

虚拟对象的控制方法、装置、设备、存储介质及程序产品

本申请要求于 2022 年 08 月 03 日提交的申请号为 202210928035.6、发明名称为“虚拟对象的控制方法、装置、设备、存储介质及程序产品”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

本申请实施例涉及计算机和互联网技术领域，特别涉及一种虚拟对象的控制方法、装置、设备、存储介质及程序产品。

背景技术

对战游戏是一种多个用户帐号在同一场景内进行竞技的游戏。可选地，对战游戏可以是多人在线战术竞技（Multiplayer Online Battle Arena，MOBA）游戏。

在典型的 MOBA 游戏中，用户可以控制虚拟对象在虚拟环境中释放技能，每个虚拟对象各自具有一套技能。例如，虚拟对象具有第一技能、第二技能和第三技能，则用户界面上可设置有与该三个技能分别对应的三个技能控件，用户通过触发技能控件，来控制虚拟对象使用对应的技能。

相关技术中，如果想控制虚拟对象使用组合技能，则只能简单对该三个技能进行组合，因此技能释放的组合方式较少。

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、存储介质及程序产品。所述技术方案如下：

根据本申请实施例的一个方面，提供了一种虚拟对象的控制方法，所述方法由终端设备执行，所述方法包括：

在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，所述技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能；

响应于针对所述至少两个技能控件中的第一技能控件的操作，从所述至少两个技能点中确定将使用的第一技能点；

控制第一虚拟对象释放所述第一技能控件对应的技能组中，与所述第一技能点相对应的技能。

根据本申请实施例的一个方面，提供了一种虚拟对象的控制装置，所述装置包括：

控件显示模块，用于在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，所述技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能；

技能点确定模块，用于响应于针对所述至少两个技能控件中的第一技能控件的操作，从所述至少两个技能点中确定将使用的第一技能点；

控制模块，用于控制第一虚拟对象释放所述第一技能控件对应的技能组中，与所述第一技能点相对应的技能。

根据本申请实施例的一个方面，提供了一种终端设备，所述终端设备包括处理器和存储器，所述存储器中存储有计算机程序，所述计算机程序由所述处理器加载并执行以实现上述方法。

根据本申请实施例的一个方面，提供了一种计算机可读存储介质，所述可读存储介质中存储有计算机程序，所述计算机程序由处理器加载并执行以实现上述方法。

根据本申请实施例的一个方面，提供了一种计算机程序产品，该计算机程序产品包括计算机程序，该计算机程序存储在计算机可读存储介质中。终端设备的处理器从计算机可读存储介质读取该计算机程序，处理器执行该计算机程序，使得该终端设备执行上述方法。

本申请实施例提供的技术方案可以包括如下有益效果：通过支持至少两个技能控件共用同一技能点集合，且当用户执行针对至少两个技能控件中的第一个技能控件的操作时，从技能点集合中确定出将使用的第一技能点，同时控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能，实现了通过技能点来确定当前所触发的技能控件对应的当前技能。因此，在有至少两个技能点、至少两个技能控件的情况下，根据不同的技能点选择和不同技能控件的选择，技能的释放有多种可能的组合方式，从而实现技能释放的组合多样化，进而提升技能释放的组合灵活性和多样性。

附图说明

- 图1是本申请一个实施例提供的方案实施环境的示意图；
图2是本申请一个实施例提供的用户界面的示意图；
图3是本申请另一个实施例提供的用户界面的示意图；
图4是本申请另一个实施例提供的用户界面的示意图；
图5-9是相关技术示例性示出的用户界面的示意图；
图10是本申请一个实施例提供的虚拟对象的控制方法的流程图；
图11是本申请另一个实施例提供的虚拟对象的控制方法的流程图；
图12是本申请另一个实施例提供的虚拟对象的控制方法的流程图；
图13是本申请另一个实施例提供的虚拟对象的控制方法的流程图；
图14是相关技术一个实施例提供的虚拟对象的控制方法的框图；
图15是本申请一个实施例提供的虚拟对象的控制方法的框图；
图16是本申请另一个实施例提供的虚拟对象的控制方法的框图；
图17是本申请一个实施例提供的技能切换逻辑的框图；
图18是本申请一个实施例提供的CD（Cool Down Time，冷却时间）切换逻辑的框图；
图19是本申请一个实施例提供的虚拟对象的控制装置的框图；
图20是本申请另一个实施例提供的虚拟对象的控制装置的框图；
图21是本申请一个实施例提供的终端设备的结构框图。

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。

首先，对本申请实施例中涉及的名词进行简单介绍：

虚拟环境：是应用程序在终端上运行时显示（或提供）的虚拟环境。该虚拟环境可以是真实世界的仿真世界，也可以是半仿真半虚构的三维世界，还可以是纯虚构的三维世界。虚拟环境可以是二维虚拟环境、2.5维虚拟环境和三维虚拟环境中的任意一种。可选地，该虚拟环境还用于至少两个虚拟对象之间的对战，如为该至少两个虚拟对象提供对战环境。该虚拟环境中可以具有可供至少两个虚拟对象使用的虚拟资源。可选地，该虚拟环境可以包括对称的左下角区域和右上角区域，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。

虚拟对象：是指在虚拟环境中的可活动对象和不可活动对象。该可活动对象可以是虚拟人物、虚拟动物、动漫人物中的至少一种。该不可活动对象可以是虚拟建筑、虚拟植物、虚拟地形中的至少一种。可选地，当虚拟环境为三维虚拟环境时，虚拟对象可以是三维虚拟模型，每个虚拟对象在三维虚拟环境中具有自身的形状和体积，占据三维虚拟环境中的一部分空间。可选地，虚拟对象是基于三维人体骨骼技术构建的三维角色，该虚拟对象通过穿戴不同的皮肤来实现不同的外在形象。在一些实现方式中，虚拟对象也可以采用2.5维或2维模型来实现，本申请实施例对此不加以限定。示例性的，根据控制虚拟对象的方式不同，可以将虚拟对象分为用户控制的虚拟对象和非用户控制的虚拟对象（如服务器控制的虚拟对象），

