

对象跳跃控制方法、装置、计算机设备及存储介质

本申请要求于 2019 年 05 月 23 日提交中国专利局，申请号为 201910436562.3，申请名称为“对象跳跃控制方法、装置、计算机设备及存储介质”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

本申请实施例涉及网络技术领域，特别涉及一种对象跳跃控制方法、装置、计算机设备及存储介质。

背景技术

目前，许多游戏中配置有与用户虚拟对象进行对抗的对抗虚拟对象，例如，在游戏场景中，僵尸形状的对抗虚拟对象可以移动至用户虚拟对象周围，对用户虚拟对象进行攻击。在对抗虚拟对象移动过程中，如果对抗虚拟对象移动至不可行走的区域时，服务器可以控制对抗虚拟对象进行跳跃，例如，从不可行走区域的一端跳跃至另一端。

相关技术中，对象跳跃控制过程可以包括：服务器基于用户选取的跳跃点，在每个不可行走区域人工配置一个链接桥，该链接桥包括用户选取的两个跳跃点。服务器可以控制对抗虚拟对象，从一个跳跃点跳跃至另一个跳跃点。

上述方法需要在每个不可行走区域人工配置链接桥，然而，游戏场景通常包括大量不可行走区域，需要人工配置大量链接桥，也会占用服务器的大量内存，从而消耗大量人力、内存等资源，成本较高，从而导致资源利用率较低。

发明内容

根据本申请提供的各种实施例，提供了一种对象跳跃控制方法、装置、计算机设备及存储介质。

一方面，提供了一种对象跳跃控制方法，由计算机设备执行，所述方法包括：

在虚拟场景中第一虚拟对象向目的地移动过程中，当所述第一虚拟对象的当前位置满足跳跃条件时，根据所述目的地的位置，确定所述第一虚拟对象的目标落地点；

根据所述当前位置和所述目标落地点，确定所述第一虚拟对象从所述当前位置跳跃至所述目标落地点的跳跃轨迹；及

根据所述跳跃轨迹，控制所述第一虚拟对象从所述当前位置向所述目标落地点跳跃。

另一方面，提供了一种对象跳跃控制装置，所述装置包括：

确定模块，用于在虚拟场景中第一虚拟对象向目的地移动过程中，当所述第一虚拟对象的当前位置满足跳跃条件时，根据所述目的地的位置，确定所述第一虚拟对象的目标落地点；

所述确定模块，还用于根据所述当前位置和所述目标落地点，确定所述第一虚拟对象从所述当前位置跳跃至所述目标落地点的跳跃轨迹；及

跳跃控制模块，用于根据所述跳跃轨迹，控制所述第一虚拟对象从所述当前位置向所述目标落地点跳跃。

在一种可能实现方式中，所述目的地是指所述虚拟场景中目标虚拟对象的当前位置，或者，所述虚拟场景中被选中的位置。

在一种可能实现方式中，所述跳跃条件包括：所述当前位置与所述目的地之间不包括行走路线，或者，在所述当前位置与所述目的地之间的最短路线方向上、所述当前位置前方包括障碍物。

在一种可能实现方式中，所述确定模块，还用于根据所述当前位置和所述目标落地点，确定所述第一虚拟对象在所述当前位置的跳跃速度；根据所述跳跃速度，确定所述第一虚拟对象从所述当前位置到所述目标落地点的跳跃轨迹。

另一方面，提供了一种计算机设备，包括存储器和处理器，所述存储器中存储有计算机可读指令，所述计算机可读指令被所述处理器执行时，使得所述处理器执行如上述的对象跳跃控制方法的步骤。

另一方面，提供了一个或多个存储有计算机可读指令的非易失性存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行如上述的对象跳跃控制方法的步骤。

本申请的一个或多个实施例的细节在下面的附图和描述中提出。本申请的其它特征、目的和优点将从说明书、附图以及权利要求书变得明显。

附图说明

为了更清楚地说明本申请实施例中的技术方案，下面将对实施例描述中所

需要使用的附图作简单地介绍，显而易见地，下面描述中的附图仅仅是本申请的一些实施例，对于本领域普通技术人员来讲，在不付出创造性劳动的前提下，还可以根据这些附图获得其他的附图。

- 图 1 是本申请实施例提供的一种对象跳跃控制方法的场景示意图；
- 图 2 是本申请实施例提供的一种对象跳跃控制方法的场景界面示意图；
- 图 3 是本申请实施例提供的一种对象跳跃控制方法的流程图；
- 图 4 是本申请实施例提供的一种对象状态检测的流程图；
- 图 5 是本申请实施例提供的一种对象状态检测的流程图；
- 图 6 是本申请实施例提供的一种跳跃阻挡物的示意图；
- 图 7 是本申请实施例提供的一种跳跃阻挡物的界面示意图；
- 图 8 是本申请实施例提供的一种跳跃轨迹的示意图；
- 图 9 是本申请实施例提供的一种对象跳跃控制的流程图；
- 图 10 是本申请实施例提供的一种对象跳跃控制装置的结构示意图；
- 图 11 是本申请实施例提供的一种终端的结构示意图；及
- 图 12 是本申请实施例提供的一种服务器的结构示意图。

具体实施方式

下面将结合本申请实施例中的附图，对本申请实施例中的技术方案进行清楚、完整地描述，显然，所描述的实施例是本申请一部分实施例，而不是全部的实施例。基于本申请中的实施例，本领域普通技术人员在没有做出创造性劳动前提下所获得的所有其他实施例，都属于本申请保护的范围。

本申请实施例主要涉及电子游戏场景，该电子游戏场景包括虚拟场景和虚拟对象，该虚拟对象是指虚拟场景中具备一定行为能力的对象，例如，跳跃、行走或者跑步等至少一种行为；该虚拟对象可以是用于代表用户形象的用户虚拟对象；或者，该虚拟对象可以是用于代表游戏中预先配置的、与用户虚拟对象进行对抗的对抗虚拟对象，例如，僵尸形状的对抗虚拟对象可以对用户虚拟对象进行攻击。或者，该虚拟对象还可以是用于代表在虚拟场景中用户虚拟对象有关联关系的用户附属虚拟对象，例如，该用户虚拟对象在虚拟场景中拥有的道具、虚拟宠物，或者搭载的载具等。该虚拟对象可以为任何一种形态，例如，人、动物或者虚拟出的任一种形状。例如，僵尸形状的对抗虚拟对象。

在一种可能场景中，对抗虚拟对象可以朝用户虚拟对象方向进行移动，对

抗虚拟对象移动至用户虚拟对象周围，对用户虚拟对象进行攻击，而用户也可以控制用户虚拟对象移动，以躲避对抗虚拟对象的攻击，或者，用户可以控制用户虚拟对象与对抗虚拟对象之间进行作战，例如，用户虚拟对象使用虚拟道具与对抗虚拟对象之间进行搏击。在对抗虚拟对象向用户虚拟对象移动过程中，对抗虚拟对象也可以采用跳跃行为，朝用户虚拟对象方向跳跃；例如，如图 1 所示，如果用户虚拟对象位于房屋的屋顶上，对抗虚拟对象位于地面，则对抗虚拟对象可以跳跃至屋顶。如图 2 所示，图 2 为图 1 的实际场景界面示意图，从图 2 可以更清晰的了解图 1 对应的虚拟场景。

