21日 |
| | | | | US | 2020211254 | A1 | 2020年7月2日 |
| | | | | KR | 20180088859 | A | 2018年8月7日 |
| | | | | EP | 3373250 | A4 | 2018年11月7日 |
| | | | | US | 10510175 | B2 | 2019年12月17日 |
| | | | | JP | 6641484 | B2 | 2020年2月5日 |
| | | | | EP | 3373250 | B1 | 2020年4月8日 |
| | | | | KR | 102130361 | B1 | 2020年8月5日 |
| | | | | US | 10872455 | B2 | 2020年12月22日 |
| | | | | CN | 108369726 | B | 2021年5月18日 |
| CN | 106204700 | A | 2016年12月7日 | JP | 2016051469 | A | 2016年4月11日 |
| | | | | KR | 20160026569 | A | 2016年3月9日 |
| | | | | US | 2016063752 | A1 | 2016年3月3日 |
| | | | | EP | 2991038 | A1 | 2016年3月2日 |
| | | | | US | 9898853 | B2 | 2018年2月20日 |
| | | | | CN | 106204700 | B | 2020年9月1日 |
| | | | | EP | 2991038 | B1 | 2020年12月2日 |
| | | | | KR | 102315280 | B1 | 2021年10月20日 |
| CN | 109064538 | A | 2018年12月21日 | 无 | | | |