其中，用户控制的虚拟对象是由客户端控制的在虚拟环境中可活动的对象。服务器控制的虚拟对象是由客户端或服务器上的自动控制算法或人工智能程序控制的虚拟对象。服务器控制的虚拟对象包括虚拟环境中可活动对象和不可活动对象。示例性的，不可活动对象可以响应或影响可活动对象的活动，例如，可活动对象可以摧毁不可活动对象，或，当可活动对象进入不可活动对象时，可活动对象进入隐身状态。示例性的，本申请中的虚拟对象是由客户端控制的虚拟对象。示例性的，本申请中的虚拟对象可以是由其他客户端或服务器控制的虚拟对象。

MOBA 游戏是指：在虚拟环境中，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。该胜利条件包括以下至少之一：占领据点或摧毁敌对阵营据点、击杀敌对阵营的虚拟对象、在指定场景和时刻内保证自身的存活、抢夺到某种资源和在指定时刻内比分超过对方。战术竞技可以以局为单位来进行（即竞技对局），每局战术竞技的地图可以相同，也可以不同。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。在一些实施例中，MOBA 游戏是一种在虚拟环境中提供若干个据点，处于不同阵营的用户控制虚拟对象在虚拟环境中对战，占领据点或摧毁敌对阵营据点的游戏。例如，MOBA 游戏可将用户分成两个敌对阵营，将用户控制的虚拟对象分散在虚拟环境中互相竞争，以摧毁或占领敌方的全部据点作为胜利条件，一局 MOBA 游戏的持续时刻是从游戏开始的时刻至达成胜利条件的时刻。

UI (User Interface, 用户界面) 控件，用户界面中所展示的供用户进行人机交互的控件，如用户可通过 UI 控件对虚拟对象进行控制、通过 UI 控件对客户端进行功能实现等。其可是在应用程序的用户界面上能够看见的任何可视控件或元素，比如图片、输入框、文本框、按钮、标签等控件，其中一些 UI 控件可响应用户的操作，比如用于控制虚拟对象释放技能的技能控件，用户通过触发技能控件，可控制虚拟对象释放技能。本申请实施例中涉及的 UI 控件包括以下至少之一：技能控件、位移控件、系统功能控件和进化控件（用于触发将一个技能进化为另一个技能）。

冷却时间 (Cool Down Time, 简称 CD)：也可称为技能冷却时间，其为可以使用技能的时间间隔，即释放一次技能后再次释放该技能需要等待的时间。

技能点：用来管理技能释放次数的资源。如， n （如 2、3 等）个技能点可用于指示释放 n 次技能（ n 次技能可以不相同），在一些实施例中，技能点随着时间推进而生成，可以存储若干个。在一些实施例中，技能点消耗完毕之后，技能才进入完整的冷却时间，在有技能点的情况下，技能的冷却时间发生改变。

请参考图 1，其示出了本申请一个实施例提供的方案实施环境的示意图。该方案实施环境可以实现成为虚拟对象的控制系统。该方案实施环境可以包括：终端设备 10 和服务器 20。

终端设备 10 可以是诸如手机、平板电脑、游戏主机、电子书阅读器、多媒体播放设备、可穿戴设备、PC (Personal Computer, 个人计算机)、车载终端、智能机器人等电子设备。终端设备 10 中可以安装目标应用程序（如游戏应用程序）的客户端。可选地，该目标应用程序可以是需要下载安装的应用程序，也可以是即点即用的应用程序，本申请实施例对此不作限定。

在本申请实施例中，上述目标应用程序可以是仿真程序、大逃杀射击游戏、虚拟现实 (Virtual Reality, 简称 VR) 应用程序、增强现实 (Augmented Reality, 简称 AR) 程序、三维地图程序、虚拟现实游戏、增强现实游戏、第一人称射击游戏 (First-Person Shooting Game, 简称 FPS)、多人枪战类生存游戏、第三人称射击游戏 (Third-Person Shooting Game, 简称 TPS)、多人在线战术竞技游戏、策略游戏 (Simulation Game, 简称 SLG)、社交类应用程序、互动娱乐类应用程序中的任意一种。在本申请实施例中，以该目标应用程序是 MOBA 游戏来举例说明。另外，对于不同的应用程序来说，其所提供的虚拟对象的形态也会有所不同，且相应的

功能也会有所不同，这都可以根据实际需求进行设计，本申请实施例对此不作限定。可选地，终端设备 10 中安装运行有上述目标应用程序的客户端。

其中，上述虚拟环境可以是目标应用程序（如游戏应用程序）的客户端在终端设备上运行时显示（或提供）的场景。以竞技对局为例，该虚拟环境是指营造出的供虚拟对象进行活动（如竞技）的场景，如虚拟房屋、虚拟岛屿、虚拟地图等。

上述虚拟对象可以是指用户帐号在目标应用程序中控制的虚拟角色、虚拟载具、虚拟物品等等，本申请实施例对此不作限定。以目标应用程序为 MOBA 游戏应用程序为例，虚拟对象可以是指用户帐号在游戏应用程序中控制的游戏角色。虚拟对象可以是人物形态，可以是动物、卡通或者其它形态，本申请实施例对此不作限定。虚拟对象可以三维形式展示，也可以二维形式展示，本申请实施例对此不作限定。可选地，当虚拟环境为三维虚拟环境时，虚拟对象是基于动画骨骼技术创建的三维立体模型。每个虚拟对象在三维虚拟环境中具有自身的形状和体积，占据三维虚拟环境中的一部分空间。虚拟对象的活动包括以下至少之一：调整身体姿态、爬行、步行、奔跑、骑行、飞行、跳跃、驾驶、拾取、射击、攻击和投掷。示意性的，虚拟对象是虚拟人物，比如仿真人物角色或动漫人物角色。