在另一种可能场景中，用户也可以事先在虚拟场景中选定一个目的地，用户虚拟对象可以朝该目的地的方向进行移动，在移动过程中，用户虚拟对象也可以采用跳跃行为，例如，如果用户虚拟对象沿最短路线跑步过程中，如果遇到障碍物，例如，一条河流，用户虚拟对象也可以跳跃至河流的对岸。

在另一种可能场景中，用户附属虚拟对象也可以实时跟随用户虚拟对象进行移动，在实时跟随移动过程中，用户附属虚拟对象也可以采用跳跃行为，例如，用户虚拟对象在虚拟楼梯上快速前进时，用户虚拟对象的虚拟宠物狗也可以跳跃至用户虚拟对象当前所在阶梯上。

其中，该虚拟场景可以是对真实世界的仿真场景，也可以是半仿真半虚构的场景，还可以是纯虚构的场景。例如，该虚拟场景中可以包括天空、陆地或者海洋的至少一种，该陆地可以包括沙漠或者城市等至少一种环境元素。虚拟对象可以在该虚拟场景中移动，以射击类游戏为例，虚拟对象在该虚拟场景的天空中自由下落、滑翔或者打开降落伞进行下落等，在陆地上中跑动、跳动、爬行或者弯腰前行等，也可以在海洋中游泳、漂浮或者下潜等，当然，该虚拟对象也可以乘坐载具在该虚拟场景中进行移动，在此仅以上述场景进行举例说明，本申请实施例对此不作具体限定。

本申请实施例中，计算机设备可以预先获取游戏应用的游戏配置文件，该游戏配置文件可以包括该电子游戏的应用程序、页面显示数据或虚拟场景数据等，以使得该计算机设备开启该电子游戏时可以调用该游戏配置文件，对电子游戏页面进行渲染显示。该游戏配置文件中还可以包括虚拟对象的行为逻辑，该计算机设备可以基于该行为逻辑，控制该虚拟对象的跳跃行为。该计算机设备可以为终端，该终端实时基于该虚拟对象的行为逻辑，控制该虚拟对象进行跳跃，基于该跳跃过程，确定该跳跃过程所对应的游戏数据，并对该游戏数据

进行渲染显示，该游戏数据可以包括虚拟场景数据、该虚拟场景中虚拟对象的行为数据等。该计算机设备也可以为服务器，该服务器可以为该游戏应用的后台服务器，该服务器可以基于该虚拟对象的行为逻辑，实时向终端发送该虚拟对象的跳跃信息，例如，跳跃时的落地点、跳跃速度或者跳跃轨迹的至少一种，以使终端基于该跳跃信息，确定该跳跃过程所对应的游戏数据，并对该游戏数据进行渲染显示。

该终端可以为安装该应用程序的任一终端，本申请实施例对此不做具体限定。例如，终端可以为手机终端、PAD（Portable Android Device，平板电脑）终端或者电脑终端等任一安装有该电子游戏的设备。

图 3 是本申请实施例提供的一种对象跳跃控制方法的流程图。该发明实施例的执行主体为计算机设备，参见图 3，该方法包括：

301、计算机设备获取第一虚拟对象的目的地，控制第一虚拟对象向目的地移动。

本申请实施例中，该第一虚拟对象可以为用户虚拟对象、与用户虚拟对象进行对抗的对抗虚拟对象、用户虚拟对象的用户附属虚拟对象或者对抗虚拟对象的附属虚拟对象等。该目的地是指该虚拟场景中目标虚拟对象的当前位置，或者，该虚拟场景中被选中的位置。该目标虚拟对象也可以为用户虚拟对象、与用户虚拟对象进行对抗的对抗虚拟对象、用户虚拟对象的用户附属虚拟对象或者对抗虚拟对象的附属虚拟对象中的任一种。

在一些可能的实施场景中，该第一虚拟对象可以实时跟随该目标虚拟对象进行移动，在一种可能示例中，对抗虚拟对象追击用户虚拟对象；例如，僵尸虚拟对象追击目标场景范围内的用户虚拟对象。在另一种可能示例中，用户附属虚拟对象实时跟随用户虚拟对象进行移动，例如，虚拟宠物实时跟随用户虚拟对象进行移动。在另一种可能示例中，对抗虚拟对象中的附属虚拟对象实时跟随对抗虚拟对象进行移动，例如，小僵尸虚拟对象实时跟随大僵尸虚拟对象移动，或者，虚拟士兵跟随虚拟队长移动等。在另一种可能示例中，第一虚拟对象的任务协作对象也可以实时跟随第一虚拟对象。本申请实施例仅列举上述可能示例进行说明，但本申请实施例中该目标虚拟对象和第一虚拟对象具体表现形态并不做具体限定。

在一种可能的实施方式中，该目标虚拟对象可以包括但不限于：第一虚拟对象的目标场景范围内的第二虚拟对象、与第一虚拟对象具备对抗关联关系的

第三虚拟对象或者第一虚拟对象的任务协作对象等。本步骤中，该计算机设备可以基于以下四种方式中任一种，确定该第一虚拟对象的目的地。

第一种方式、当该第一虚拟对象对应的目标场景范围内包括第二虚拟对象时，该计算机设备将该第二虚拟对象确定为该目标虚拟对象，获取该目标虚拟对象的当前位置，将该目标虚拟对象的当前位置确定为该目的地。

在一种可能示例中，该目标虚拟对象可以为用户虚拟对象，该第一虚拟对象可以为用户虚拟对象的对抗虚拟对象，计算机设备可以在一定场景范围内配置一个或多个对抗虚拟对象，每个对抗虚拟对象对应一个目标场景范围，当计算机设备检测到对抗虚拟对象对应的目标场景范围内包括用户虚拟对象时，则将该用户虚拟对象的当前位置作为目的地，控制对抗虚拟对象追击用户虚拟对象。

在另一种可能示例中，该目标虚拟对象和第一虚拟对象可以为用户虚拟对象，例如，该目标虚拟对象和第一虚拟对象可以属于同一团队，该第一虚拟对象的自动跟随模式为开启状态时，当第一虚拟对象的目标场景范围内包括属于同一团队中的队友虚拟对象时，该计算机设备将该队友虚拟对象的当前位置作为该目的地，控制第一虚拟对象实时跟随队友虚拟对象进行移动。

第二种方式、该计算机设备将与该第一虚拟对象具备对抗关联关系的第三虚拟对象确定为该目标虚拟对象，获取该目标虚拟对象的当前位置，将该目标虚拟对象的当前位置确定为该目的地。

在一种可能的示例中，该目标虚拟对象可以为用户虚拟对象，该第一虚拟对象可以为用户虚拟对象的对抗虚拟对象，计算机设备可以存储第一虚拟对象具备和第三虚拟对象的对抗关联关系，例如，对抗虚拟对象 A 实时追击的用户虚拟对象 B，则对抗虚拟对象 A 的目的地为该用户虚拟对象 B 的当前位置。

第三种方式、该计算机设备将该第一虚拟对象的任务协作对象确定为该目标虚拟对象，获取该目标虚拟对象的当前位置，将该目标虚拟对象的当前位置确定为该目的地。

在一种可能的示例中，该第一虚拟对象可以为用户虚拟对象或者对抗虚拟对象，该第一虚拟对象的任务协作对象可以为具备虚拟附属关系的虚拟对象或者属于同一团队的队友虚拟对象或者，例如，虚拟宠物、虚拟队长的士兵虚拟对象或者小僵尸虚拟对象。例如，用户虚拟对象的虚拟宠物可以实时跟随用户虚拟对象进行移动。

