

虚拟对象的控制方法、装置、设备及介质

本申请要求于2020年05月15日提交的申请号为2020104120065、发明名称为“虚拟对象的控制方法、装置、设备及介质”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

本申请涉及计算机技术领域，特别涉及一种虚拟对象的控制方法、装置、设备及介质。

背景技术

随着计算机技术的发展以及终端功能的多样化，在终端上能够进行的游戏种类越来越多。其中，MOBA（Multiplayer Online Battle Arena，多人在线战术竞技）游戏是一种比较盛行的游戏，终端可以在界面中显示虚拟场景，并在虚拟场景中显示虚拟对象，该虚拟对象可以通过释放技能，与其他虚拟对象进行对战。

一般地，虚拟场景的显示是以当前控制的第一虚拟对象为中心的。目前，虚拟对象的控制方法通常是检测到对技能的释放操作时，在以当前控制的第一虚拟对象为中心的虚拟场景中，根据释放操作的操作位置，来确定该技能的释放目标，从而控制第一虚拟对象进行技能释放，该释放目标为虚拟场景中的一个位置、虚拟对象或方向。

上述控制方法中，对技能进行释放操作时，仅能够在以第一虚拟对象为中心的虚拟场景内选择释放目标，如果用户想影响的对象并未显示于虚拟场景中，则需要进行粗略估计，来控制技能释放，这样上述控制方法的精度和准确性较低。

发明内容

本申请实施例提供了一种虚拟对象的控制方法、装置、设备及介质，能够提高控制方法的精度和准确性。该技术方案如下：

一方面，提供了一种虚拟对象的控制方法，该方法包括：

显示第一虚拟场景，所述第一虚拟场景中显示有地图控件；

响应于对所述地图控件的第一触发操作，显示第一操作位置对应的第二虚拟场景，所述第一触发操作作用于所述第一操作位置；

响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置；

控制第一虚拟对象按照所述技能释放目标释放所述目标技能

一方面，提供了一种虚拟对象的控制装置，该装置包括：

显示模块，用于显示虚拟场景，所述虚拟场景中显示有地图控件，响应于对所述地图控件的第一触发操作，显示第一操作位置对应的第二虚拟场景，所述第一触发操作作用于所述第一操作位置；

确定模块，用于响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置；

控制模块，用于控制第一虚拟对象按照所述技能释放目标释放所述目标技能。

一方面，提供了一种电子设备，该电子设备包括一个或多个处理器和一个或多个存储器，该一个或多个存储器中存储有至少一条程序代码，该至少一条程序代码由该一个或多个处理器加载并执行以实现如上述任一种可能实现方式的虚拟对象的控制方法所执行的操作。

一方面，提供了一种存储介质，该存储介质中存储有至少一条程序代码，该至少一条程序代码由处理器加载并执行以实现如上述任一种可能实现方式的虚拟对象的控制方法所执行的操作。

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终端界面示意图；

图2是本申请实施例提供的一种终端界面示意图；

图3是本申请实施例提供的一种终端界面示意图；

图4是本申请实施例提供的一种终端界面示意图；

图5是本申请实施例提供的一种虚拟对象的控制方法的实施环境示意图；

图6是本申请实施例提供的一种虚拟对象的控制方法的流程图；

图7是本申请实施例提供的一种虚拟对象的控制方法的流程图；

图8是本申请实施例提供的一种小地图与虚拟场景的对应关系的示意图；

图9是本申请实施例提供的一种终端界面示意图；

图10是本申请实施例提供的一种镜头位置与角色位置的关系示意图；

图11是本申请实施例提供的一种终端界面示意图；

图12是本申请实施例提供的一种终端界面示意图；

图13是本申请实施例提供的一种终端界面示意图；

图14是本申请实施例提供的一种虚拟场景与虚拟摄像头的位置关系示意图；

图15是本申请实施例提供的一种操作区域与虚拟场景的映射关系图；

图16是本申请实施例提供的一种终端界面示意图；

图17是本申请实施例提供的一种操作区域与虚拟场景的映射关系图；

图18是本申请实施例提供的一种虚拟对象的控制方法的流程图；

图19是本申请实施例提供的一种终端界面示意图；

图20是本申请实施例提供的一种虚拟对象的控制装置的结构示意图；

图21是本申请实施例提供的一种终端2100的结构示意图。

具体实施方式

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

本申请中术语“第一”“第二”等字样用于对作用和功能基本相同的相同项或相似项进行区分，应理解，“第一”、“第二”、“第n”之间不具有逻辑或时序上的依赖关系，也不对数量和执行顺序进行限定。

本申请中术语“至少一个”是指一个或一个以上，“至少两个”是指两个或两个以上，例如，

至少两个节点设备是指两个或两个以上的节点设备。

以下，对本申请涉及到的术语进行解释。

虚拟场景：是应用程序在终端上运行时显示（或提供）的虚拟场景。该虚拟场景是对真实世界的仿真环境，或，半仿真半虚构的虚拟环境，或，是纯虚构的虚拟环境。虚拟场景可以是二维虚拟场景、2.5 维虚拟场景或者三维虚拟场景中的任意一种，本申请实施例对虚拟场景的维度不加以限定。例如，虚拟场景包括天空、陆地、海洋等，该陆地包括沙漠、城市等环境元素，用户能够控制虚拟对象在该虚拟场景中进行移动。可选地，该虚拟场景用于至少两个虚拟对象之间的虚拟场景对战，在该虚拟场景中具有可供至少两个虚拟对象使用的虚拟资源。可选地，该虚拟场景中包括对称的两个区域，属于两个敌对阵营的虚拟对象分别占据其中一个区域，并以摧毁对方区域深处的目标建筑/据点/基地/水晶来作为胜利目标，其中，对称的区域比如左下角区域和右上角区域，又比如左侧中部区域和右侧中部区域等。

虚拟对象：是指在虚拟场景中的可活动对象。该可活动对象是虚拟人物、虚拟动物或动漫人物等，比如：在虚拟场景中显示的人物、动物、植物、油桶、墙壁、石块等。该虚拟对象是该虚拟场景中的一个虚拟的、用于代表用户的虚拟形象。在虚拟场景中可以包括多个虚拟对象，每个虚拟对象在虚拟场景中具有自身的形状和体积，占据虚拟场景中的一部分空间。可选地，当虚拟场景为三维虚拟场景时，可选地，虚拟对象是一个三维立体模型，该三维立体模型是基于三维人体骨骼技术构建的三维角色，同一个虚拟对象通过穿戴不同的皮肤来展示出不同的外在形象。在一些实施例中，虚拟对象采用 2.5 维或 2 维模型来实现，本申请实施例对此不加以限定。

可选地，该虚拟对象是通过客户端上的操作进行控制的玩家角色，或，通过训练设置在虚拟场景对战中的人工智能（Artificial Intelligence, AI），或，设置在虚拟场景互动中的非玩家角色（Non-Player Character, NPC）。可选地，该虚拟对象是在虚拟场景中进行竞技的虚拟人物。可选地，该虚拟场景中参与互动的虚拟对象的数量是预先设置的，或者，是根据加入互动的客户端的数量动态确定的。

MOBA（Multiplayer Online Battle Arena，多人在线战术竞技）游戏：是一种在虚拟场景中提供若干个据点，处于不同阵营的用户控制虚拟对象在虚拟场景中对战，占领据点或摧毁敌对阵营据点的游戏。例如，MOBA 游戏可将用户分成至少两个敌对阵营，分属至少两个敌对阵营的不同虚拟队伍分别占据各自的地图区域，以某一种胜利条件作为目标进行竞技。该胜利条件包括但不限于：占领据点或摧毁敌对阵营据点、击杀敌对阵营的虚拟对象、在指定场景和时间内保证自身的存活、抢夺到某种资源、在指定时间内互动比分超过对方中的至少一种。例如，MOBA 游戏可将用户分成两个敌对阵营，将用户控制的虚拟对象分散在虚拟场景中互相竞争，以摧毁或占领敌方的全部据点作为胜利条件。

可选地，每个虚拟队伍包括一个或多个虚拟对象，比如 1 个、2 个、3 个或 5 个，根据参与战术竞技中各队伍内虚拟对象的数量，可以将战术竞技划分为 1V1 竞技比拼、2V2 竞技比拼、3V3 竞技比拼、5V5 竞技比拼等，其中，1V1 是指“1 对 1”的意思，这里不做赘述。

可选地，MOBA 游戏可以以局（或称为回合）为单位来进行，每局战术竞技的地图相同或者不同。一局 MOBA 游戏的持续时间是从游戏开始的时刻至达成胜利条件的时刻。

在 MOBA 游戏中，用户能够控制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、弯腰前行等，也能够控制虚拟对象在