服务器 20 用于为终端设备 10 中安装运行的目标应用程序的客户端提供后台服务。例如，服务器 20 可以是上述游戏应用程序的后台服务器。服务器 20 可以是一台服务器，也可以是由多台服务器组成的服务器集群，或者是一个云计算服务中心。可选地，服务器 20 同时为多个终端设备 10 中的目标应用程序提供后台服务。

在一些实施例中，终端设备 10 包括第一终端设备 11（图中未示出）和第二终端设备 12（图中未示出），其中第一终端设备上安装有目标应用程序的第一客户端，第二终端设备上安装有目标应用程序的第二客户端，第一用户的第一用户帐号在目标应用程序的第一客户端中控制第一虚拟对象，第二用户的第二用户帐号在目标应用程序的第二客户端中控制第二虚拟对象。可选地，第一虚拟对象和第二虚拟对象处于同一虚拟环境中。可选地，第一虚拟对象和第二虚拟对象可以属于同一个阵营、同一个队伍、同一个组织、具有好友关系或具有临时性的通讯权限。可选地，第一虚拟对象和第二虚拟对象可以属于不同的阵营、不同的队伍、不同的组织或具有敌对关系。可选地，第一终端设备和第二终端设备上安装的客户端是相同的，或两个终端上安装的客户端是不同操作系统平台下的。第一终端设备可以泛指多个终端设备中的一个，第二终端设备可以泛指多个终端设备中的另一个，本实施例仅以第一终端设备和第二终端设备来举例说明。

终端设备 10 和服务器 20 之间可通过网络进行互相通信。

结合上述对虚拟环境的介绍以及方案实施环境的说明，对本申请实施例提供的虚拟对象的控制方法进行说明，以该方法的执行主体为图 1 所示出的终端设备上运行的客户端来举例说明。该终端设备安装运行有上述目标应用程序，该目标应用程序可支持虚拟环境。

示例性的，以本申请提供的虚拟对象的控制方法应用在 MOBA 游戏中为例。

首先，对 MOBA 游戏中的技能升级机制进行说明。MOBA 游戏中的虚拟对象具有角色等级，虚拟对象可以通过获得经验值来升级角色等级，经验值可以根据对局时间、虚拟对象击杀其他虚拟对象、虚拟对象击杀野怪等来获得。虚拟对象的角色等级每升一级，就可以获得一次技能升级机会，虚拟对象可以对技能进行升级。示例性的，以虚拟对象具有三个技能为例进行说明，在对局刚开始时，虚拟对象的角色等级为 1 级，此时虚拟对象具有一次技能升级机会，用户可以从第一技能和第二技能中任选一个进行升级，第一技能或第二技能升级后，虚拟对象获得该技能。示例性的，在对局开始时，用户只有对技能进行一次升级，才能使虚拟对象获得该技能，使虚拟对象可以使用这个技能，即，1 级的虚拟对象只可以使用一个技能，2 级的虚拟对象如果将两次升级机会分别用在两个技能上，则虚拟对象可以使用两个技能，若两次升级机会都使用在同一个技能上，则虚拟对象仍只可以使用一个技能。示例性

的,虚拟对象在每个等级获得的技能升级机会只能用于升级指定技能中的一个,例如,1-3级可以升级一技能、二技能中的一个,4级可以升级一技能、二技能、三技能中的一个。当虚拟对象对技能进行第一次升级,获得该技能后,虚拟对象还可以继续对该技能进行升级,升级可以提高该技能某些属性的数值,例如,升级可以提高技能的作用时长、作用距离、作用范围、攻击值、防御值、护盾数值、虚拟对象模型大小中的至少一种。例如,一技能在升级前可以对命中的虚拟对象造成100点物理伤害并使其击晕3秒,在升级后可以对命中的虚拟对象造成200点物理伤害并使其击晕6秒。即,升级只会在技能原有的技能机制的基础上,更改技能机制的数值大小,不会为技能带来新的属性。

在一种可选的实施例中,虚拟对象的技能具有技能等级,技能每进行一次升级可以提升一个技能等级。当虚拟对象拥有四个技能时,虚拟对象的一技能、二技能、三技能的技能等级最高为4级,即,可以升级四次,四技能的技能等级最高为3级,即,可以升级三次。当虚拟对象拥有三个技能时,虚拟对象的一技能、二技能的技能等级最高为6级,即,可以升级六次,三技能的技能等级最高为3级,即,可以升级三次。当每个技能的技能等级大于或等于1时,虚拟对象获得该技能;当技能的技能等级小于1时,虚拟对象无法使用该技能,且该技能对应的技能控件显示为灰态,灰态的UI控件无法接收用户的触发操作。

在技能升级机制的基础上,本实施例提供了一种技能进化机制,技能进化机制可以将虚拟对象的技能进化为另一个技能,例如,虚拟对象在进化前的一技能是第一技能,在对一技能进行进化后,一技能由第一技能变为第二技能。第一技能和第二技能可以是完全不同的两个技能;第二技能也可以对第一技能进行改进,增加新的技能机制后得到的技能。例如,一技能在进化前可以对命中的虚拟对象造成100点物理伤害并使其击晕3秒,在进化后可以对命中的虚拟对象造成200点物理伤害、使其击晕6秒并降低其50%的防御值3秒。

在技能进化机制的基础上,本申请实施例提供了一种技能的技能组机制。技能组机制是一个技能控件对应多个技能。可选地,一个技能控件对应至少两个技能,在不同的触发条件下可以控制虚拟对象释放不同的技能。例如,同样是响应于针对第一技能控件的触摸或者滑动操作,当虚拟对象满足第一条件时,则释放的是第一技能控件对应的技能组中的第一个技能,当虚拟对象不满足第一条件时,则释放的是第一技能控件对应的技能组中的第二个技能。可选地,第一条件是第一虚拟对象的第一技能控件对应的第二个技能命中第二虚拟对象,则当再次触发对第一技能控件时,第一虚拟对象释放第一技能控件对应的第一个技能。在另一些实施例中,第一技能控件对应的多个技能是提前设置好的,当第一次触发第一技能控件时,则释放第一技能控件对应的第一个技能,当第二次触发第一技能控件时,则释放第一技能控件对应的第二个技能,当第三次触发第一技能控件时,则释放第一技能控件对应的第三个技能。本申请实施例对于技能控件对应的技能组的构成类型不作限定。