第四种方式、该计算机设备获取该虚拟场景中被选中的位置，将该目标虚拟对象的当前位置确定为该目的地。

在一种可能场景中，该目的地可以为预先被选中的场景位置，例如，预先选中一个目的地，多个虚拟对象迅速抢占该目的地，例如，山顶或者武器库门前等。

需要说明的是，该计算机设备可以将目标虚拟对象的当前位置作为该目的地，也可以将用户选中的位置作为目的地，控制第一虚拟对象进行移动，且该目标虚拟对象可以与第一虚拟对象具备对抗关系、任务协作关系等，丰富了对对象跳跃控制方法的适用性。

302、在虚拟场景中第一虚拟对象向目的地移动过程中，该计算机设备根据该第一虚拟对象的当前位置，检测该第一虚拟对象的当前位置是否满足跳跃条件。

本申请实施例中，该计算机设备可以获取该第一虚拟对象的当前位置，根据该当前位置和该目的地，检测该第一虚拟对象的当前位置是否满足跳跃条件。

在一种可能的实施方式中，该跳跃条件可以为：该当前位置与该目的地之间不包括行走路线。该行走路线是指采用行走方式进行移动的路线。本步骤中，该计算机设备可以根据该当前位置，检测该当前位置与目的地之间是否包括行走路线，当该当前位置与该目的地之间不包括行走路线时，该计算机设备确定该第一虚拟对象的当前位置满足跳跃条件。在一种可能示例中，该计算机设备可以采用导航网格表示该虚拟场景中各个场景对象以及虚拟对象的位置，例如，房屋、河流、集装箱或者载具等位置。该导航网格包括多个多边形网格，每个多边形网格用于表示虚拟场景中的一个位置节点，该计算机设备还可以基于该导航网格，确定该当前位置与该目的地之间的行走路线。本步骤中，该计算机设备可以根据该当前位置所在的位置节点，采用目标寻路算法，查找从当前位置所在位置节点到目的地所在位置节点之间的行走路线，如果没有查找到从当前位置所在位置节点到目的地所在位置节点的行走路线，则确定该当前位置满足跳跃条件，否则，确定该当前位置不满足跳跃条件。

在一种可能示例中，该目标寻路算法可以基于需要进行设置，本申请实施例对此不做限制，例如，该目标寻路算法可以为 A*（A-Star，启发式搜索）算法，该导航网格可以为 Nav Mesh（导航网格），每个多边形网格周围包括多个相邻多边形，相邻多边形网格之间是连通的，虚拟对象可以从多边形网格行走

至相邻多边形网格。相邻多边形网格是指多边形网格周围与该多边形网格有公共边的多边形网格。行走路线查找过程可以包括：计算机设备将当前位置所在多边形网格作为起点，将目的地所在多边形网格作为终点，计算机设备通过 A* 算法，依次查找从起点到终点的多个中间节点，从而查找出一条路线，当查找失败时，即表示两点间不可达。

需要说明的是，计算机设备在查找路线时，当查找到一个中间节点时，可以基于当前中间节点和终点，继续查找下一个中间节点，直至到达终点，且该计算机设备无需存储查找出的多个中间节点，只需判断出该当前位置和目的地之间存在行走路线即可，不需要获取具体的行走路线所包括的各个中间节点。因此，节省了存储时间，可以快速确定出路线查找结果，提高了处理效率。

在另一种可能的实施方式中，该跳跃条件还可以为：在该当前位置与该目的地之间的最短路线方向上、该当前位置前方包括障碍物。本步骤中，该计算机设备可以确定该当前位置和目的地之间的最短路线，该最短路线可以是当前位置到目的地之间的直线方向，在第一虚拟对象朝目的地移动过程中，该计算机设备可以控制该第一虚拟对象沿最短路线移动，该计算机设备可以在沿该最短路线方向上，检测该当前位置前方的第一目标范围内是否包括障碍物，当该当前位置前方第一目标范围内包括障碍物时，该计算机设备确定该当前位置满足跳跃条件（将本步骤中目标范围称为第一目标范围，以区分步骤 303 中第二目标范围）。该第一目标范围可以基于需要进行设置，例如，该第一目标范围是指该第一虚拟对象的前方 1 米或 2 米内的范围。该障碍物是指阻挡第一虚拟对象行走的物理对象，例如，河流、石块或者栏杆的至少一种。

本申请实施例中，该计算机设备可以周期性执行步骤 302 的过程，当该当前位置满足跳跃条件时，执行后续步骤 303-306 的过程，也即是，对象跳跃过程。在一种可能的实施方式中，该第一虚拟对象可以为游戏中预先配置的 AI（Artificial Intelligence，人工智能）对象，该 AI 对象可以用于与用户虚拟对象进行对抗，例如，僵尸形状的对抗虚拟对象，该 AI 对象可以基于预先配置的行为树执行跳跃过程，行为树是指由 AI 对象行为节点组成、用于指示 AI 对象的行为逻辑的树状结构，行为节点的执行受到父节点的影响和控制。本步骤还可以为：当该第一虚拟对象为 AI 对象时，该计算机设备可以在该第一虚拟对象的行为树中第一节点，该第一节点用于指示按照目标周期执行步骤 302 的检测过程，从而实现周期行更新该当前位置的检测状态。该计算机设备还可以在该行

为树中添加第二节点，该第二节点用于指示当该当前位置满足跳跃条件时，执行步骤 303 的过程。如图 4 所示，当该 AI 对象的当前位置满足跳跃条件时，例如，AI 对象不可行走至用户虚拟对象时，便执行后续步骤 303-306 的跳跃过程；当该 AI 对象的当前位置不满足跳跃条件时，便执行 AI 对象的正常寻路攻击逻辑，也即是，控制该 AI 对象行走至用户虚拟对象并攻击用户虚拟对象。

在一种可能示例中，该第一节点可以为 service（服务）节点，该第二节点可以为任务节点，例如，decorator（装饰器）节点。如图 5 所示，以 service 节点为例进行说明，该 service 节点中可以设置一秒执行一次步骤 302 的过程，以判断当前位置是否满足跳跃条件，例如，AI 对象是否可行走至用户虚拟对象，当该 AI 对象不可行走至用户虚拟对象时，计算机设备将该 AI 对象的状态设置为不可达，否则，设置为可达，当检测到该 AI 对象的状态为不可达时，执行后续步骤 302-305 的跳跃过程。

303、当该第一虚拟对象的当前位置满足跳跃条件时，计算机设备根据该目的地的位置，确定该第一虚拟对象的目标落地点。

该目标落地点是指该第一虚拟对象执行跳跃动作所要到达的位置。本申请实施例中，该计算机设备可以在该目的地的行走区域内选取一点，作为该目标落地点。或者，该计算机设备还可以在目的地的较近位置选取一点，作为该目标落地点，又或者，该计算机设备还可以直接将目的地作为目标落地点。相应的，本步骤可以包括以下四种实现方式中的任一种方式。

第一种方式、当该当前位置满足该跳跃条件时，该计算机设备基于该目的地的行走区域，确定位于该行走区域内的目标落地点。

其中，该行走区域内任一位置与该目的地之间包括行走路线，也即是，该行走区域内的任一位置上的第一虚拟对象，均可以行走至该目的地。本步骤中，该目的地可以为虚拟场景中目标虚拟对象的当前位置。