海洋中游泳、漂浮或者下潜等，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。

在 MOBA 游戏中，用户还能够控制虚拟对象释放技能从而与其他虚拟对象进行战斗，例如，该技能的技能类型可以包括攻击技能、防御技能、治疗技能、辅助技能、斩杀技能等，每个虚拟对象都可以具有各自固定的一个或多个技能，而不同的虚拟对象通常具有不同的技能，不同的技能可以产生不同的作用效果。比如，若虚拟对象释放攻击技能击中了敌对虚拟对象，那么会对敌对虚拟对象造成一定的伤害，通常表现为扣除敌对虚拟对象的一部分虚拟生命值，又比如，若虚拟对象释放治疗技能命中了友方虚拟对象，那么会对友方虚拟对象产生一定的治疗效果，通常表现为恢复友方虚拟对象的一部分虚拟生命值，其他各类技能均可以产生相应的作用效果，这里不再一一枚举。

在本申请实施例中，提供了两种技能释放方式，不同的技能释放方式对应不同的操作方式，用户可以根据自身使用习惯，任意选择或切换技能释放方式进行技能释放，来满足自身需求，大大提高了技能释放的准确性。

该两种技能释放方式可以分别为：主动释放和快速释放。其中，该主动释放是指通过用户操作来确定技能的释放目标，快速释放是指由终端自动确定技能的释放目标。

在一些实施例中，为两种技能释放方式设置对应的操作区域，主动释放对应的操作区域为第一操作区域，快速释放对应的操作区域为第二操作区域。该第一操作区域包围于该第二操作区域外。

在一些实施例中，终端根据技能的释放操作结束时的操作位置与操作区域之间的关系，来确定为哪一种技能释放方式。例如，释放操作结束时操作位置位于第一操作区域内，则为主动释放；如果位于第二操作区域内，则为快速释放。通过快速释放，无需用户操作选择释放目标，大大简化了用户操作，减小了操作复杂度，提供了一种便利的操作方式。通过主动释放，用户可以自由选择释放目标，能够更精确地，提高了用户操作的技巧性，更符合高端玩家的操作需求，提高了用户体验。

下面针对技能的释放相关内容进行简单介绍。

技能释放可以通过对技能控件进行操作实现，包括该技能控件在内的一个区域可以为技能轮盘，上述技能的释放方式即可通过对技能轮盘进行操作实现。在一些实施例中，该第二操作区域可以为该技能控件所在区域，或者与该技能控件的中心位置的距离小于距离阈值的区域，该第一操作区域可以为第二操作区域外的一个区域。技能轮盘即为上述第一操作区域和第二操作区域共同组成的区域。

例如，如图 1 所示，该虚拟对象可以具备多个技能：技能 1、技能 2、技能 3、技能 4，在对技能 3 进行释放操作时，可以显示技能轮盘 101，该技能轮盘 101 中可以包括第一操作区域 102 和第二操作区域 103，其中，该第二操作区域中显示有技能 3 的技能控件，在进行拖动操作时，通过控制技能摇杆 104 在技能轮盘中移动，来实现操作位置的改变。该技能摇杆 104 仅能够位于该技能轮盘 101 中。

以对技能的释放操作通过拖动技能摇杆来实现为例，用户能够对技能摇杆 104 进行拖动操作，如果未将技能摇杆 104 拖动出第一操作区域即结束了操作，则确定为快速释放，如果将技能摇杆 104 从第一操作区域拖动出来，进入到了第二操作区域并结束了操作，则可以确定为主动释放。也即是，若对该技能摇杆 104 的拖动操作的结束位置在该第一操作区域中，

则对技能进行快速释放,若对该技能摇杆 104 的拖动操作的结束位置在该第一操作区域以外,且位于第二操作区域,则对技能进行主动释放。

在一些实施例中,终端在用户图形界面中显示取消释放控件,该取消释放控件用于取消对该技能的释放。可选的,终端响应于对技能的触发操作结束,且该触发操作的结束位置位于该取消释放控件所在位置,取消对该技能的释放。通过取消释放控件,提供了一种取消技能释放的方式,丰富了技能释放操作,为用户提供了更多技能释放功能,提高了用户体验。例如,如图 1 所示,界面中可以显示有该取消释放控件 105。如果用户持续释放操作一直移动到该取消释放控件 105 处,则可以取消本次技能释放。

虚拟对象的技能包括不同类型的技能,例如,有些技能为目标型技能,有些技能为位置型技能,有些技能是方向型技能。例如,如图 2 所示,该技能为目标型技能,需要选定释放的目标虚拟对象。如图 3 所示,该技能为位置型技能,需要选定释放位置。如图 4 所示,该技能为方向型技能,需要选定释放方向。

以下,对本申请涉及的系统架构进行介绍。

图 5 是本申请实施例提供的一种虚拟对象的控制方法的实施环境示意图。参见图 5,该实施环境包括:第一终端 120、服务器 140 和第二终端 160。

第一终端 120 安装和运行有支持虚拟场景的应用程序。该应用程序可以是多人在线战术竞技游戏(Multiplayer Online Battle Arena games, MOBA)、虚拟现实应用程序、二维或三维地图程序、仿真程序中的任意一种。当然,该应用程序还可以为其他程序,例如,多人枪战类生存游戏,本申请实施例对此不作限定。第一终端 120 可以是第一用户使用的终端,第一用户使用第一终端 120 操作位于虚拟场景中的第一虚拟对象进行活动,该活动包括但不限于:步行、奔跑、调整身体姿态、普通攻击、释放技能中的至少一种。当然,该活动还可以包括其他项,例如,射击、投掷等,本申请实施例对此不作具体限定。示意性的,第一虚拟对象是第一虚拟人物,比如仿真人物角色或动漫人物角色。示意性的,第一虚拟对象可以是第一虚拟动物,比如仿真猴子或者其他动物等。

第一终端 120 以及第二终端 160 通过无线网络或有线网络与服务器 140 相连。

服务器 140 可以包括一台服务器、多台服务器、云计算平台或者虚拟化中心中的至少一种。服务器 140 用于为支持虚拟场景的应用程序提供后台服务。可选地,服务器 140 可以承担主要计算工作,第一终端 120 和第二终端 160 可以承担次要计算工作;或者,服务器 140 承担次要计算工作,第一终端 120 和第二终端 160 承担主要计算工作;或者,服务器 140、第一终端 120 和第二终端 160 三者之间采用分布式计算架构进行协同计算。

其中,服务器 140 可以是独立的物理服务器,也可以是多个物理服务器构成的服务器集群或者分布式系统,还可以是提供云服务、云数据库、云计算、云函数、云存储、网络服务、云通信、中间件服务、域名服务、安全服务、内容分发网络(Content Distribute Network, CDN)、以及大数据和人工智能平台等基础云计算服务的云服务器。第一终端 120 和第二终端 160 可以是智能手机、平板电脑、笔记本电脑、台式计算机、智能音箱、智能手表等,但并不局限于此。第一终端 120、第二终端 160 可以与服务器通过有线或无线通信方式进行直接或间接地连接,本申请实施例在此不做限制。

示意性的,第一终端 120 和第二终端 160 可以将产生的数据发送至服务器 140,服务器 140 可以对自身产生的数据与终端产生的数据进行校验,如果与任一终端的校验结果指示数

据不一致,则可以将服务器产生的数据发送至任一终端,该任一终端以服务器产生的数据为准。

在一些实施例中,第一终端120和第二终端160可以根据用户的触发操作,确定每一帧虚拟场景,并将虚拟场景发送至服务器140,也可以将用户的触发操作的信息发送至服务器140,服务器140可以接收该触发操作的信息和虚拟场景,根据该触发操作,确定虚拟场景,与终端上传的虚拟场景进行对比,如果一致,则可以继续后续计算;如果不一致,则可以将自身确定的虚拟场景发送至各个终端进行同步。在一个具体的可能实施例中,服务器140还可以根据该触发操作的信息,确定各个终端的下一帧虚拟场景,将下一帧虚拟场景发送至各个终端,使得各个终端执行相应的步骤,得到与服务器140确定的下一帧虚拟场景一致的虚拟场景。

第二终端160安装和运行有支持虚拟场景的应用程序。该应用程序可以是MOBA、虚拟现实应用程序、二维或三维地图程序、仿真程序中的任意一种。当然,该应用程序还可以为其他程序,例如,多人枪战类生存游戏,本申请实施例对此不作限定。第二终端160可以是第二用户使用的终端,第二用户使用第二终端160操作位于虚拟场景中的第二虚拟对象进行活动,该活动包括但不限于:步行、奔跑、调整身体姿态、普通攻击、释放技能中的至少一种。当然,该活动还可以包括其他项,例如,射击、投掷等,本申请实施例对此不作具体限定。示意性的,第二虚拟对象是第二虚拟人物,比如仿真人物角色或动漫人物角色。示意性的,第二虚拟对象可以是第二虚拟动物,比如仿真猴子或者其他动物等。

可选地,第一终端120控制的第一虚拟对象和第二终端160控制的第二虚拟对象处于同一虚拟场景中,此时第一虚拟对象可以在虚拟场景中第二虚拟对象进行互动。在一些实施例中,第一虚拟对象以及第二虚拟对象可以为敌对关系,例如,第一虚拟对象与第二虚拟对象可以属于不同的队伍、组织或阵营,敌对关系的虚拟对象之间,可以在虚拟场景中的任一位置通过释放技能进行对战方式的互动。