可选地,技能组中的多个技能是相同属性的,例如都是防守技能,一个技能组中包括多个不同的防守技能。可选地,技能组中的多个技能是不同属性的,例如一个技能组中有防守属性的技能,也有攻击属性的技能,如一个技能组中有两个攻击属性的技能以及一个防守属性的技能。也即,当第一技能控件被触发时,客户端可以控制虚拟对象释放第一技能控件对应的技能组中的攻击属性的技能,也可以控制虚拟对象释放第一技能控件对应的技能组中的防守属性的技能,当然该技能组中不同的技能的触发条件不同。

在技能组机制的基础上,本申请实施例提供了一种技能的组合机制。在本申请实施例中,提供了一种用来管理虚拟对象的技能释放次数的资源——技能点。在一些实施例中,可以存储若干个技能点。参考图2,用户界面100上显示有虚拟对象101、第一技能控件102、第二技能控件103、第三技能控件104,以及用于指示虚拟对象101拥有的三个技能点的技能点集合105。其中,技能点集合105中的技能点是可以被消耗的,当虚拟对象101释放第一技能控件102对应的技能时,消耗一个技能点,可选地,当虚拟对象101释放第二技能控件103对应的技能时,也消耗技能点。可选地,当虚拟对象101释放第三技能控件104对应的技能

时，不会消耗技能点（也即技能点为第一技能控件 102 和第二技能控件 103 所共用，并不被第三技能控件 104 所共用）。在技能点集合 105 中的三个技能点都消耗结束之后，用户无法对第一技能控件 102 以及第二技能控件 103 执行触发操作，即第一技能控件 102 以及第二技能控件 103 处于不可选中状态，同时虚拟对象 101 也无法释放第一技能控件 102 对应的技能以及第二技能控件 103 对应的技能。但是，虚拟对象 101 可以响应于用户对于第三技能控件 104 的触发操作，释放第三技能控件 104 对应的技能。

在本申请实施例中，技能点集合 105 中的技能点在消耗完之后，还可以重新恢复。在一些实施例中，一个技能控件对应一个技能组，假设第一技能控件 102 对应的技能组中有三个技能，分别是 A1、A2、A3，第二技能控件 103 对应的技能组中有三个技能，分别是 B1、B2、B3，第三技能控件 104 对应技能 C。对于有三个技能点的虚拟对象 101 来说，可以响应于用户的触发操作，释放不同的技能。可选地，在有三个技能点的情况下，若用户对第一技能控件 102 或者第二技能控件 103 执行触发操作，则虚拟对象 101 释放 A1 技能或者 B1 技能。当第一技能点被消耗之后，只剩第二个技能点和第三个技能点的情况下，若用户继续对第一技能控件 102 或者第二技能控件 103 执行触发操作，则虚拟对象 101 释放 A2 技能或者 B2 技能。当第二技能点消耗被消耗之后，只剩第三技能点的情况下，若用户继续对第一技能控件 102 或者第二技能控件 103 执行触发操作，则虚拟对象 101 释放 A3 技能或者 B3 技能。因此，对于有三个技能点的虚拟对象 101 来说，第一技能控件 102 和第二技能控件 103 对应的释放的技能组合可以是 A1A2A3、B1A2A3、A1B2A3、A1A2B3、B1B2A3、B1A2B3、A1B2B3 和 B1B2B3 中的至少一种。

本申请实施例对技能点集合 105 中的技能点的数量不作限定，对于消耗技能点以释放技能的组合形式也不作限定，以及对于消耗技能点的技能所对应的技能控件的数量不作限定，如三个技能控件所对应的技能组中的技能在释放时都消耗技能点，也可以是其中的一个或者两个。参考图 3，响应于用户针对第一技能控件 102 的触发操作，客户端控制虚拟对象 101 释放第一技能控件 102 对应的技能，此时技能点集合 105 中的技能点数量变为两个，并且在虚拟对象 101 释放完技能之后，第一技能控件 102 进入冷却状态。参考图 4，在技能点集合 105 中的技能点均被消耗时，第一技能控件 102 和第二技能控件 103 都进入了冷却状态（也可称为不可选中状态）。

相关技术中，也提供了一些技能的组合方式。在一些实施例中，在如图 5 所示的用户界面 500 的示意图中，第一虚拟对象 501 在第二技能控件 502 对应的第一个技能命中第二虚拟对象（图中未示出）时，则刷新第二技能控件 502 的冷却时间，当再次触发第二技能控件 502 时，则控制虚拟对象 501 释放第二技能控件 502 对应的第二个技能，其中第二技能控件 502 对应的第一个技能和第二个技能是不同的技能。在满足一定条件下，才可以控制虚拟对象 501 释放第二个技能，虚拟对象 501 的第二个技能和第一个技能不同，可选地，第二个技能造成的攻击值提升 25%。

在另一些实施例中，如图 6 所示的用户界面 600 的示意图中，第四个技能图标 602 处于不可使用状态。可选地，在虚拟对象 601 释放了第一个技能图标 603 对应的第一个技能时，第四个技能图标 602 对应的第四个技能可以复制第一个技能，则第四个技能图标 602 对应的技能是第一个技能图标对应的第一个技能。根据虚拟对象 601 释放的技能不同，第四个技能图标 602 可以复制的技能不同。

在另一些实施例中，以某种条件刷新技能的冷却时间，如利用游戏内已有的其他交互条件达成时，去刷新特定技能，来实现技能的多次释放。这些条件可以是：某个技能命中目标、战胜目标、释放了多次技能、命中带有特定标记的目标等等。如图 7 所示的用户界面 700 的示意图中，第一虚拟对象 701 释放了第三技能，并且该第三技能命中了第二虚拟对象 702，则在第二虚拟对象 702 的头顶上方显示用于指示被标记的标识 703，同时基于第二虚拟对象 702 被标记，将第一虚拟对象 701 的第三个技能的技能冷却时间调整为 0，也即第一虚拟对象