本申请实施例中，该虚拟场景中包括多个行走区域，该计算机设备可以根据该目的地的位置，将包括该目的地在内的行走区域，确定该目的地的行走区域。在一种可能的实施方式中，该计算机设备还可以选取该目的地所在的行走区域中的较小范围的区域，作为该目的地的行走区域，该过程可以包括：该计算机设备根据该目的地的位置，将包括该目的地在内的目标范围内的行走区域，确定为该目的地的行走区域。例如，该计算机设备可以以该目的地为中心，获取距离该目的地不超过第一目标距离的行走区域。

该计算机设备根据该目的地的位置，获取该目的地所在的行走区域，该计算机设备还可以继续获取以该目的地为中心、半径为第一目标距离的圆形区域，作为该目的地的行走区域。例如，获取半径为 5 米的圆形行走区域。

在一种可能的实施方式中，该计算机设备可以基于该第一虚拟对象的跳跃速度，选取该目标落地点，也即是以下步骤 a 的过程；在另一种可能的实施方式中，该计算机设备还可以基于用户选择，获取该跳跃落点，也即是以下步骤 b 的过程。

步骤 a、该计算机设备根据该第一虚拟对象的最大跳跃速度，获取该最大跳跃速度在该行走区域内对应的目标落地点。

本申请实施例中，该第一虚拟对象的跳跃速度范围有限，该计算机设备可以根据该第一虚拟对象的最大跳跃速度，获取一个使得该第一虚拟对象跳跃最远的点，作为该目标落地点。在一种可能的实施方式中，该计算机设备可以获取该第一虚拟对象的最大跳跃速度，根据该最大跳跃速度，确定该第一虚拟对象的最大跳跃距离，该计算机设备根据该最大跳跃距离和该当前位置，获取该行走区域内该最大跳跃距离对应的目标落地点。

在一种可能示例中，该目的地和该第一虚拟对象的当前位置位于同一水平面时，该计算机设备可以根据该第一虚拟对象的最大跳跃速度，确定该第一虚拟对象在水平面上的最远跳跃距离，该计算机设备根据该最远跳跃距离，在第一虚拟对象的行走区域中确定该最远跳跃距离对应的目标落地点。

在另一种可能实例中，当该目的地和该第一虚拟对象的当前位置位于不同水平高度时，该计算机设备可以根据该第一虚拟对象的最大跳跃速度，分别确定该第一虚拟对象在水平面和垂直面上的最远跳跃距离，该计算机设备根据该第一虚拟对象在水平面和垂直面上的最远跳跃距离，在第一虚拟对象的行走区域中确定该最远跳跃距离对应的目标落地点。

步骤 b、该计算机设备获取该行走区域内被选中的位置，将该被选中的位置确定为该目标落地点。

本步骤中，该第一虚拟对象可以为用户虚拟对象，该目的地可以为目标虚拟对象的当前位置，或者，该虚拟场景中被选中的位置。

当计算机设备为终端时，该计算机设备可以在应用界面中显示该目的地的行走区域，用户可以在该行走区域内选择该第一虚拟对象的目标落地点。该计算机设备获取该用户选中的目标落地点。在一种可能示例中，该计算机设备还

可以获取用户选中的目标落地点后，将该用户选中的目标落地点发送至服务器。

当该计算机设备为服务器时，用户所在的终端可以显示该目的地的行走区域，终端将该用户选中的目标落地点发送至该计算机设备。

需要说明的是，该第一虚拟对象的跳跃速度有限，而使得该第一虚拟对象跳跃距离有限，该计算机设备可以结合第一虚拟对象的最大跳跃速度，在行走区域内选择该最大跳跃速度所能到达的点，作为该目标落地点，使得第一虚拟对象的一次跳跃，便尽可能跳跃至较远的距离，避免虚拟对象跳跃较短距离时还需跑步移动，如果该第一虚拟对象为用户虚拟对象，还可以避免损失用户虚拟对象过多的生命力值、血量等，最大化提高了第一虚拟对象在游戏中的虚拟能力。并且，该计算机设备还可以为用户提供选择入口，将用户选取的点作为目标落地点，更好的满足用户需要，提高了用户体验。

第二种方式、当该当前位置满足该跳跃条件时，该计算机设备将该虚拟场景中离该目的地最近的位置确定为该目标落地点。

本步骤中，该目的地可以为虚拟场景中目标虚拟对象的当前位置。该计算机设备可以根据该目的地的位置，以该目的地为中心，获取该目的地的多个相邻位置，该计算机设备获取每个相邻位置与该目的地之间的距离，根据该多个距离，从该多个相邻位置中选取距离该目的地最近的位置，作为该目标落地点。

在一种可能示例中，如果该计算机设备采用导航网格表示虚拟场景中对象的位置，该计算机设备可以根据该目的地所在的多边形网格，获取该目的地所在的多边形网格的多个相邻多边形网格，依次计算每个相邻多边形网格与目的地所在多边形网格之间的距离，从而确定出离该目的地所在多边形网格最近的相邻多边形网格，将该相邻多边形网格对应的位置节点，作为该目标落地点。

第三种方式、当该当前位置满足该跳跃条件时，该计算机设备将该虚拟场景中该目的地的相邻位置确定为该目标落地点。

本步骤中，该目的地可以为虚拟场景中目标虚拟对象的当前位置。在一种可能示例中，如果该计算机设备采用导航网格表示虚拟场景中对象的位置，该计算机设备可以根据该目的地所在的多边形网格，获取该目的地所在的多边形网格的任一相邻多边形网格，将该任一相邻多边形网格对应的位置节点，作为该目标落地点。

第四种方式、当该当前位置满足该跳跃条件时，该计算机设备将该目的地确定为该目标落地点。

本步骤中，该目的地可以为虚拟场景中被选中的位置，该计算机设备还可以获取导航网格中该目的地所在的多边形网格，将该目的地所在的多边形网格对应的位置节点，作为该第一虚拟对象的目标落地点。该第一虚拟对象可以直接跳跃至该目的地。

需要说明的是，当该目的地为目标虚拟对象的当前位置时，该计算机设备可以基于上述第一种方式至第三种方式，基于该目的地的行走区域中的位置、相邻位置、最近位置等，确定该目标落地点，从而避免第一虚拟对象和目标虚拟对象之间发生碰撞。当该目的地为事先选中的位置时，该计算机设备可以通过上述第四种方式，直接将该目的地作为该目标落地点，节省了计算时间，提高了对象跳跃控制方法的适用性。

304、计算机设备检测该当前位置与目的地之间是否包括跳跃阻挡物，当该当前位置与目的地之间包括跳跃阻挡物时，对该第一虚拟对象进行移动，获取该第一虚拟对象移动后的当前位置。

跳跃阻挡物是指第一虚拟对象起跳之后、到达目标落地点之前在空中会碰到的物体。当该当前位置与目的地之间不包括跳跃阻挡物时，该计算机设备直接执行步骤 305。本步骤中，该计算机设备可以基于该当前位置和该目的地之间的距离，检测该当前位置的第二目标范围内，是否包括跳跃阻挡物，当该第二目标范围内包括跳跃阻挡物时，该计算机设备控制该第一虚拟对象朝远离该跳跃阻挡物的方向移动目标移动距离，也即是，控制该第一虚拟对象后退目标移动距离，该计算机设备获取移动后的当前位置。