在另一些实施例中,该第二虚拟对象也可以与该第一虚拟对象为队友关系,例如,第一虚拟人物和第二虚拟人物可以属于同一个队伍、同一个组织、同一个阵营、具有好友关系或具有临时性的通讯权限。

可选地,第一终端120和第二终端160上安装的应用程序是相同的,或两个终端上安装的应用程序是不同操作系统平台的同一类型应用程序。第一终端120可以泛指多个终端中的一个,第二终端160可以泛指多个终端中的一个,本实施例仅以第一终端120和第二终端160来举例说明。第一终端120和第二终端160的设备类型相同或不同,该设备类型包括:智能手机、平板电脑、电子书阅读器、动态影像专家压缩标准音频层面3(Moving Picture Experts Group Audio Layer III, MP3)播放器、动态影像专家压缩标准音频层面4(Moving Picture Experts Group Audio Layer IV, MP4)播放器、膝上型便携计算机和台式计算机中的至少一种。例如,第一终端120和第二终端160可以是智能手机,或者其他手持便携式游戏设备。以下实施例,以终端包括智能手机来举例说明。

本领域技术人员可以知晓,上述终端的数量可以更多或更少。比如上述终端可以仅为一个,或者上述终端为几十个或几百个,或者更多数量。本申请实施例对终端的数量和设备类型不加以限定。

图6是本申请实施例提供的一种虚拟对象的控制方法的流程图,该方法应用于电子设备,

电子设备可以为终端，也可以为服务器，本申请实施例对此不作限定，本实施例中，以应用于终端为例进行说明。参见图6，该方法可以包括以下步骤。

601、终端显示第一虚拟场景，该第一虚拟场景中显示有地图控件，响应于对地图控件的第一触发操作，显示第一操作位置对应的第二虚拟场景，第一触发操作作用于所述第一操作位置。

在上述过程中，终端响应于对地图控件的第一触发操作，根据该第一触发操作的第一操作位置，将用户图形界面中显示的虚拟场景（也即是第一虚拟场景）切换至该第一操作位置对应的目标虚拟场景（也即是第二虚拟场景）。

其中，该地图控件用于显示虚拟场景的地图，可以通过对地图控件进行操作，来改变当前显示的虚拟场景。需要说明的是，如果未对地图控件进行操作，当前一般显示的虚拟场景为以当前控制的第一虚拟对象为中心的局部虚拟场景，也即是第一虚拟场景。如果对地图控件上的某个位置点进行了操作，则可以调整虚拟摄像机的位置，显示其他的局部虚拟场景。

其中，该第一触发操作作为点击操作或滑动操作，本申请实施例对此不作限定。例如，以第一触发操作作为点击操作为例，用户点击地图控件中某个位置，该位置即为第一操作位置，则第二虚拟场景为以该第一操作位置为中心的虚拟场景，或，该第二虚拟场景为该第一操作位置为起始点的虚拟场景，本申请实施例对此不作限定。以该第一触发操作作为拖动操作为例，用户可以在地图控件上滑动，这样可以根据滑动时的操作位置，来实时更新显示的虚拟场景，以便于对显示的虚拟场景的更细致、更精准的调整。

602、终端响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置。

在上述过程中，终端响应于对目标技能的释放操作，根据该释放操作的第二操作位置，确定该第二操作位置在该目标虚拟场景中对应的技能释放目标。

其中，该目标技能是指虚拟场景中虚拟对象的能力。从技能的释放效果角度来看，该目标技能可以为一种动作技能，也可以为一种属性改变技能，例如，某个虚拟对象可以拥有三项技能，其中，一项技能为向前方冲刺的动作技能，另一项技能为提升自身的移动速度的属性增益（buff）技能，再一项技能为减弱附近队友所受攻击的属性减益（Debuff）技能。从技能的释放类型来看，该目标技能可以为位置型技能、方向型技能或目标型技能中任一种。

在一些实施例中，该释放操作作为点击操作，或者，拖动操作，本申请实施例对此不作限定。对应于该两种技能释放方式，如果释放操作作为点击操作，则为快速释放，如果释放操作作为拖动操作，则可以根据释放操作结束时的操作位置，来确定为哪一种释放方式。

在本申请实施例中，通过上述步骤601，已经将当前显示的第一虚拟场景切换到通过地图控件选定的第二虚拟场景中，如果用户想要在该第二虚拟场景中某个位置进行技能释放，或者对第二虚拟场景中的某个虚拟对象进行技能释放，又或者在该第二虚拟场景中虚拟对象的位置，来确定技能释放方向，则可以进行对技能的释放操作。终端检测到该释放操作，可以响应于该释放操作，根据第二操作位置来确定技能释放目标。其中，该技能释放目标为技能释放位置、目标虚拟对象或技能释放方向中的任一项，也即是，该技能释放目标为该第二虚拟场景中的目标虚拟对象、位置或者位置与第一虚拟对象所构成的方向。

通过将用户图形界面的显示内容切换至第一操作位置对应的第二虚拟场景，这样技能释放目标的选择范围可以不局限于以虚拟对象为中心的虚拟场景中，释放操作的自由度更高，

且在技能释放时能够精准地根据想要释放位置的情况来进行选择，而非在当前显示的虚拟场景中粗略估计，提高了对虚拟对象的控制方法的精度和准确性。

603、终端控制第一虚拟对象按照该技能释放目标释放该目标技能。

终端确定了技能释放目标，即可控制向该按照该技能释放目标释放技能。在一些实施例中，该释放技能的过程还可以为：终端在用户图形界面中显示向该目标虚拟对象释放该技能时产生的释放效果，例如，该技能释放目标为目标虚拟对象，可以在第一虚拟对象和该目标虚拟对象之间显示技能的释放过程效果，并在该目标虚拟对象身上显示释放后的效果。又例如，该技能释放目标为释放位置，则可以在该释放位置处显示目标动画，体现释放效果，如果该释放位置处包括有第二虚拟对象，则可以显示该第二虚拟对象的属性值受到影响。又例如，该技能释放目标为释放方向，则可以沿着该释放方向显示技能的释放过程效果。

本申请实施例通过根据对地图控件的第一触发操作的第一操作位置，在用户图形界面中显示该第一操作位置对应的第二虚拟场景，在这种情况下，响应于对目标技能的释放操作，能够根据释放操作对应的操作位置，来确定该目标技能在当前显示的第二虚拟场景中对应的技能释放目标，从而进行技能释放，在上述通过地图控件来控制技能释放的方式中，能够在对地图控件进行第一触发操作时，显示相应的第二虚拟场景，这样技能释放目标的选择范围可以不局限于以虚拟对象为中心的虚拟场景中，释放操作的自由度更高，且在技能释放时能够精准地根据想要释放位置的情况来进行选择，而非在当前显示的虚拟场景中粗略估计，提高了对虚拟对象的控制方法的精度和准确性。

图7是本申请实施例提供的一种虚拟对象的控制方法的流程图，参见图7，该方法可以包括：

701、终端响应于对地图控件的第一触发操作，根据该第一触发操作的第一操作位置以及该地图控件中显示信息与虚拟场景的对应关系，获取该第一操作位置对应的第二虚拟场景。

用户如果想要通过地图控件，改变当前显示的虚拟场景，对地图控件进行第一触发操作，终端能够根据该第一触发操作的第一操作位置，来进行虚拟场景的切换，从而实现对虚拟场景的观察视角的调整，实现视野画面的调整。

在一些实施例中，该地图控件中显示有全局虚拟场景的简要信息，例如，显示有全局虚拟场景的缩略图。在一些实施例中，该地图控件中根据虚拟场景中部分或全部虚拟对象的位置，显示部分或全部虚拟对象的标识信息，例如，该标识信息为头像。

其中，该地图控件中显示信息与虚拟场景具有对应关系。例如，在一个具体示例中，该地图控件中显示的虚拟场景的缩略图为2D的信息，虚拟场景为一个3D的虚拟空间，该缩略图为将虚拟场景的俯瞰图按照一定比例缩小后的图像，或，包括缩小后的图像的部分重要信息的图像。

例如，如图8所示，对于该地图控件中显示信息与虚拟场景的对应关系，在此提供一个地图控件（也可以称其为小地图）与虚拟场景的俯瞰图（2D虚拟场景）之间的对应关系，在三维坐标中，可以忽略y轴，地图控件中显示信息的x轴与z轴分别与2D虚拟场景的x轴和z轴映射。

在一个具体示例中，假设全局虚拟场景是一个正方形，使用MapLength和SceneLength分别表示小地图的边长和场景的边长。其中，MimiMapStartPos表示小地图左下角，即小地图的起始位置，一般在初始化小地图的UI（User Interface，用户界面）时会设置该参数。

SceneStartPos 表示虚拟场景中的左下角，也即是虚拟场景的起始位置，一般在地图编辑时会设置该参数。将第一操作位置命名为拖拽位置（DragPos），我们可知 DragPos 在小地图（MiniMap）中的位置相当于目标镜头位置（AimCameraPos）在虚拟场景（Scene）中位置，也即是，可以通过下述公式一表示：

公式一： $(\text{DragPos} - \text{MiniMapStartPos}) / \text{MapLength} = (\text{AimCameraPos} - \text{SceneStartPos}) / \text{SceneLength}$