701 可以立刻响应于用户对第三技能控件 704 的触发操作，释放第三个技能。

在另一些实施例中，如图 8 所示的用户界面 800 的示意图中，第三技能控件 801 被放大显示，同时储存多个第三技能，也即可以响应于用户对于第三技能控件 801 的触发操作，虚拟对象可连续释放多次第三技能。在另一些实施例中，以技能豆的形式存储某个技能。即设定一种技能豆作为某个技能的释放次数的资源，在技能豆没有消耗完毕的情况下，虚拟对象可以多次释放该技能，且技能不进入 CD 或者不进入完整 CD。如图 9 所示的用户界面 900 的示意图中，第二技能控件 901 上显示有数字 3，说明储存了三个第二技能，也即可以响应于用户对于第二技能控件 901 的连续触发操作，客户端控制虚拟对象连续释放多次第二技能。

在相关技术中，技能的组合相对比较简单，一种是在满足条件下出现技能控件对应的另一个技能（实质为一个技能对应的两种阶段），但是如果在阈值时间内不释放该技能，则此次无法再继续释放另一个技能。或者，以技能豆的形式储存技能，但是该技能的存储，针对的是同一个技能，也即可以控制虚拟对象连续多次释放相同的技能，而不能释放不同的技能（实质为一个技能对应的不同阶段）。因此相关技术中对于技能的组合相对来说比较简单。

不同的是，本申请实施例中提出技能点的概念，在一些实施例中，在用户界面上有三个技能控件，其中响应于针对第一技能控件或者第二技能控件的操作，消耗技能点，才能控制虚拟对象释放技能。在另一些实施例中，消耗技能点的技能控件对应有技能组，技能组中多个技能。可选地，技能点的总数为 3 个，技能组中的技能也有三个并且各不相同，技能组中的技能和技能点相对应。因此，对于不同的技能点数，用户对于技能控件的操作不同，可以使得虚拟对象释放更多不同的技能组合。

请参考图 10，其示出了本申请一个实施例提供的虚拟对象的控制方法的流程图。该方法各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是上述目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤(1001~1003)中的至少一个步骤：

步骤 1001，在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能。

用户界面是指上述客户端的用户界面，其可用于内容展示、提供人机交互服务等。示例性地，该用户界面可以是指上述竞技对局的用户界面，其可用于显示竞技对局对应的虚拟环境画面。可选地，用户界面可以包括显示层和控件层，控件层位于显示层的上层，显示层用于显示内容，控件层用于部署 UI 控件。

技能控件：用于释放技能的控件，如在用户界面上显示的用于响应于用户的操作而控制虚拟对象释放技能的 UI 控件。本申请对于技能控件的数量限定为至少两个，如此可以增加技能之间的组合方式，增加竞技的趣味性。技能控件的样式、布局、尺寸等可以根据实际使用需求进行设置与调整。

技能组：是指技能的集合，如由多个不同的技能所组合得到的集合。技能组中的各个技能共用同一个技能控件，如在不同的条件下，响应于用户针对同一个技能控件的操作，客户端控制虚拟对象释放不同的技能。可选地，本申请实施例以当前的技能点来指示技能控件当前所对应的技能。例如，针对不同技能点的选择，各个技能控件对应的技能不同；又例如，针对技能点的数量不同，各个技能控件对应的技能不同，本申请实施例对此不作限定。

技能点集合：由技能点构成的集合，技能点集合中的技能点的数量会发生改变。示例性的，用户每消耗一个技能点，技能点集合中的技能点就会减少。可选地，当技能点都消耗完毕时，技能点集合中没有技能点。在一些实施例中，技能点集合中的技能点可以随着时间恢复。技能点的显示位置、显示样式也可以根据实际使用需求进行设置与调整。

在一些实施例中，至少两个技能控件中，存在至少一个技能控件，其对应的技能组中包

括至少两个技能。在另一些实施例中，每个技能组中包括至少两个技能。本申请对此不作限定。

在一些实施例中，上述至少两个技能控件对应的技能组中的第一技能组包括至少两个技能，第一技能组中包括的不同技能对应于不同的技能点。

该第一技能组可以是指至少两个技能组中的任一技能组。可选地，第一技能组中的技能的数量可以与技能点的数量相对应，如技能点集合中包括 3 个技能点，则第一技能组中可以包括 3 个技能。第一技能组中的技能的数量可以小于技能点的数量，如技能点集合中包括 3 个技能点，则第一技能组中可以包括 2 个技能。

示例性的，第一技能组中有技能 a1 和技能 a2，技能 a1 和技能 a2 对应于不同的技能点。例如，技能集合中总共有两个技能点，分别是技能点 1 和技能点 2。可选地，技能点 1 对应技能 a1，技能点 2 对应技能 a2，技能点 1 可用于使得第一技能组对应的技能控件用于释放技能 a1，而技能点 2 可用于使得第一技能组对应的技能控件用于释放技能 a2，且技能点 2 与技能 a1 不对应，技能点 1 与技能 a2 不对应。

可选地，技能控件当前对应的技能也可以与技能点集合中的当前技能点数量相关，如在技能点集合中的当前技能点数量为 2 的情况下，技能控件对应技能 a1；在技能点集合中的当前技能点数量为 1 的情况下，技能控件对应技能 a2。

在一些实施例中，一个技能组中的各个技能是不同的。可选地，一个技能组中的各个技能都是相同属性的，例如，一个技能组中的各个技能都是防御属性的，或者一个技能组中的各个技能都是攻击属性。可选地，一个技能组中的技能是不同属性的，例如，一个技能组中既有攻击属性的技能，也有防御属性的技能。