在一种可能的实施方式中，该计算机设备还可以检测第二目标范围内跳跃阻挡物的高度，当该跳跃阻挡物的高度超过目标高度时，再控制该第一虚拟对象朝远离该跳跃阻挡物的方向移动目标移动距离。该第二目标范围可以为该目的地方向上距离该当前位置不超过第二目标距离的范围，该第二目标距离、该目标高度和该目标移动距离可以基于需要进行设置，本申请实施例对此不做具体限定，例如，该第二目标距离可以为 2 米或者 3 米等。该目标高度可以为 10 米或者 8 米等。该目标移动距离可以为 3 米或者 1 米等。在一种可能示例中，该计算机设备可以存储跳跃阻挡物的多个高度与多个移动距离之间的对应关系。该计算机设备获取该第二目标范围内跳跃阻挡物的高度，根据该多个高度和移动距离之间的对应关系，获取该跳跃阻挡物的高度所对应的目标移动距离，控制该第一虚拟对象朝远离该跳跃阻挡物的方向移动该目标移动距离。例如，

当跳跃阻挡物高度为 10 米时, 该第一虚拟对象对应后退 3 米; 当该跳跃阻挡物高度为 15 米时, 该第一虚拟对象对应后退 5 米等。

如图 6 所示, 当该第一虚拟对象面前包括一个较高的集装箱时, 基于预测的可能跳跃轨迹可以得知, 该第一虚拟对象起跳之后、到达目标落地点之前在空中会碰到集装箱, 该集装箱为该第一虚拟对象的跳跃阻挡物。则该计算机设备可以控制该第一虚拟对象后退一定距离后再跳跃。如图 7 所示, 图 7 为图 6 所示虚拟场景的实际场景界面示意图, 从图 7 可以更清晰的了解该虚拟场景的实际形态。

305、计算机设备根据该当前位置和该目标落地点, 确定该第一虚拟对象从该当前位置跳跃至该目标落地点的跳跃轨迹。

本申请实施例中, 该计算机设备可以先确定出可从当前位置跳跃至目标落地点的跳跃速度, 基于该跳跃速度, 确定该跳跃轨迹。本步骤可以包括: 该计算机设备可以根据该当前位置和该目标落地点, 确定该第一虚拟对象在该当前位置的跳跃速度, 该计算机设备根据该跳跃速度, 确定该第一虚拟对象从该当前位置到该目标落地点的跳跃轨迹。该跳跃速度包括该跳跃速度的速度大小和速度方向。

在一种可能的实施方式中, 该跳跃轨迹可以为抛物线轨迹, 该计算机设备可以根据该当前位置和目标落地点, 根据以下公式一, 确定该第一虚拟对象的跳跃速度;

$$\text{公式一: } \begin{cases} x = v_0 t \cos(\theta) \\ y = v_0 t \sin(\theta) - \frac{1}{2} g t^2; \end{cases}$$

其中, x 用于表示当前位置到目标落地点的水平位移, y 用于表示当前位置到目标落地点的垂直于水平面的垂直位移, v_0 用于表示跳跃速度的速度大小, θ 用于表示跳跃速度的速度方向, g 表示重力加速度。

本申请实施例中, θ 可以为速度方向与水平面的夹角, θ 如公式二所示, 公

式二: $\theta = \arctan\left(\frac{v^2 \pm \sqrt{v^4 - g(gx^2 + 2yv^2)}}{gx}\right)$; 该计算机设备可以获取该第一虚拟对象

的速度范围, 基于该速度范围包括的多个速度值, 获取满足上述公式一的速度值以及速度方向, 例如, 该计算机设备可以从该速度范围中速度大小的最小值 $v_{\min}$ 开始, 代入上述公式一中, 判断是否满足上述公式一, 如果不满足, 则将该

最小值增大目标阈值，增大为 $v_{\min} + \Delta v$ ，代入上述公式一，直到获取到该满足上述公式一的跳跃速度。

在一种可能的实施方式中，如图 8 所示，该计算机设备基于上述公式一，确定出两个跳跃轨迹，图 8 中左侧为直接向下跳跃时对应的第一跳跃轨迹，图 8 中右侧图为先向上跳跃再落地时对应的第二跳跃轨迹。图 8 中，两个坐标系的横坐标为水平方向上的距离，纵坐标为垂直方向上的距离，当前位置为 $(0, 200)$ ，目标落地点为 $(200, 0)$ ，单位可以为米，该计算机设备可以选择先向上跳跃再落地对应的第二跳跃轨迹。在另一种可能的实施方式中，该计算机设备还可以结合该当前位置和该目标落地点之间的相对高度，确定该跳跃轨迹。当该当前位置的高度高于该目标落地点时，该计算机设备也可以选择直接向下跳跃时对应的第一跳跃轨迹；当该当前位置的高度低于该目标落地点时，该计算机设备选择先向上跳跃再落地时对应的第二跳跃轨迹。

需要说明的是，如果该第一虚拟对象为 AI 对象，该计算机设备可以采用在该 AI 对象的行为树中添加节点的方式，实现上述步骤 302-305 的过程，以对该虚拟对象的跳跃过程进行控制，在一种可能示例中，该计算机设备可以将该步骤 302 的检测逻辑配置在第一节点中，将步骤 303-305 的跳跃逻辑配置在第二节点中，在行为树中添加该第一节点和第二节点，即可实现虚拟对象的跳跃控制过程；由于修改 AI 对象的行为树，而无需修改该 AI 对象的原有的 AI 逻辑，例如，底层的寻路攻击逻辑，该行为树中的节点可以随时添加以及删除，使得本申请实施例的方法实现更加便捷，而且，也无需在导航网格中配置大量的链接桥，从而提高了对象跳跃控制方法的适用性。

306、计算机设备根据该跳跃轨迹，控制该第一虚拟对象从该当前位置向该目标落地点跳跃。

该计算机设备可以根据该跳跃轨迹，在该第一图像和第二图像之间插入显示该跳跃过程对应的多个中间图像。本步骤中，该计算机设备可以根据该跳跃轨迹，获取该第一虚拟对象在离开该当前位置之后、到达该目标落地点之前的多个空中位置；该计算机设备基于该多个空中位置，获取该第一虚拟对象的多个中间图像，每个中间图像用于显示位于一个空中位置的第一虚拟对象；该计算机设备基于该多个中间图像的播放顺序，播放该多个中间图像。

需要说明的是，如果该计算机设备为终端，则基于上述步骤，显示第一虚拟对象从该当前位置向该目标落地点跳跃的过程。如果该计算机设备为服务器，

该计算机设备可以获取该多个空中位置，将该多个空中位置发送至终端，终端获取多个中间图像，基于该多个中间图像的播放顺序，播放该多个中间图像。

在一种可能的实施方式中，该计算机设备可以根据该跳跃轨迹，从该当前位置对应的跳跃起始时间开始，每隔目标时段，获取一次该第一虚拟对象的中间位置，直到该目标落地点对应的跳跃结束时间为止，停止获取，从而可以得到多个中间位置。该计算机设备根据该第一虚拟对象的对象显示数据以及每个中间位置，在每个中间图像中渲染显示位于对应中间位置的第一虚拟对象。

在一种可能的场景中，该目的地为目标虚拟对象的当前位置，该第一虚拟对象可以为AI对象，该目标虚拟对象可以为用户虚拟对象，当该计算机设备控制该第一虚拟对象跳跃至目标落地点时，该计算机设备还可以基于该AI对象的寻路攻击逻辑，显示该AI对象攻击该目标虚拟对象的过程。