通过上述公式一，可以得到下述公式二，通过公式二即可计算得到小地图中位置 DragPos 在虚拟场景中的位置 AimCameraPos。

公式二： $\text{AimCameraPos} = (\text{DragPos} - \text{MiniMapStartPos}) * \text{SceneLength} / \text{MapLength} + \text{SceneStartPos}$

上述公式一和公式二中，MaxAimRadius 为技能按钮的瞄准的最大范围，AimCameraPos 为第二虚拟场景中屏幕中心点的场景位置，DragPos 为小地图中的拖拽位置，MiniMapStartPos 为小地图的起始位置，SceneLength 为场景长度，也即是场景的边长，MapLength 为小地图长度，也即是小地图的边长，SceneStartPos 为场景的起始位置。其中，* 表示乘法运算。

如果用户并未对地图控件进行操作，则将 AimCameraPos 赋值为 InvalidAimCameraPos，InvalidAimCameraPos 的意思是表示当前小地图没有被按下。

在一些实施例中，上述显示信息为某个位置或某个区域。相应地，该步骤 701 中确定第二虚拟场景的过程可以包括两种实现方式。

实现方式一、终端根据所述第一触发操作的第一操作位置，在所述地图控件中，确定以所述第一操作位置为中心、尺寸为第一目标尺寸的目标区域，根据所述地图控件中显示信息与虚拟场景的对应关系，确定所述目标区域在虚拟场景中对应的第二虚拟场景。

在上述实现方式一中，终端根据该第一触发操作的第一操作位置，获取该地图控件中，以该第一操作位置为中心、尺寸为第一目标尺寸的目标区域，根据该地图控件中的显示信息与虚拟场景的对应关系，获取该目标区域对应的目标虚拟场景，也即是第二虚拟场景。

其中，该目标区域可以为矩形区域或其他形状的区域，本申请实施例对此不作限定。例如，如图 9 所示，该目标区域 901 为该地图控件中第一操作位置为中心的区域。

实现方式二、终端根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置在所述虚拟场景中对应的目标位置，确定以所述目标位置为中心、尺寸为第二目标尺寸的第二虚拟场景。

终端根据该第一操作位置以及该地图控件中的显示信息与虚拟场景的对应关系，获取该第一操作位置在虚拟场景中对应的位置，获取以该对应的位置为中心、尺寸为第二目标尺寸的目标虚拟场景，也即是第二虚拟场景。

在该实现方式二中，用户在对地图控件进行第一触发操作时，该第一触发操作的第一操作位置即为第二虚拟场景的获取依据。用户通过改变该第一操作位置，可以改变第二虚拟场景。

在一些实施例中，该第二虚拟场景为以第一操作位置在虚拟场景中对应的位置为中心的虚拟场景。因而，终端能够参考上述对应关系，来确定第一操作位置在虚拟场景中对应的位置，从而分析到底第二虚拟场景以哪个位置为中心，再结合第二虚拟场景的显示视野（也即是尺寸），即可获取到该第二虚拟场景。

在一个具体示例中，终端根据地图控件中显示信息与虚拟场景的对应关系，将 2D 的第一操作位置转换为虚拟场景中的 3D 位置。一般地，该显示虚拟场景的过程通常是通过虚拟

摄像机观察实现，来模拟通过某个摄像机观察某个真实环境时的观察视野。为了达到更好的3D效果，该虚拟摄像机在虚拟场景的地面上方某个高度，通过某个倾斜的视角来观察虚拟场景，因而，终端能够根据该第一操作位置在虚拟场景中的对应位置、虚拟摄像机的高度以及目标角度，来获取虚拟摄像机的位置，通过该虚拟摄像机的位置，从全局虚拟场景中获取该第二虚拟场景。

例如，如图10所示，可以通过上述方式，确定出第一操作位置DragPos对应虚拟场景中的位置AimCameraPos，进而将AimCameraPos赋值给ActorPos，用ActorPos计算出虚拟摄像机（也可以称之为镜头）的位置。在一些实施例中，终端可以判断是否有对地图控件的第一触发操作，也即是，判断AimCameraPos是否为InvalidAimCameraPos，如果是，则镜头跟随第一虚拟对象，可以将第一虚拟对象的位置赋值给ActorPos。如果否，则镜头跟随小地图拖动的镜头位置，也即是，可以获取AimCameraPos，将AimCameraPos赋值给ActorPos。

其中，通过ActorPos获取虚拟摄像机的位置时，可以通过下述公式三至公式五实现。

公式三： $\text{cameraPos.x} = \text{ActorPos.x}$,

公式四： $\text{cameraPos.y} = \text{ActorPos.y} + \text{height} * \cos(\text{angle})$,

公式五： $\text{cameraPos.z} = \text{ActorPos.z} - \text{height} * \sin(\text{angle})$,

其中，cameraPos.x、cameraPos.y、cameraPos.z分别为虚拟摄像机的x、y、z轴的坐标，ActorPos.x、ActorPos.y、ActorPos.z分别为ActorPos的x、y、z轴的坐标，height为虚拟摄像机的高度，angle为虚拟摄像机的倾斜角度。cos()为余弦函数，sin()为正弦函数。

702、终端将用户图形界面中显示的第一虚拟场景切换至该第二虚拟场景。

终端获取到第二虚拟场景后，在用户图形界面中显示该第二虚拟场景，这样将视野调整到合适的时候，用户能够更精准的进行技能的释放操作。

上述步骤701和步骤702为响应于对地图控件的第一触发操作，根据该第一触发操作的第一操作位置，将用户图形界面中显示的第一虚拟场景切换至该第一操作位置对应的第二虚拟场景的过程，例如，如图9和图11，终端显示以第一虚拟对象为中心的虚拟场景900，如果用户对地图控件进行了第一触发操作，终端可以获取对应的第二虚拟场景，并进行虚拟场景的切换，切换后如图11所示，不再是以第一虚拟对象为中心的虚拟场景，可以是第二虚拟场景1100。

703、终端响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置。

在该步骤703中，终端响应于对目标技能的释放操作，根据该释放操作对应的第二操作位置，确定该第二操作位置在该第二虚拟场景中对应的技能释放目标。

该释放操作的第二操作位置不同的情况下，该技能的释放方式可能不同，相应地，根据第二操作位置确定技能释放目标的过程又不同。

在一些实施例中，该释放操作为对技能控件的第二触发操作。例如，如果该释放操作为主动释放，终端响应于对目标技能的技能控件的第二触发操作，根据该第二触发操作的第二操作位置与该技能控件之间的位置关系，确定该第二操作位置在该第二虚拟场景中对应的技能释放目标。

在一些实施例中，上述释放操作持续过程中，用户能够通过改变触发操作的操作位置，来改变技能释放目标，该目标技能的最终技能释放目标由释放操作的结束位置确定。例如，

终端响应于对目标技能的释放操作结束，将该释放操作的结束位置获取为该第二操作位置，且该第二操作位置位于第一操作区域内，执行该根据该第二操作位置与该技能控件之间的位置关系，确定该第二操作位置在该第二虚拟场景中对应的技能释放目标。

在该主动释放方式中，该技能控件的操作区域与虚拟场景之间具有对应关系，对该操作区域中某一位置的操作会被映射至虚拟场景中的对应位置，该操作区域中的位置关系可以映射为虚拟场景中的位置关系。

在一些实施例中，该技能释放目标为技能释放位置、目标虚拟对象或技能释放方向中的任一项，上述步骤 703 中，根据第二操作位置确定技能释放目标的过程通过下述步骤一至步骤三实现。

步骤一、终端获取该第二操作位置与该技能控件之间的位置关系。

在一些实施例中，该位置关系通过该第二操作位置与技能控件的中心位置获取得到。例如，该位置关系指代位移，表示为一个方向向量，该方向向量从技能控件的中心位置指向第二操作位置。如图 12 所示，假设技能控件的中心位置为 A，第二操作位置为 B，该位置关系表现为 A 指向 B 的向量 B-A。

步骤二、终端根据该技能控件的操作区域与虚拟场景之间的转换关系，对该第二操作位置与该技能控件之间的位置关系进行转换，得到技能释放位置与该第二虚拟场景的中心位置之间的目标位置关系。

该技能的操作区域与虚拟场景之间具有一定的转换关系，该操作区域为一个 2D 的区域，虚拟场景为 3D 的虚拟空间，且该操作区域的尺寸与虚拟场景的尺寸并不相同，因而二者之间还包括一定缩放比例的映射。

下面基于图 13 所示，来说明该转换关系，假设“小地图视野中心”到点“A”的线段为 X，“小地图视野中心”到点“A”的方向上，延长到屏幕边缘的线段为 Y，第二操作位置到技能控件的中心位置的线段为 Z，则“小地图视野中心”到点“A”的方向与轮盘中心到技能摇杆的方向相同，即“X 平行于 Z”。X 线段长度和 Y 线段长度的比例等同于 Z 线段长度和轮盘半径的比例，也即是 $X = (Z / \text{轮盘半径}) * Y$ 。基于方向和长度，能够得到点“A”的坐标。再把技能指示器按照规则显示在点“A”位置上。