示例性的，一个技能组中相同属性的技能之间存在攻击值的区别，例如技能 a1 造成的攻击值是 100，技能 a2 造成的攻击值是 200。在一些实施例中，一个技能组中相同属性的技能之间存在防御值的区别，例如技能 a1 给虚拟对象的防御值是 100，技能 a2 给虚拟对象的防御值是 200。在另一些实施例中，一个技能组中的不同技能可以对应虚拟对象的不同形态，例如技能 a1 可以使得虚拟对象处于第一形态，技能 a2 可以使得虚拟对象处于第二形态，第一形态和第二形态是两种不同的形态，具体的形态可以是人形态、动物形态、走形态、飞形态中的至少一种，本申请对此不作限定。

本申请实施例提供的技术方案，通过在技能组中设置多个技能，可以使得一个技能控件对应一个技能组，并且可以从该技能组中选择出一个技能作为该技能控件当前对应的技能。在一个技能控件对应一个技能组，并且该技能组中至少包括两个技能的情况下，用户可以根据实际情况，选择性地控制虚拟对象释放技能组中的某种技能，因此可以满足更多用户的需求，同时虚拟对象的控制方式更多样，技能释放的组合方式也更多样。

步骤 1002，响应于针对至少两个技能控件中的第一技能控件的操作，从至少两个技能点中确定将使用的第一技能点。

第一技能控件可以是至少两个技能控件中的任一技能控件。可选地，第一技能控件可以是用户以第一技能控件为目标，使用触摸设备或手指针对客户端所在的终端设备的显示屏所选中的，也可以是用户通过外接设备（如手柄、摇杆、鼠标、键盘等）针对第一技能控件进行操作所选中的，本申请实施例对此不作限定。

在一些实施例中，上述操作可以是点击操作、滑动操作、触摸操作等中的至少一种，本申请对此不作限定。如响应于用户针对第一技能控件的点击操作，客户端从至少两个技能点中确定将使用的第一技能点。

本申请实施例对第一技能点的确定方式不作限定。示例性的，可以随机从至少两个技能点中选择一个技能点作为将使用的第一技能点，也可以按照从前往后的顺序，确定当前将要使用的技能点，以及下一个将使用的技能点等，还可以按照从后往前的顺序，确定当前将要使用的技能点，以及下一个将使用的技能点等。可选地，该第一技能点也

可以由用户设置，如根据用户设置的顺序，依次确定各个操作下的第一技能点。

在一些实施例中，客户端根据技能点使用顺序，从至少两个技能点中确定将使用的第一技能点；其中，技能点使用顺序用于定义至少两个技能点的使用顺序。该技能点使用顺序可以用用户调整设置，也可以是应用程序预设的固定顺序。

可选地，若技能点使用顺序是从前往后/从左往右，则技能点集合中的技能点从前往后/从左往右依次被选定。例如，参考图 2，技能点集合 105 中包括三个技能点，技能点使用顺序是从左往右，则按照从左往右的顺序使用第一技能点之后，技能点集合 105 中最左边的技能点消失，技能点集合 105 更新为如图 3 所示的技能点集合 105（仅剩 2 个技能点）。本申请对技能点的使用顺序不作限定，除此之外，本申请对技能点的形式不作限定。在如图 2 或 3 所示的用户界面 100 的示意图中，技能点是以月亮的形式出现在虚拟对象 101 的血条上方，可选地，技能点也可以以太阳、怪兽、星星、数字、格子等其他形式来显示。

可选地，技能点可以完全显示在用户界面上，也可以不完全显示在用户界面上。例如，用户界面上仅显示有技能点控件而不显示具体的技能点的数量，需要用户点击技能点所在的技能点控件之后，才能完全展开所有的技能点。在另一些实施例中，技能点还可以完全不显示在用户界面中，例如在用户界面上完全不显示技能点的数量，但是可以存在语音播报，例如“用户 xxx 获得了 3 个技能点”。当用户使用技能点时，也存在播报“用户 xxx 使用了技能点 1，还剩 2 个技能点”。另外，本申请对技能点在用户界面中的显示位置不作限定，可选地，技能点在虚拟对象所在的虚拟环境（即显示层）中显示，可选地，技能点是出现在用户界面的空间层上，其可以被用户触控或点击。

本申请实施例中，通过根据技能点使用顺序确定下一个将使用的技能点，用户可以明确将使用的技能点，根据将使用的技能点而重新确定将控制虚拟对象释放的技能，方便用户实时了解竞技状态，同时对于新手用户也十分友好。

步骤 1003，控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能。

第一虚拟对象可以是指登录终端设备中客户端的当前用户帐号所控制的虚拟对象，对于当前终端设备，第一虚拟对象也可以被称为当前终端设备对应的当前用户界面中的主控虚拟对象，上文中已对虚拟对象作了详细的叙述，在此不作赘述。

每个技能组中的技能和技能点存在着对应关系。示例性的，技能组中的技能，按照顺序与技能点集合中的技能点一一对应；技能组中的技能与技能点之间的对应关系也可以由用户进行设置；技能组中的技能与技能点之间的对应关系也可以是随机的，本申请实施例对此不作限定。

例如，第一技能控件对应的技能组中有 3 个技能，技能点集合中也有 3 个技能点，响应于对第一技能控件的操作，若使用的是技能点集合中的第一个技能点，客户端则控制第一虚拟对象释放第一技能控件对应的技能组中的第一个技能。响应于对第一技能控件的操作，若使用的是技能点集合中的第二个技能点时，客户端则控制第一虚拟对象释放第一技能控件对应的技能组中的第二个技能。响应于对第一技能控件的操作，若使用的是技能点集合中的第三个技能点时，客户端则控制第一虚拟对象释放第一技能控件对应的技能组中的第三个技能。

在一些实施例中，用户界面显示至少两个技能点中，未使用的技能点，也即用户界面上将未使用的技能点显示出来，便于用户知道还有几个或者哪几个技能点未使用，或者已使用了哪些技能点，从而提高用户体验。例如，参考图 2，技能点集合 105 中包含的三个技能点就是未使用的技能点。又例如，图 3 中的技能点集合 105 所包含的两个技能点就是未使用的技能点。