为了更清晰介绍本申请实施例的过程，下面以图9所示的流程图，对上述步骤301-306进行介绍，以AI对象为例，计算机设备检测到该AI对象不可行走至用户虚拟对象时，在用户虚拟周围选取目标落地点，例如，位于用户虚拟对象的行走区域中的点，该计算机设备基于该AI对象的速度范围，基于上述步骤305中公式一，从速度范围中最小值开始代入上述公式一进行计算，如果失败，则再增大该最小值，继续代入上述公式一进行计算，直至确定出满足上述公式一的跳跃速度，控制该第一虚拟对象以该跳跃速度，跳跃至目标落地点，以对用户虚拟对象进行攻击。

本申请实施例中，由于计算机设备可以基于目的地的位置获取该目标落地点，确定跳跃轨迹并控制第一虚拟对象从当前位置跳跃至该目标落地点，无需按照事先配置的跳跃点进行跳跃，因此，省略了事先在不可行走区域配置大量链接桥的过程，节省了对对象跳跃控制的成本，提高了资源利用率。

应该理解的是，本申请各实施例中的各个步骤并不是必然按照步骤标号指示的顺序依次执行。除非本文中有明确的说明，这些步骤的执行并没有严格的顺序限制，这些步骤可以以其它的顺序执行。而且，各实施例中至少一部分步骤可以包括多个子步骤或者多个阶段，这些子步骤或者阶段并不必然是在同一时刻执行完成，而是可以在不同的时刻执行，这些子步骤或者阶段的执行顺序也不必然是依次进行，而是可以与其它步骤或者其它步骤的子步骤或者阶段的至少一部分轮流或者交替地执行。

图 10 是本申请实施例提供的一种对象跳跃控制装置的结构示意图。参见图 10, 该装置包括:

确定模块 1001, 用于在虚拟场景中第一虚拟对象向目的地移动过程中, 当该第一虚拟对象的当前位置满足跳跃条件时, 根据该目的地的位置, 确定该第一虚拟对象的目标落地点;

该确定模块 1001, 还用于根据该当前位置和该目标落地点, 确定该第一虚拟对象从该当前位置跳跃至该目标落地点的跳跃轨迹;

跳跃控制模块 1002, 用于根据该跳跃轨迹, 控制该第一虚拟对象从该当前位置向该目标落地点跳跃。

在一种可能实现方式中, 该确定模块 1001, 还用于以下任一项:

当该当前位置满足该跳跃条件时, 基于该目的地的行走区域, 确定位于该行走区域内的目标落地点, 该行走区域内任一位置与该目的地之间包括行走路线;

当该当前位置满足该跳跃条件时, 将该虚拟场景中离该目的地最近的位置确定为该目标落地点;

当该当前位置满足该跳跃条件时, 将该虚拟场景中该目的地的相邻位置确定为该目标落地点;

当该当前位置满足该跳跃条件时, 将该目的地确定为该目标落地点。

在一种可能实现方式中, 该确定模块 1001, 还用于以下任一项:

根据该第一虚拟对象的最大跳跃速度, 获取该最大跳跃速度在该行走区域内对应的目标落地点;

获取该行走区域内被选中的位置, 将该被选中的位置确定为该目标落地点。

在一种可能实现方式中, 该目的地是指该虚拟场景中目标虚拟对象的当前位置, 或者, 该虚拟场景中被选中的位置。

在一种可能实现方式中, 该跳跃条件包括: 该当前位置与该目的地之间不包括行走路线, 或者, 在该当前位置与该目的地之间的最短路线方向上, 该当前位置前方包括障碍物。

在一种可能实现方式中, 该确定模块 1001, 还用于根据该当前位置和该目标落地点, 确定该第一虚拟对象在该当前位置的跳跃速度; 根据该跳跃速度, 确定该第一虚拟对象从该当前位置到该目标落地点的跳跃轨迹。

在一种可能实现方式中, 该装置还包括:

获取模块，用于当该当前位置与该目的地之间包括跳跃阻挡物时，对该第一虚拟对象进行移动，获取该第一虚拟对象移动后的当前位置，该第一虚拟对象移动后与该目标落地点的距离大于该第一虚拟对象移动前与该目标落地点的距离。

在一种可能实现方式中，该跳跃控制模块 1002，还用于根据该跳跃轨迹，获取该第一虚拟对象在离开该当前位置之后、到达该目标落地点之前的多个空中位置；基于该多个空中位置，获取该第一虚拟对象的多个中间图像，每个中间图像用于显示位于一个空中位置的第一虚拟对象；基于该多个中间图像的播放顺序，播放该多个中间图像。

本申请实施例中，由于计算机设备可以基于目的地的位置获取该目标落地点，确定跳跃轨迹并控制第一虚拟对象从当前位置跳跃至该目标落地点，无需按照事先配置的跳跃点进行跳跃，因此，省略了事先在不可行走区域配置大量链接桥的过程，节省了对对象跳跃控制的成本，提高了资源利用率。

上述所有可选技术方案，可以采用任意结合形成本公开的可选实施例，在此不再一一赘述。

需要说明的是：上述实施例提供的对象跳跃控制装置在控制对象跳跃时，仅以上述各功能模块的划分进行举例说明，实际应用中，可以根据需要而将上述功能分配由不同的功能模块完成，即将计算机设备的内部结构划分成不同的功能模块，以完成以上描述的全部或者部分功能。另外，上述实施例提供的对象跳跃控制装置与对象跳跃控制方法实施例属于同一构思，其具体实现过程详见方法实施例，这里不再赘述。

图 11 是本申请实施例提供的一种终端的结构示意图。该终端 1100 可以是：智能手机、平板电脑、MP3 播放器（Moving Picture Experts Group Audio Layer III，动态影像专家压缩标准音频层面 3）、MP4（Moving Picture Experts Group Audio Layer IV，动态影像专家压缩标准音频层面 4）播放器、笔记本电脑或台式电脑。终端 1100 还可能被称为用户设备、便携式终端、膝上型终端、台式终端等其他名称。

通常，终端 1100 包括有：处理器 1101 和存储器 1102。

处理器 1101 可以包括一个或多个处理核心，比如 4 核心处理器、8 核心处理器等。处理器 1101 可以采用 DSP（Digital Signal Processing，数字信号处理）、FPGA（Field - Programmable Gate Array，现场可编程门阵列）、PLA

(Programmable Logic Array, 可编程逻辑阵列) 中的至少一种硬件形式来实现。处理器 1101 也可以包括主处理器和协处理器, 主处理器是用于对在唤醒状态下的数据进行处理的处理单元, 也称 CPU (Central Processing Unit, 中央处理器); 协处理器是用于对在待机状态下的数据进行处理的低功耗处理器。在一些实施例中, 处理器 1101 可以在集成有 GPU (Graphics Processing Unit, 图像处理器), GPU 用于负责显示屏所需要显示的内容的渲染和绘制。一些实施例中, 处理器 1101 还可以包括 AI (Artificial Intelligence, 人工智能) 处理器, 该 AI 处理器用于处理有关机器学习的计算操作。

存储器 1102 可以包括一个或多个计算机可读存储介质, 该计算机可读存储介质可以是非暂态的。存储器 1102 还可包括高速随机存取存储器, 以及非易失性存储器, 比如一个或多个磁盘存储设备、闪存存储设备。在一些实施例中, 存储器 1102 中的非暂态的计算机可读存储介质用于存储至少一个指令, 该至少一个指令用于被处理器 1101 所执行以实现本申请中方法实施例提供的对象跳跃控制方法。