如图 14 所示，通过虚拟摄像机观察虚拟场景，其观察到的区域为一个梯形区域。假设该操作区域为圆形区域，该转换关系为用于将圆形区域转换为椭圆形区域的映射关系，或者用于将圆形区域转换为梯形区域的映射关系，本申请实施例对具体采用哪种方式不作限定。

在一些实施例中，可以提供一个映射关系选项，由用户根据自身需求来从映射关系选项中选择使用的一种映射关系。终端根据该映射关系选项中设定的目标映射关系来进行上述步骤二。

在一些实施例中，终端根据该第二虚拟场景的中心位置，确定该第二虚拟场景的边缘位置，根据该第二虚拟场景的中心位置、该第二虚拟场景的边缘位置以及该操作区域的尺寸，将该第二操作位置与该技能控件之间的位置关系进行映射。

步骤三、终端根据该第二虚拟场景的中心位置以及该目标位置关系，确定该释放操作的第二操作位置在该第二虚拟场景中对应的技能释放位置，将该技能释放位置确定为该技能释放目标，或者将该技能释放位置上的虚拟对象确定为目标虚拟对象，或者将该技能释放位置相对于当前控制的第一虚拟对象的方向确定为技能释放方向。

在确定了目标位置关系，该目标位置关系即为技能释放位置相对于第二虚拟场景的中心位置的位置关系，从而可以根据该中心位置以及该目标位置关系，得到该技能释放位置。

下面针对上述两种转换关系中，确定技能释放位置的过程进行说明，在圆形区域映射为椭圆形区域的方式中，如图 15 所示，该操作区域也可以称之为技能拖动范围，第一步，如图 12 所示，将(B-A)将 UI 向量转化为场景中的向量，以屏幕中心点 Ta 加上(B-A)向量得到 Tb 点，再通过 2d 转 3d 的方式将 UI 上的 Ta, Tb 点转换成场景中 Da, Db 点，如图 16 所示。进而可以得到在虚拟场景中对应的方向向量 AimDir， $AimDir = Normalize(Db - Da)$ 。其中，Normalize 为归一化函数。

在第二步中，可以计算屏幕中心点的场景位置 (AimCameraPos) 与屏幕边缘的距离。在此为了避免将技能释放位置拖动到屏幕边缘处的 UI 上，上述距离可以为去除了边界值的距离。具体的，可以设置四个值来分别表示去除了边界值的屏幕边缘的距离，例如，该四个值分别为 paddingLeft (左边距)，paddingRight (右边距)，paddingTop (上边距)，paddingBot (底边距)，分别表示场景中在屏幕范围左右上下四条边与 AimCameraPos (屏幕中心点的场景位置) 的距离。对于四个值的获取过程同理，在此以计算 paddingTop 为例进行说明。首先，将 AimCameraPos 转换成 UICenterPos (UI 中心位置)，也即是 3D 的坐标点转换为 2D 坐标点，然后用 UICenterPos 加上屏幕高度的一半，并减去需要去除的边界值获得 UITopPos，然后再将 UITopPos 转换成 3d 虚拟场景中的位置 SceneTopPos，最后用 $(SceneTopPos - AimCameraPos).z$ ，即可获取到 paddingTop，其他边界值同理获取。

在第三步中，可以通过公式六和公式七，根据上述步骤计算得到的 AimCameraPos、方向向量 AimDir 以及各个方向的最大值，计算技能释放位置 FocusPoint。

$FocusPoint.x = AimCameraPos.x + AimDir.x * (|B-A| / AimMaxRadius) * (AimDir.x < 0 ? paddingLeft : paddingRight)$ ，公式六

$FocusPoint.z = AimCameraPos.z + AimDir.z * (|B-A| / AimMaxRadius) * (AimDir.y < 0 ? paddingBot : paddingTop)$ ，公式七

上述公式六和七中，MaxAimRadius 为技能按钮的瞄准的最大范围，|B-A| 表示技能按钮拖动的距离，FocusPoint 为技能释放位置，AimCameraPos 为第二虚拟场景中屏幕中心点的场景位置，BorderLength 为屏幕中心点与屏幕边缘之间的边界长度，AimDir 为向量(B-A)在虚拟场景中对应的方向向量。 $(AimDir.x < 0 ? paddingLeft : paddingRight)$ 用于表示 AimDir.x < 0 满足时，取 paddingLeft，AimDir.x < 0 不满足时，取 paddingRight。

在圆形区域映射为梯形区域的方式中，如图 17，首先计算处于梯形范围的四个顶点在场景中的位置，分别为 LT 为左上点，LB 为左下点，RT 为右上点，RB 为右下点。

然后，根据 AimDir 的值判断 AimCameraPos 顺着 AimDir 与梯形的相交点，这里比较简单，比如 $AimDir.x > 0 \ \&\& \ AimDir.y > 0$ 需要判断 AimCameraPos 已 AimDir 朝向的射线与 (RT-LT) 线段的交点，再用这条射线找到与 (RT-RB) 的交点，而这两个交点与 AimCameraPos 距离较近的一点为计算所需要的点，通过线段与线段的交点公式即可确定，之后通过下述公式八计算技能释放位置。

$FocusPoint = AimCameraPos + (|B-A| / MaxAimRadius) * BorderLength * AimDir$ ，公式八

上述公式八中，MaxAimRadius 为技能按钮的瞄准的最大范围，|B-A| 表示技能按钮拖动的距离，FocusPoint 为技能释放位置，AimCameraPos 为第二虚拟场景中屏幕中心点的场景位

置, BorderLength 为屏幕中心点与屏幕边缘之间的边界长度, AimDir 为向量(B-A)在虚拟场景中对应的方向向量。

上述过程为主动释放方式中如何确定技能释放位置的过程, 如果为快速释放, 终端响应于对目标技能的释放操作结束, 且该释放操作结束时的第二操作位置位于第二操作区域内, 根据该第二虚拟场景中至少一个第二虚拟对象的信息, 从该至少一个第二虚拟对象中确定目标虚拟对象, 将该目标虚拟对象确定为该技能释放目标, 或者将该目标虚拟对象所在位置确定为该技能释放目标, 或者将该目标虚拟对象相对于第一虚拟对象的方向确定为该技能释放目标, 该第一操作区域包围于该第二操作区域外。

其中, 根据该第二虚拟场景中至少一个第二虚拟对象的信息, 从该至少一个第二虚拟对象中确定目标虚拟对象的过程可以参考的信息可以不同, 例如, 虚拟生命值, 或者与第一虚拟对象之间的距离等, 本申请实施例对此不作具体限定。在一些实施例中, 终端根据虚拟场景中至少一个虚拟对象的信息, 确定该技能的候选释放目标信息的过程可以通过释放目标确定规则实现, 该释放目标确定规则用于确定释放目标, 因而也可以称该释放目标确定规则为搜索规则。该释放目标确定规则可以由相关技术人员根据需求进行设置, 也可以由该用户根据自身使用习惯进行设置, 本申请实施例对此不作限定。例如, 终端可以根据虚拟场景中至少一个虚拟对象的信息, 选择敌方或己方生命值最小的目标虚拟对象。又例如, 将距离自己最近的一个虚拟对象作为目标虚拟对象。又例如, 选择优先级最高的虚拟对象。

上述主动释放和快速释放两种方式中, 均以释放操作结束时的第二操作位置为准, 来进行技能释放目标的确定步骤, 也即是, 上述步骤 703 可以为终端响应于对目标技能的释放操作结束, 根据该释放操作结束时的第二操作位置, 确定该第二操作位置在该第二虚拟场景中对应的技能释放目标。

在一些实施例中, 在释放操作持续过程中, 终端还可以获取并突出显示候选技能释放目标, 从而用户能够根据自身需求来判断是否符合预期。在一些实施例中, 终端可以响应于对目标技能的释放操作持续过程中, 根据该释放操作的第二操作位置, 确定该第二操作位置在该第二虚拟场景中对应的候选技能释放目标, 在该第二虚拟场景中对该候选技能释放目标进行突出显示。例如, 如图 13 所示, 可以突出显示候选技能释放目标, 如果此时释放操作结束, 即可将其作为第二操作位置对应的释放位置。

在一些实施例中, 该目标技能具有释放范围, 该目标技能的释放不能超出该释放范围。在该实现方式中, 终端可根据当前控制的第一虚拟对象在虚拟场景中的位置以及该目标技能的释放范围, 确定该目标技能的可释放区域, 其中, 该可释放区域是指该技能能够释放的区域, 技能无法释放到该可释放区域外的位置。例如, 一些技能具有释放距离(也即是可释放范围), 根据该释放距离能够确定出一个可释放区域, 技能无法释放到超出该释放距离的位置, 也就无法释放到可释放区域外的位置。