本申请实施例提供的技术方案，通过显示技能点中未使用的技能点，可以有效给予用户以提示，告知用户还未使用的技能点，因此可以帮助用户更准确的判断需要释放何种技能，便于用户的操作。

在一些实施例中，客户端在技能控件的显示区域，显示与下一个使用的技能点相对应的技能的图标。示例性的，在使用了第一个技能点之后，若下一个将使用的技能点是第二个技能点，则第一技能控件对应的技能和第二技能控件对应的技能将会发生改变，如与第二个技能点相对应，则第一技能控件和第二技能控件更新显示变换后的技能的图标。在一些实施例中，下一个使用的技能点相对应的技能，图标会被显示，以使得用户知道下一个触发的技能是什么。可选地，下一个使用的技能点是指按照默认逻辑（既定顺序）确定的、下一个将要被使用的技能点。在另一些实施例中，若用户可以选择“将使用的技能点”，则下一个使用的技能点可以是用户所选择的“将使用的技能点”。本申请对于是否由用户选择“将使用的技能点”不作限定。

本申请实施例提供的技术方案，通过显示与下一个使用的技能点相对应的技能的图标，可以有效地告知用户下一个技能点对应的技能，便于用户从多个技能控件中作出选择，方便了竞技的操作，同时便于用户根据实际情况确定所要控制虚拟对象释放的技能，提高了用户体验。

在一些实施例中，在使用技能点释放技能的情况下，技能的冷却时长小于技能的额定冷却时长。

额定冷却时长相当于是技能的原始的冷却时长，也即在没有技能点参与的情况下的技能的冷却时长。可选地，第一技能控件对应的技能组的额定冷却时长是10s，则在用户使用技能点释放技能的情况下，技能的冷却时间可降至2s。在用户使用最后一个技能点的情况下，例如最后一个技能点对应的是用户释放了第一技能控件的技能组的第三个技能，则在该技能用完之后，该技能的冷却时长变为额定冷却时长。

本申请实施例提供的技术方案，在使用技能点释放技能的情况下，技能的冷却时长小于技能的额定冷却时长，通过使用技能点可以缩短技能的冷却时长（冷却时间），也即方便用户通过技能控件，在短时间内可以连续释放多个技能，技能之间的衔接性较好，用户体验感也更好，对于老手用户来说，更能体现其竞技实力，从而提高了操作连续性。

本申请实施例提供的技术方案，通过支持至少两个技能控件共用同一技能点集合，且当用户执行针对至少两个技能控件中的第一个技能控件的操作时，从技能点集合中确定出将使用的技能点，同时控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能，实现了通过技能点来确定当前所触发的技能控件对应的当前技能。因此，在有至少两个技能点、至少两个技能控件的情况下，根据不同的技能点选择和不同技能控件的选择，技能的释放有多种可能的组合方式，从而实现技能释放的组合多样化，进而提升技能释放的组合灵活性和多样性。

另外，通过采用本申请实施例提供的技术方案，仅需对技能组中的技能数量或技能点集合中的技能点数量进行调整，即可在不增加多余实体（如技能控件）的情况下，提升技能释放的组合多样性，从而提高了本申请实施例提供的技术方案的可扩展性。另外，通过采用本申请实施例提供的技术方案，也可对用户界面进行优化，如利用技能点减少技能控件数量，从而有利于提高用户界面的利用率。

请参考图11，其示出了本申请另一个实施例提供的虚拟对象的控制方法的流程图。该方法各步骤的执行主体可以是图1所示方案实施环境中的终端设备10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（1101~1103）中的至少一个步骤：

步骤1101，在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能。

步骤1101与步骤1103与上述实施例介绍相同，本申请实施例未说明的内容可以参考上

述实施例，这里不再赘述。

步骤 1102，响应于针对至少两个技能控件中的第一技能控件的操作，从至少两个技能点中，将技能点选择操作所选择的技能点确定为第一技能点；其中，技能点选择操作是指用户执行的用于选择技能点的操作。

本申请实施例对于技能点选择操作的类型不作限定，其可以是滑动、点击、双击中的至少一种。

在一些实施例中，技能点选择操作和针对第一技能控件的操作为一步滑动操作；其中，一步滑动操作是指一次不间断的滑动操作，也即初始位置和终止位置明确，且从而初始位置到终止位置一步到位的滑动操作。

可选地，该一步滑动操作的起始位置所对应的技能控件为第一技能控件，一步滑动操作的终止位置所对应的技能点为第一技能点。例如，用户界面中显示有技能控件 1 和技能控件 2，以及技能点 1 和技能点 2，响应于针对技能控件 1 到技能点 1 的一步滑动操作，客户端将技能控件 1 确定为第一技能控件，将技能点 1 确定为第一技能点。

或者，该一步滑动操作的起始位置所对应的技能点为第一技能点，一步滑动操作的终止位置所对应的技能控件为第一技能控件。例如，用户界面中显示有技能控件 1 和技能控件 2，以及技能点 1 和技能点 2，响应于针对技能点 1 到技能控件 1 的一步滑动操作，客户端将技能控件 1 确定为第一技能控件，将技能点 1 确定为第一技能点。

一方面，可以通过从第一技能控件滑动到技能点，另一方面可以从技能点滑动至技能控件，这两种方式都可用于选择当前操作所对应的技能点，在确定出技能点之后，可以根据使用的技能点，来确定技能控件对应的技能组中的当前技能。

在一些实施例中，可以分别点击技能控件选择技能，以及点击技能点选择技能点，来实现选择哪一个技能点的目的，然而这样操作效率低，会影响技能释放效率。即便对于高水平玩家，尝试通过两个手指同时点击，那也是对操作有较高要求。不同的是，本申请实施例提供的技术方案仅需一步滑动操作就可完成技能和技能点的选择，非常高效，且操作难度低。