在一些实施例中, 终端 1100 还可选包括有: 外围设备接口 1103 和至少一个外围设备。处理器 1101、存储器 1102 和外围设备接口 1103 之间可以通过总线或信号线相连。各个外围设备可以通过总线、信号线或电路板与外围设备接口 1103 相连。具体地, 外围设备包括: 射频电路 1104、触摸显示屏 1105、摄像头 1106、音频电路 1107、定位组件 1108 和电源 1109 中的至少一种。

外围设备接口 1103 可被用于将 I/O (Input /Output, 输入/输出) 相关的至少一个外围设备连接到处理器 1101 和存储器 1102。在一些实施例中, 处理器 1101、存储器 1102 和外围设备接口 1103 被集成在同一芯片或电路板上; 在一些其他实施例中, 处理器 1101、存储器 1102 和外围设备接口 1103 中的任意一个或两个可以在单独的芯片或电路板上实现, 本实施例对此不加以限定。

射频电路 1104 用于接收和发射 RF (Radio Frequency, 射频) 信号, 也称电磁信号。射频电路 1104 通过电磁信号与通信网络以及其他通信设备进行通信。射频电路 1104 将电信号转换为电磁信号进行发送, 或者, 将接收到的电磁信号转换为电信号。可选地, 射频电路 1104 包括: 天线系统、RF 收发器、一个或多个放大器、调谐器、振荡器、数字信号处理器、编解码芯片组、用户身份模块卡等等。射频电路 1104 可以通过至少一种无线通信协议来与其它终端进行通信。该无线通信协议包括但不限于: 城域网、各代移动通信网络 (2G、3G、4G

及 5G)、无线局域网和/或 WiFi(Wireless Fidelity, 无线保真)网络。在一些实施例中,射频电路 1104 还可以包括 NFC(Near Field Communication, 近距离无线通信)有关的电路,本申请对此不加以限定。

显示屏 1105 用于显示 UI(User Interface, 用户界面)。该 UI 可以包括图形、文本、图标、视频及其它们的任意组合。当显示屏 1105 是触摸显示屏时,显示屏 1105 还具有采集在显示屏 1105 的表面或表面上方的触摸信号的能力。该触摸信号可以作为控制信号输入至处理器 1101 进行处理。此时,显示屏 1105 还可以用于提供虚拟按钮和/或虚拟键盘,也称软按钮和/或软键盘。在一些实施例中,显示屏 1105 可以为一个,设置终端 1100 的前面板;在另一些实施例中,显示屏 1105 可以为至少两个,分别设置在终端 1100 的不同表面或呈折叠设计;在再一些实施例中,显示屏 1105 可以是柔性显示屏,设置在终端 1100 的弯曲表面上或折叠面上。甚至,显示屏 1105 还可以设置成非矩形的不规则图形,也即异形屏。显示屏 1105 可以采用 LCD(Liquid Crystal Display, 液晶显示屏)、OLED(Organic Light-Emitting Diode, 有机发光二极管)等材质制备。

摄像头组件 1106 用于采集图像或视频。可选地,摄像头组件 1106 包括前置摄像头和后置摄像头。通常,前置摄像头设置在终端的前面板,后置摄像头设置在终端的背面。在一些实施例中,后置摄像头为至少两个,分别为主摄像头、景深摄像头、广角摄像头、长焦摄像头中的任意一种,以实现主摄像头和景深摄像头融合实现背景虚化功能、主摄像头和广角摄像头融合实现全景拍摄以及 VR(Virtual Reality, 虚拟现实)拍摄功能或者其它融合拍摄功能。在一些实施例中,摄像头组件 1106 还可以包括闪光灯。闪光灯可以是单色温闪光灯,也可以是双色温闪光灯。双色温闪光灯是指暖光闪光灯和冷光闪光灯的组合,可以用于不同色温下的光线补偿。

音频电路 1107 可以包括麦克风和扬声器。麦克风用于采集用户及环境的声波,并将声波转换为电信号输入至处理器 1101 进行处理,或者输入至射频电路 1104 以实现语音通信。出于立体声采集或降噪的目的,麦克风可以为多个,分别设置在终端 1100 的不同部位。麦克风还可以是阵列麦克风或全向采集型麦克风。扬声器则用于将来自处理器 1101 或射频电路 1104 的电信号转换为声波。扬声器可以是传统的薄膜扬声器,也可以是压电陶瓷扬声器。当扬声器是压电陶瓷扬声器时,不仅可以将电信号转换为人类可听见的声波,也可以将电信号转换为人类听不见的声波以进行测距等用途。在一些实施例中,音频电路 1107

还可以包括耳机插孔。

定位组件 1108 用于定位终端 1100 的当前地理位置，以实现导航或 LBS（Location Based Service，基于位置的服务）。定位组件 1108 可以是基于美国的 GPS（Global Positioning System，全球定位系统）、中国的北斗系统、俄罗斯的格雷纳斯系统或欧盟的伽利略系统的定位组件。

电源 1109 用于为终端 1100 中的各个组件进行供电。电源 1109 可以是交流电、直流电、一次性电池或可充电电池。当电源 1109 包括可充电电池时，该可充电电池可以支持有线充电或无线充电。该可充电电池还可以用于支持快充技术。

在一些实施例中，终端 1100 还包括有一个或多个传感器 1110。该一个或多个传感器 1110 包括但不限于：加速度传感器 1111、陀螺仪传感器 1112、压力传感器 1113、指纹传感器 1114、光学传感器 1115 以及接近传感器 1116。

加速度传感器 1111 可以检测以终端 1100 建立的坐标系的三个坐标轴上的加速度大小。比如，加速度传感器 1111 可以用于检测重力加速度在三个坐标轴上的分量。处理器 1101 可以根据加速度传感器 1111 采集的重力加速度信号，控制触摸显示屏 1105 以横向视图或纵向视图进行用户界面的显示。加速度传感器 1111 还可以用于游戏或者用户的运动数据的采集。

陀螺仪传感器 1112 可以检测终端 1100 的机体方向及转动角度，陀螺仪传感器 1112 可以与加速度传感器 1111 协同采集用户对终端 1100 的 3D 动作。处理器 1101 根据陀螺仪传感器 1112 采集的数据，可以实现如下功能：动作感应（比如根据用户的倾斜操作来改变 UI）、拍摄时的图像稳定、游戏控制以及惯性导航。

压力传感器 1113 可以设置在终端 1100 的侧边框和/或触摸显示屏 1105 的下层。当压力传感器 1113 设置在终端 1100 的侧边框时，可以检测用户对终端 1100 的握持信号，由处理器 1101 根据压力传感器 1113 采集的握持信号进行左右手识别或快捷操作。当压力传感器 1113 设置在触摸显示屏 1105 的下层时，由处理器 1101 根据用户对触摸显示屏 1105 的压力操作，实现对 UI 界面上的可操作性控件进行控制。可操作性控件包括按钮控件、滚动条控件、图标控件、菜单控件中的至少一种。

指纹传感器 1114 用于采集用户的指纹，由处理器 1101 根据指纹传感器 1114 采集到的指纹识别用户的身份，或者，由指纹传感器 1114 根据采集到的指纹识

别用户的身份。在识别出用户的身份为可信身份时，由处理器 1101 授权该用户执行相关的敏感操作，该敏感操作包括解锁屏幕、查看加密信息、下载软件、支付及更改设置等。指纹传感器 1114 可以被设置终端 1100 的正面、背面或侧面。当终端 1100 上设置有物理按键或厂商 Logo 时，指纹传感器 1114 可以与物理按键或厂商 Logo 集成在一起。