终端获取到可释放区域后, 可以判断当前选定的释放位置是否在该可释放区域内, 终端可以响应于该释放操作的第二操作位置在该第二虚拟场景中对应的位置位于该可释放区域内, 执行该步骤 703。当然, 还有另一种可能情况, 终端可以响应于该释放操作的第二操作位置在该第二虚拟场景中对应的位置位于该可释放区域外, 根据该释放操作的第二操作位置以及该第一虚拟对象在虚拟场景中的位置, 确定该第二操作位置在该虚拟场景中对应的技能释放目标。当然, 在该另一种可能情况中, 终端也可以不执行该技能释放目标的选择步骤,

取消对目标技能的释放。

704、终端控制第一虚拟对象按照该技能释放目标释放该目标技能。

终端确定技能释放目标之后，即可控制第一虚拟对象释放该技能。例如，确定了目标虚拟对象为第二虚拟对象 A，该技能为对选中目标发射火球。终端显示的释放效果即可为：对第二虚拟对象 A 发射火球。

在一些实施例中，该技能的释放效果可以通过该技能的释放动画实现，例如，该步骤 704 中，终端可以获取该技能的释放动画，在第一虚拟对象和目标虚拟对象之间播放该释放动画。

本申请实施例通过根据对地图控件的第一触发操作的第一操作位置，在用户图形界面中显示该第一操作位置对应的第二虚拟场景，在这种情况下，响应于对目标技能的释放操作，能够根据释放操作的第二操作位置，来确定该目标技能在当前显示的第二虚拟场景中对应的技能释放目标，从而进行技能释放，在上述通过地图控件来控制技能释放的方式中，能够在对地图控件进行第一触发操作时，显示相应的第二虚拟场景，这样技能释放目标的选择范围可以不局限于以虚拟对象为中心的虚拟场景中，释放操作的自由度更高，且在技能释放时，能够精准地根据想要释放位置的情况来进行选择，而非在当前显示的虚拟场景中粗略估计，提高了对虚拟对象的控制方法的精度和准确性。

下面通过具体示例，对上述方法流程进行示例性说明。如图 18 所示，对于小地图（也即是地图控件）逻辑，用户可以对小地图进行操作，例如，按下/拖动/抬起操作。终端可以根据触摸点位置（第一操作位置）映射出场景位置（也即是第一操作位置在虚拟场景中对应的位置），终端可以将映射出的场景位置设置给 AimCameraPos，该 AimCameraPos 可以在后续被获取（Get），进行后续的逻辑计算。如果没有对小地图进行操作，则后续计算时，可以获取 CenterActorPos（第一虚拟对象所在位置）进行计算。

在此将对小地图操作后触发技能的方式称为小地图瞄准机制，将未对小地图进行操作触发技能的方式称为普通技能瞄准机制。在技能按钮操作时，可以判断是否进行了拖动，如果否，则为快速释放方式，如果是，则为主动释放方式。在快速释放方式中，可以判断 AimCameraPos（屏幕中心点的场景位置）是否合法也就是判断是否有小地图操作，如果没有小地图的相关操作，那么直接将当前玩家控制英雄（第一虚拟对象）的位置 CenterActorPos 赋值给 FocusPoint（技能释放位置），如果有小地图操作，那么 AimCameraPos 合法，将 AimCameraPos 的值赋值给 FocusPoint。在主动释放方式中，也可以判断 AimCameraPos 是否合法，如果合法，则通过小地图瞄准机制计算 FocusPoint，如果不合法，则通过简单技能瞄准机制计算 FocusPoint。计算得到 FocusPoint 后，可以通过普通技能释放逻辑同理的步骤，具体如下：

- 1.以玩家当前位置 ActorPos，FocusPoint（技能释放位置），技能范围等作为参数搜索合适的技能目标。

- 2.以玩家当前位置 ActorPos 和 FocusPoint 以及第一步搜索的目标为参数显示出技能指示器，该技能指示器用于对技能目标进行预览显示。

从镜头步骤我们知道当有小地图拖动时，AimCameraPos 是我们的屏幕的中心点，而在快速点击的流程中我们也是将 AimCameraPos 赋值给 FocusPoint，而我们的技能流程中根据 FocusPoint 位置的不同表现出不同的技能表现，镜头和技能同时以 AimCameraPos 为基础点的逻辑让我们的在操作上达到了所见即所得的目的。

对于普通技能瞄准机制，在没有小地图拖动的方案中，即可通过下述公式九得到技能释放位置 FocusPoint。

$\text{FocusPoint} = H + \text{Normalize}(\text{Db} - \text{Da}) * (|\text{B}-\text{A}| * M)$ ，公式九

对于公式九中的参数，可以如图 19 所示，假设第一虚拟对象在 H 点，技能范围为 SkillRange，可以在 UI 层拿到一个(B-A)的瞄准向量，|B-A|是(B-A)的长度，H 点即第一虚拟对象当前的位置，还有 Normalize(Db - Da) 为(Db-Da)的归一化后的向量，套入上述公式九得出 FocusPoint，M 为技能范围的半径。

对于镜头更新逻辑，镜头帧更新时，可以判断 AimCameraPos 是否合法，如果合法，则镜头跟随小地图中的屏幕中心点 AimCameraPos，如果否，镜头跟随第一虚拟对象的位置 (CenterActorPos)。

图 20 是本申请实施例提供的一种虚拟对象的控制装置的结构示意图，该装置包括：

显示模块 2001，用于响应于对地图控件的第一触发操作，根据该第一触发操作的第一操作位置，将用户图形界面中显示的虚拟场景切换至该第一操作位置对应的目标虚拟场景；在一些实施例中，该显示模块 2001 用于显示第一虚拟场景，所述第一虚拟场景中显示有地图控件；响应于对所述地图控件的第一触发操作，显示第一操作位置对应的第二虚拟场景，所述第一触发操作作用于所述第一操作位置；

确定模块 2002，用于响应于对目标技能的释放操作，根据该释放操作的第二操作位置，确定该第二操作位置在该目标虚拟场景中对应的技能释放目标；在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置；

控制模块 2003，用于控制第一虚拟对象按照该技能释放目标释放该目标技能。

在一些实施例中，该显示模块 2001 包括第一获取单元和显示单元，其中：

该第一获取单元用于根据该第一触发操作的第一操作位置以及该地图控件中显示信息与虚拟场景的对应关系，获取该第一操作位置对应的目标虚拟场景；在一些实施例中，该第一获取单元用于根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置对应的第二虚拟场景；

该显示单元用于将用户图形界面中显示的虚拟场景切换至该目标虚拟场景。在一些实施例中，该显示单元用于将所述第一虚拟场景切换至所述第二虚拟场景。

在一些实施例中，该第一获取单元用于执行下述任一项：

根据该第一触发操作的第一操作位置，获取该地图控件中，以该第一操作位置为中心、尺寸为第一目标尺寸的目标区域，根据该地图控件中的区域与虚拟场景的对应关系，获取该目标区域对应的目标虚拟场景；

根据该第一操作位置以及该地图控件中的位置与虚拟场景的对应关系，获取该第一操作位置对应的虚拟场景中的目标位置，获取以该目标位置为中心、尺寸为第二目标尺寸的目标虚拟场景。

在一些实施例中，该第一获取单元用于执行下述任一项：

根据所述第一触发操作的第一操作位置，在所述地图控件中，确定以所述第一操作位置为中心、尺寸为第一目标尺寸的目标区域，根据所述地图控件中显示信息与虚拟场景的对应关系，确定所述目标区域在虚拟场景中对应的第二虚拟场景；

根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置在所述虚拟场景中对应的位置，确定以所述位置为中心、尺寸为第二目标尺寸的第二虚拟场景。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的技能控件的第二触发操作，根据该第二触发操作的第二操作位置相对于该技能控件的位置关系，确定该第二操作位置在该目标虚拟场景中对应的技能释放目标。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的技能控件的第二触发操作，以所述第二触发操作的操作位置作为所述第二操作位置，根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

在一些实施例中，该技能释放目标为技能释放位置、目标虚拟对象或技能释放方向中的任一项；

该确定模块 2002 包括第二获取单元、转换单元和确定单元，其中：

该第二获取单元用于获取该第二操作位置相对于该技能控件的位置关系；

该转换单元用于根据该技能控件的操作区域与虚拟场景之间的转换关系，对该第二操作位置相对于该技能控件的位置关系进行转换，得到技能释放位置相对于该目标虚拟场景的中心位置的目标位置关系；

该确定单元用于根据该目标虚拟场景的中心位置以及该目标位置关系，确定该释放操作的操作位置在该目标虚拟场景中对应的技能释放位置，将该技能释放位置确定为该技能释放目标，或者将该技能释放位置上的虚拟对象确定为目标虚拟对象，或者将该技能释放位置相对于当前控制的第一虚拟对象的方向确定为技能释放方向。

在一些实施例中，该第二获取单元用于确定所述第二操作位置与所述技能控件之间的位置关系；该转换单元用于根据所述技能控件的操作区域与虚拟场景之间的转换关系，对所述位移进行转换，得到技能释放位置与所述第二虚拟场景的中心位置之间的目标位置关系；该确定单元用于根据所述第二虚拟场景的中心位置以及所述目标位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放位置，将所述技能释放位置确定为所述技能释放目标，或者，将所述技能释放位置上的虚拟对象确定为目标虚拟对象，或者，将所述技能释放位置相对于当前控制的第一虚拟对象的方向确定为技能释放方向。

在一些实施例中，该转换单元用于：根据该目标虚拟场景的中心位置，确定该目标虚拟场景的边缘位置；根据该目标虚拟场景的中心位置、该目标虚拟场景的边缘位置以及该操作区域的尺寸，对该第二操作位置相对于该技能控件的位置关系进行转换。

在一些实施例中，该转换单元用于：根据所述第二虚拟场景的中心位置，确定所述第二虚拟场景的边缘位置；根据所述第二虚拟场景的中心位置、所述第二虚拟场景的边缘位置以及所述操作区域的尺寸，对所述第二操作位置与所述技能控件之间的位置关系进行转换。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作结束，且该释放操作结束时的第二操作位置位于第一操作区域内，执行该根据该第二触发操作的第二操作位置相对于该技能控件的位置关系，确定该第二操作位置在该目标虚拟场景中对应的技能释放目标。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作结束，以所述释放操作的结束位置作为所述第二操作位置，且所述第二操作位置位于第一操作区域内，执行所

述根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作结束，且该释放操作结束时的第二操作位置位于第二操作区域内，根据该目标虚拟场景中至少一个第二虚拟对象的信息，从该至少一个第二虚拟对象中确定目标虚拟对象，将该目标虚拟对象确定为该技能释放目标，或者将该目标虚拟对象所在位置确定为该技能释放目标，或者将该目标虚拟对象相对于第一虚拟对象的方向确定为该技能释放目标，该第一操作区域包围于该第二操作区域外。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作结束，以所述释放操作的结束位置作为所述第二操作位置，且所述第二操作位置位于第二操作区域内，根据所述第二虚拟场景中至少一个第二虚拟对象的信息，从所述至少一个第二虚拟对象中确定目标虚拟对象，将所述目标虚拟对象确定为所述技能释放目标，或者，将所述目标虚拟对象所在位置确定为所述技能释放目标，或者，将所述目标虚拟对象相对于第一虚拟对象的方向确定为所述技能释放目标，所述第一操作区域包围于所述第二操作区域外。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作结束，根据该释放操作结束时的第二操作位置，确定该第二操作位置在该目标虚拟场景中对应的技能释放目标。

在一些实施例中，该确定模块 2002 用于响应于对目标技能的释放操作结束，以所述释放操作的结束位置作为所述第二操作位置，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

在一些实施例中，该确定模块 2002 还用于响应于对目标技能的释放操作持续过程中，根据该释放操作的第二操作位置，确定该第二操作位置在该目标虚拟场景中对应的候选技能释放目标；该显示模块 2001 还用于在该目标虚拟场景中对该候选技能释放目标进行突出显示。

在一些实施例中，该确定模块 2002 还用于响应于对目标技能的释放操作处于实施过程中，根据所述释放操作的实时操作位置，确定所述第二虚拟场景中的候选技能释放目标；

该显示模块 2001 还用于在所述第二虚拟场景中对所述候选技能释放目标进行突出显示。

在一些实施例中，该确定模块 2002 用于：根据当前控制的第一虚拟对象在虚拟场景中的位置以及该目标技能的释放范围，确定该目标技能的可释放区域；响应于该释放操作的第二操作位置在该目标虚拟场景中对应的位置位于该可释放区域内，执行该根据该释放操作的第二操作位置，确定该第二操作位置在该目标虚拟场景中对应的技能释放目标的步骤。

在一些实施例中，该确定模块 2002 用于：根据当前控制的第一虚拟对象在虚拟场景中的位置以及所述目标技能的释放范围，确定所述目标技能的可释放区域；响应于所述第二操作位置在所述第二虚拟场景中对应的位置位于所述可释放区域内，执行所述确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标的步骤。

在一些实施例中，该确定模块 2002 还用于响应于该释放操作的第二操作位置在该目标虚拟场景中对应的位置位于该可释放区域外，根据该释放操作的第二操作位置以及该第一虚拟对象在虚拟场景中的位置，确定该第二操作位置在该虚拟场景中对应的技能释放目标。

在一些实施例中，该确定模块 2002 还用于响应于目标位置位于所述可释放区域外，根据所述第二操作位置以及所述第一虚拟对象在虚拟场景中的位置，确定所述第二操作位置在所述虚拟场景中对应的技能释放目标，所述目标位置为所述第二操作位置在所述第二虚拟场景

中对应的位置。

本申请实施例提供的装置，通过根据对地图控件的第一触发操作的第一操作位置，在用户图形界面中显示该第一操作位置对应的目标虚拟场景，在这种情况下，响应于对目标技能的释放操作，能够根据释放操作的第二操作位置，来确定该目标技能在当前显示的目标虚拟场景中对应的技能释放目标，从而进行技能释放，在上述通过地图控件来控制技能释放的方式中，能够在对地图控件进行第一触发操作时，显示相应的目标虚拟场景，这样技能释放目标的选择范围可以不局限于以虚拟对象为中心的虚拟场景中，释放操作的自由度更高，且在技能释放时能够精准地根据想要释放位置的情况来进行选择，而非在当前显示的虚拟场景中粗略估计，提高了对虚拟对象的控制方法的精度和准确性。

需要说明的是：上述实施例提供的虚拟对象的控制装置在控制虚拟对象时，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，即将电子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，上述实施例提供的虚拟对象的控制装置与虚拟对象的控制方法实施例属于同一构思，其具体实现过程详见虚拟对象的控制方法实施例，这里不再赘述。

上述电子设备可以被提供为下述图 21 所示的终端。

图 21 是本申请实施例提供的一种终端 2100 的结构示意图，该终端 2100 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 2100 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 2100 包括有：处理器 2101 和存储器 2102。

处理器 2101 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 2101 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 2101 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器，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 2101 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 2101 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 2102 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 2102 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 2102 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 2101 所执行以实现本申请中各个实施例提供的虚拟对象的控制方法中终端侧的方法步骤。

在一些实施例中，终端 2100 还可选包括有：外围设备接口 2103 和至少一个外围设备。处理器 2101、存储器 2102 和外围设备接口 2103 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 2103 相连。在一些实施例中，外围设备包括：射频电路 2104、触摸显示屏 2105、音频电路 2106 中的至少一种。

外围设备接口 2103 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 2101 和存储器 2102。在一些实施例中, 处理器 2101、存储器 2102 和外围设备接口 2103 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 2101、存储器 2102 和外围设备接口 2103 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。

显示屏 2105 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 2105 是触摸显示屏时, 显示屏 2105 还具有采集在显示屏 2105 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 2101 进行处理。此时, 显示屏 2105 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 2105 可以为一个, 设置终端 2100 的前面板; 在另一些实施例中, 显示屏 2105 可以为至少两个, 分别设置在终端 2100 的不同表面或呈折叠设计; 在再一些实施例中, 显示屏 2105 可以是柔性显示屏, 设置在终端 2100 的弯曲表面上或折叠面上。甚至, 显示屏 2105 还可以设置成非矩形的不规则图形, 也即异形屏。显示屏 2105 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。

音频电路 2106 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 2101 进行处理, 或者输入至射频电路 2104 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 2100 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 2101 或射频电路 2104 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 2106 还可以包括耳机插孔。

在一些实施例中, 终端 2100 还包括有一个或多个传感器 2110。该一个或多个传感器 2110 包括但不限于: 加速度传感器 2111、陀螺仪传感器 2112、压力传感器 2113。

加速度传感器 2111 可以检测以终端 2100 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 2111 可以用于检测重力加速度在三个坐标轴上的分量。处理器 2101 可以根据加速度传感器 2111 采集的重力加速度信号, 控制触摸显示屏 2105 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 2111 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 2112 可以检测终端 2100 的机体方向及转动角度, 陀螺仪传感器 2112 可以与加速度传感器 2111 协同采集用户对终端 2100 的 3D 动作。处理器 2101 根据陀螺仪传感器 2112 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 2113 可以设置在终端 2100 的侧边框和/或触摸显示屏 2105 的下层。当压力传感器 2113 设置在终端 2100 的侧边框时, 可以检测用户对终端 2100 的握持信号, 由处理器 2101 根据压力传感器 2113 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 2113 设置在触摸显示屏 2105 的下层时, 由处理器 2101 根据用户对触摸显示屏 2105 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。

本领域技术人员可以理解，图 21 中示出的结构并不构成对终端 2100 的限定，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。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置对应的第二虚拟场景；将所述第一虚拟场景切换至所述第二虚拟场景。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述任一项步骤：

根据所述第一触发操作的第一操作位置，在所述地图控件中，确定以所述第一操作位置为中心、尺寸为第一目标尺寸的目标区域，根据所述地图控件中显示信息与虚拟场景的对应关系，确定所述目标区域在虚拟场景中对应的第二虚拟场景；

根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置在所述虚拟场景中对应的位置，确定以所述位置为中心、尺寸为第二目标尺寸的第二虚拟场景。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：响应于对目标技能的技能控件的第二触发操作，以所述第二触发操作的操作位置作为所述第二操作位置，根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述任一项步骤：