光学传感器 1115 用于采集环境光强度。在一个实施例中，处理器 1101 可以根据光学传感器 1115 采集的环境光强度，控制触摸显示屏 1105 的显示亮度。具体地，当环境光强度较高时，调高触摸显示屏 1105 的显示亮度；当环境光强度较低时，调低触摸显示屏 1105 的显示亮度。在另一个实施例中，处理器 1101 还可以根据光学传感器 1115 采集的环境光强度，动态调整摄像头组件 1106 的拍摄参数。

接近传感器 1116，也称距离传感器，通常设置在终端 1100 的前面板。接近传感器 1116 用于采集用户与终端 1100 的正面之间的距离。在一个实施例中，当接近传感器 1116 检测到用户与终端 1100 的正面之间的距离逐渐变小时，由处理器 1101 控制触摸显示屏 1105 从亮屏状态切换为息屏状态；当接近传感器 1116 检测到用户与终端 1100 的正面之间的距离逐渐变大时，由处理器 1101 控制触摸显示屏 1105 从息屏状态切换为亮屏状态。

本领域技术人员可以理解，图 11 中示出的结构并不构成对终端 1100 的限制，可以包括比图示更多或更少的组件，或者组合某些组件，或者采用不同的组件布置。

图 12 是本申请实施例提供的一种服务器的结构示意图，该服务器 1200 可因配置或性能不同而产生比较大的差异，可以包括一个或一个以上处理器（central processing units, CPU）1201 和一个或一个以上的存储器 1202，其中，该存储器 1202 中存储有至少一条指令，该至少一条指令由该处理器 1201 加载并执行以实现上述各个方法实施例提供的对象跳跃控制方法。当然，该服务器还可以具有有线或无线网络接口、键盘以及输入输出接口等部件，以便进行输入输出，该服务器还可以包括其他用于实现设备功能的部件，在此不做赘述。

在示例性实施例中，还提供了一种计算机可读存储介质，例如包括指令的存储器，上述指令可由计算机设备中的处理器执行以完成上述实施例中的对象跳跃控制方法。例如，该计算机可读存储介质可以是 ROM（Read-Only Memory，

只读存储器)、RAM (random access memory, 随机存取存储器)、CD-ROM (Compact Disc Read-Only Memory, 只读光盘)、磁带、软盘和光数据存储设备等。

本领域普通技术人员可以理解实现上述实施例的全部或部分步骤可以通过硬件来完成,也可以通过程序来指令相关的硬件完成,该的程序可以存储于一种计算机可读存储介质中,上述提到的存储介质可以是只读存储器,磁盘或光盘等。

以上所述仅为本申请的较佳实施例,并不用以限制本申请,凡在本申请的精神和原则之内,所作的任何修改、等同替换、改进等,均应包含在本申请的保护范围之内。

权利要求书

1、一种对象跳跃控制方法，由计算机设备执行，包括：

在虚拟场景中第一虚拟对象向目的地移动过程中，当所述第一虚拟对象的当前位置满足跳跃条件时，根据所述目的地的位置，确定所述第一虚拟对象的目标落地点；

根据所述当前位置和所述目标落地点，确定所述第一虚拟对象从所述当前位置跳跃至所述目标落地点的跳跃轨迹；及

根据所述跳跃轨迹，控制所述第一虚拟对象从所述当前位置向所述目标落地点跳跃。

2、根据权利要求1所述的方法，其特征在于，所述当所述第一虚拟对象的当前位置满足跳跃条件时，根据所述目的地的位置，确定所述第一虚拟对象的目标落地点包括以下任一项：

当所述当前位置满足所述跳跃条件时，基于所述目的地的行走区域，确定位于所述行走区域内的目标落地点，所述行走区域内任一位置与所述目的地之间包括行走路线；

当所述当前位置满足所述跳跃条件时，将所述虚拟场景中离所述目的地最近的位置确定为所述目标落地点；

当所述当前位置满足所述跳跃条件时，将所述虚拟场景中所述目的地的相邻位置确定为所述目标落地点；或者

当所述当前位置满足所述跳跃条件时，将所述目的地确定为所述目标落地点。

3、根据权利要求2所述的方法，其特征在于，所述基于所述目的地的行走区域，确定位于所述行走区域内的目标落地点包括以下任一项：

根据所述第一虚拟对象的最大跳跃速度，获取所述最大跳跃速度在所述行走区域内对应的目标落地点；或者

获取所述行走区域内被选中的位置，将所述被选中的位置确定为所述目标落地点。

4、根据权利要求1所述的方法，其特征在于，所述目的地是指所述虚拟场景中目标虚拟对象的当前位置，或者，所述虚拟场景中被选中的位置。

5、根据权利要求1所述的方法，其特征在于，所述跳跃条件包括：所述当前位置与所述目的地之间不包括行走路线，或者，在所述当前位置与所述目的

地之间的最短路线方向上、所述当前位置前方包括障碍物。

6、根据权利要求1所述的方法，其特征在于，所述根据所述当前位置和所述目标落地点，确定所述第一虚拟对象从所述当前位置跳跃至所述目标落地点的跳跃轨迹包括：

根据所述当前位置和所述目标落地点，确定所述第一虚拟对象在所述当前位置的跳跃速度；及

根据所述跳跃速度，确定所述第一虚拟对象从所述当前位置到所述目标落地点的跳跃轨迹。

7、根据权利要求1所述的方法，其特征在于，所述根据所述当前位置和所述目标落地点，确定所述第一虚拟对象从所述当前位置跳跃至所述目标落地点的跳跃轨迹之前，所述方法还包括：

当所述当前位置与所述目的地之间包括跳跃阻挡物时，对所述第一虚拟对象进行移动，获取所述第一虚拟对象移动后的当前位置，所述第一虚拟对象移动后与所述目标落地点的距离大于所述第一虚拟对象移动前与所述目标落地点的距离。

8、根据权利要求1所述的方法，其特征在于，所述根据所述跳跃轨迹，控制所述第一虚拟对象从所述当前位置向所述目标落地点跳跃包括：

根据所述跳跃轨迹，获取所述第一虚拟对象在离开所述当前位置之后、到达所述目标落地点之前的多个空中位置；

基于所述多个空中位置，获取所述第一虚拟对象的多个中间图像，每个中间图像用于显示位于一个空中位置的第一虚拟对象；及

基于所述多个中间图像的播放顺序，播放所述多个中间图像。

9、一种对象跳跃控制装置，其特征在于，所述装置包括：

确定模块，用于在虚拟场景中第一虚拟对象向目的地移动过程中，当所述第一虚拟对象的当前位置满足跳跃条件时，根据所述目的地的位置，确定所述第一虚拟对象的目标落地点；

所述确定模块，还用于根据所述当前位置和所述目标落地点，确定所述第一虚拟对象从所述当前位置跳跃至所述目标落地点的跳跃轨迹；

跳跃控制模块，用于根据所述跳跃轨迹，控制所述第一虚拟对象从所述当前位置向所述目标落地点跳跃。

10、一种计算机设备，包括存储器和处理器，所述存储器中存储有计算机

可读指令，所述计算机可读指令被所述处理器执行时，使得所述处理器执行权利要求 1 至 8 中任一项所述方法的步骤。

11、一个或多个存储有计算机可读指令的非易失性存储介质，所述计算机可读指令被一个或多个处理器执行时，使得一个或多个处理器执行权利要求 1 至 8 中任一项所述方法的步骤。