所述根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标，包括：

确定所述第二操作位置与所述技能控件之间的位置关系；

根据所述技能控件的操作区域与虚拟场景之间的转换关系，对所述位移进行转换，得到技能释放位置与所述第二虚拟场景的中心位置之间的目标位置关系；

根据所述第二虚拟场景的中心位置以及所述目标位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放位置，将所述技能释放位置确定为所述技能释放目标，或者，将所述技能释放位置上的虚拟对象确定为目标虚拟对象，或者，将所述技能释放位置相对于当前控制的第一虚拟对象的方向确定为技能释放方向。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：根据所述第二虚拟场景的中心位置，确定所述第二虚拟场景的边缘位置；

根据所述第二虚拟场景的中心位置、所述第二虚拟场景的边缘位置以及所述操作区域的尺寸，对所述第二操作位置与所述技能控件之间的位置关系进行转换。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：响应于对目标技能的释放操作结束，以所述释放操作的结束位置作为所述第二操作位置，且所述第二操作位置位于第一操作区域内，执行所述根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：响应于对目标技能的释放操作结束，以所述释放操作的结束位置作为所述第二操作位置，且所述第二操作位置位于第二操作区域内，根据所述第二虚拟场景中至少一个第二虚拟对象的信息，从所述至少一个第二虚拟对象中确定目标虚拟对象，将所述目标虚拟对象确定为所述技能释放目标，或者，将所述目标虚拟对象所在位置确定为所述技能释放目标，或者，将所述目标虚

拟对象相对于第一虚拟对象的方向确定为所述技能释放目标，所述第一操作区域包围于所述第二操作区域外。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：响应于对目标技能的释放操作结束，以所述释放操作的结束位置作为所述第二操作位置，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：响应于对目标技能的释放操作处于实施过程中，根据所述释放操作的实时操作位置，确定所述第二虚拟场景中的候选技能释放目标；

在所述第二虚拟场景中对所述候选技能释放目标进行突出显示。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：根据当前控制的第一虚拟对象在虚拟场景中的位置以及所述目标技能的释放范围，确定所述目标技能的可释放区域；

响应于所述第二操作位置在所述第二虚拟场景中对应的位置位于所述可释放区域内，执行所述确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标的步骤。

在一些实施例中，该至少一个指令用于被处理器所执行以实现下述方法步骤：响应于目标位置位于所述可释放区域外，根据所述第二操作位置以及所述第一虚拟对象在虚拟场景中的位置，确定所述第二操作位置在所述虚拟场景中对应的技能释放目标，所述目标位置为所述第二操作位置在所述第二虚拟场景中对应的位置。

在示例性实施例中，还提供了一种计算机可读存储介质，例如包括至少一条程序代码的存储器，上述至少一条程序代码可由电子设备中的处理器执行以完成上述实施例中虚拟对象的控制方法。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。

上述仅为本申请的可选实施例，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

1.一种虚拟对象的控制方法，其中，所述方法包括：

显示第一虚拟场景，所述第一虚拟场景中显示有地图控件；

响应于对所述地图控件的第一触发操作，显示第一操作位置对应的第二虚拟场景，所述第一触发操作作用于所述第一操作位置；

响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置；

控制第一虚拟对象按照所述技能释放目标释放所述目标技能。

2.根据权利要求1所述的方法，其中，所述显示第一操作位置对应的第二虚拟场景，包括：

根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置对应的第二虚拟场景；

将所述第一虚拟场景切换至所述第二虚拟场景。

3.根据权利要求2所述的方法，其中，所述根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置对应的第二虚拟场景，包括下述任一项：

根据所述第一触发操作的第一操作位置，在所述地图控件中，确定以所述第一操作位置为中心、尺寸为第一目标尺寸的目标区域，根据所述地图控件中显示信息与虚拟场景的对应关系，确定所述目标区域在虚拟场景中对应的第二虚拟场景；

根据所述第一操作位置以及所述地图控件中显示信息与虚拟场景的对应关系，确定所述第一操作位置在所述虚拟场景中对应的位置，确定以所述位置为中心、尺寸为第二目标尺寸的第二虚拟场景。

4.根据权利要求1所述的方法，其中，所述响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，包括：

响应于对目标技能的技能控件的第二触发操作，以所述第二触发操作的操作位置作为所述第二操作位置，根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

5.根据权利要求4所述的方法，其中，所述技能释放目标为技能释放位置、目标虚拟对象或技能释放方向中的任一项；

所述根据所述第二操作位置与所述技能控件之间的位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标，包括：

确定所述第二操作位置与所述技能控件之间的位置关系；

根据所述技能控件的操作区域与虚拟场景之间的转换关系，对所述位移进行转换，得到技能释放位置与所述第二虚拟场景的中心位置之间的目标位置关系；

根据所述第二虚拟场景的中心位置以及所述目标位置关系，确定所述第二操作位置在所述第二虚拟场景中对应的技能释放位置，将所述技能释放位置确定为所述技能释放目标，或者，将所述技能释放位置上的虚拟对象确定为目标虚拟对象，或者，将所述技能释放位置相

对于当前控制的第一虚拟对象的方向确定为技能释放方向。

6.根据权利要求5所述的方法,其中,所述根据所述技能控件的操作区域与虚拟场景之间的转换关系,对所述位置关系进行转换,包括:

根据所述第二虚拟场景的中心位置,确定所述第二虚拟场景的边缘位置;

根据所述第二虚拟场景的中心位置、所述第二虚拟场景的边缘位置以及所述操作区域的尺寸,对所述第二操作位置与所述技能控件之间的位置关系进行转换。

7.根据权利要求4所述的方法,其中,所述响应于对目标技能的释放操作,响应于对目标技能的释放操作,确定第二操作位置在所述第二虚拟场景中对应的技能释放目标,包括:

响应于对目标技能的释放操作结束,以所述释放操作的结束位置作为所述第二操作位置,且所述第二操作位置位于第一操作区域内,执行所述根据所述第二操作位置与所述技能控件之间的位置关系,确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

8.根据权利要求1所述的方法,其中,所述响应于对目标技能的释放操作,确定第二操作位置在所述第二虚拟场景中对应的技能释放目标,包括:

响应于对目标技能的释放操作结束,以所述释放操作的结束位置作为所述第二操作位置,且所述第二操作位置位于第二操作区域内,根据所述第二虚拟场景中至少一个第二虚拟对象的信息,从所述至少一个第二虚拟对象中确定目标虚拟对象,将所述目标虚拟对象确定为所述技能释放目标,或者,将所述目标虚拟对象所在位置确定为所述技能释放目标,或者,将所述目标虚拟对象相对于第一虚拟对象的方向确定为所述技能释放目标,所述第一操作区域包围于所述第二操作区域外。

9.根据权利要求1所述的方法,其中,所述响应于对目标技能的释放操作,确定第二操作位置在所述第二虚拟场景中对应的技能释放目标,包括:

响应于对目标技能的释放操作结束,以所述释放操作的结束位置作为所述第二操作位置,确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标。

10.根据权利要求1所述的方法,其中,所述方法还包括:

响应于对目标技能的释放操作处于实施过程中,根据所述释放操作的实时操作位置,确定所述第二虚拟场景中的候选技能释放目标;

在所述第二虚拟场景中对所述候选技能释放目标进行突出显示。

11.根据权利要求1所述的方法,其中,所述响应于对目标技能的释放操作,确定第二操作位置在所述第二虚拟场景中对应的技能释放目标,包括:

根据当前控制的第一虚拟对象在虚拟场景中的位置以及所述目标技能的释放范围,确定所述目标技能的可释放区域;

响应于所述第二操作位置在所述第二虚拟场景中对应的位置位于所述可释放区域内,执行所述确定所述第二操作位置在所述第二虚拟场景中对应的技能释放目标的步骤。

12.根据权利要求11所述的方法,其中,所述方法还包括:

响应于目标位置位于所述可释放区域外,根据所述第二操作位置以及所述第一虚拟对象在虚拟场景中的位置,确定所述第二操作位置在所述虚拟场景中对应的技能释放目标,所述目标位置为所述第二操作位置在所述第二虚拟场景中对应的位置。

13.一种虚拟对象的控制装置,其中,所述装置包括:

显示模块,用于显示虚拟场景,所述虚拟场景中显示有地图控件,响应于对所述地图控

件的第一触发操作，显示第一操作位置对应的第二虚拟场景，所述第一触发操作作用于所述第一操作位置；

确定模块，用于响应于对目标技能的释放操作，基于第二操作位置，在所述第二虚拟场景中确定对应的技能释放目标，所述释放操作对应于第二操作位置；

控制模块，用于控制第一虚拟对象按照所述技能释放目标释放所述目标技能。

14.一种电子设备，其中，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中存储有至少一条程序代码，所述至少一条程序代码由所述一个或多个处理器加载并执行以实现如权利要求1至权利要求12任一项所述的虚拟对象的控制方法所执行的操作。

15.一种存储介质，其中，所述存储介质中存储有至少一条程序代码，所述至少一条程序代码由处理器加载并执行以实现如权利要求1至权利要求12任一项所述的虚拟对象的控制方法所执行的操作。