步骤 1103，控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能。

可选地，在第一技能点确定之后，第一技能控件当前对应的技能也随之确定，此时针对第一技能控件的触发操作，等效于第一技能控件当前对应的技能的触发操作。

本申请实施例提供的技术方案，通过将技能点设置在控件层上，并且用户可以对技能点执行操作，当用户在选择了技能点之后，再执行对于技能控件的操作，可以实现某个技能组中对应于该技能点的技能的触发。也正是因为，不同的技能点对应技能组中的不同技能，因此当用户可以选择对应的技能点时，也就可以有选择的释放技能组中的技能，用户可以根据瞬息万变的局势，来选择使用不同的技能点。对于新手用户来说，要多一步对技能点的选择操作可能相对较为困难，但是相反的，对于老手用户来说，提前对技能点进行选择，更能体现用户的反应速度和竞技实力，同时技能点选择操作可以让用户明确需要控制虚拟对象释放的技能，方便用户根据对局情况选择不同的技能。在本申请实施例提供的技术方案中，技能点没有了固定的使用顺序，技能点的使用依靠用户的技能点选择操作，丰富了虚拟对象的控制方式，提升了竞技对局的趣味性和挑战性。

请参考图 12，其示出了本申请另一个实施例提供的虚拟对象的控制方法的流程图。该方法各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（1201~1203）中的至少一个步骤：

步骤 1201，在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个

技能。

步骤 1201 与步骤 1203 与上述实施例介绍相同，本申请实施例未说明的内容可以参考上述实施例，这里不再赘述。

步骤 1202，响应于针对至少两个技能控件中的第一技能控件的操作，根据实时对局数据，从至少两个技能点中确定将使用的第一技能点。

其中，实时对局数据可以是指上述竞技对局的实时对局数据，如 MOBA 游戏中的竞技对局的实时对局数据。该实时对局数据可以实时反映竞技对局当前进度情况。

在一些实施例中，实时对局数据包括以下至少之一：第一虚拟对象最近 N 次释放的技能、第一虚拟对象的属性信息、第一虚拟对象正在对局的第二虚拟对象的属性信息、第一虚拟对象正在辅助的第三虚拟对象的属性信息，N 为正整数。正是由于竞技对局的形势的多变性，因此根据实时对局数据来从至少两个技能点中确定将使用的第一技能点，可以提高第一技能点的确定合理性。

示例性的，客户端可根据第一虚拟对象最近 N 次释放的技能，从至少两个技能点中确定将使用的第一技能点。例如，客户端根据第一虚拟对象最近 N 次释放的技能中对应的不同技能点的数量，确定第一虚拟对象在对局中的技能点偏好，如果第一虚拟对象偏好使用第一技能点对应的技能组中的技能，则优先从至少两个技能点中将偏好使用的技能点确定为将使用的第一技能点。

示例性的，客户端可根据第一虚拟对象的属性信息，从至少两个技能点中确定将使用的第一技能点。可选地，不同技能点对应的技能的属性不同，例如，技能点 1 对应的技能是防御型技能，技能点 2 对应的技能是攻击型技能。若第一虚拟对象的属性信息为第一虚拟对象的血量信息，则在第一虚拟对象的血量值较低的情况下，客户端可将至少两个技能点中的技能点 1 确定为将使用的第一技能点，此时技能点 1 对应的防御型技能可以提高第一虚拟对象的防御值。又例如，第一虚拟对象的血量信息十分健康，但是第一虚拟对象所处的位置很危险，比如第一虚拟对象周围有多个第二虚拟对象（此处第二虚拟对象为与第一虚拟对象处于不同阵营的虚拟对象），则客户端将至少两个技能点中的技能点 2 确定为将使用的第一技能点，此时技能点 2 对应的攻击型技能可以提高第一虚拟对象的攻击值。又例如，第一虚拟对象的属性信息还可以是第一虚拟对象身上具有的属性增益，例如第一虚拟对象身上本身具有了攻击的属性增益，此时需要提高防御力，则客户端将至少两个技能点中的技能点 1 确定为将使用的第一技能点，此时技能点 1 对应的防御型技能可以提高第一虚拟对象的防御值。又例如，第一虚拟对象身上本身具有了防御的属性增益，此时需要提高攻击力，则客户端将至少两个技能点中的技能点 2 确定为将使用的第一技能点，此时技能点 2 对应的攻击型技能可以提高第一虚拟对象的攻击值，如此可实现技能和属性增益之间的互补。

示例性的，客户端可根据第一虚拟对象正在对局的第二虚拟对象的属性信息，从至少两个技能点中确定将使用的第一技能点。例如，客户端根据第二虚拟对象的血量值、攻击值、防御值等确定至少两个技能点中将使用的第一技能点。

示例性的，客户端可根据第一虚拟对象正在辅助的第三虚拟对象的属性信息，从至少两个技能点中确定将使用的第一技能点。例如，第三虚拟对象是和第一虚拟对象处于同一阵营的虚拟对象，客户端可根据第三虚拟对象的血量值、攻击值、防御值等确定至少两个技能点中将使用的第一技能点。

步骤 1203，控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能。

本申请实施例提供的技术方案，通过根据实时对局数据来确定将使用的第一技能点，更能满足用户的需求，同时可以避免对技能点进行选择操作，如此对于新手用户十分友好，从而提高了用户的竞技体验感。

请参考图 13，其示出了本申请另一个实施例提供的虚拟对象的控制方法的流程图。该方法各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（1301~1304）中的至少一个步骤：

步骤 1301，在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能。

步骤 1302，响应于针对至少两个技能控件中的第一技能控件的操作，从至少两个技能点中确定将使用的第一技能点。

步骤 1303，控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能。

步骤 1301-步骤 1303 与上述实施例介绍相同，本申请实施例未说明的内容可以参考上述实施例，这里不再赘述。

步骤 1304，在技能点集合中未使用的技能点的数量小于额定最大数量的情况下，若满足技能点恢复条件，则增加技能点集合中未使用的技能点的数量。

额定最大数量是指技能点集合中所能包含的技能点的最大数量，其可以根据实际使用需求进行设置与调整。技能点恢复条件用于指示技能点是否能回复。

在一些实施例中，技能点集合中未使用的技能点是可以恢复到额定最大数量的。示例性的，额定最大数量是 3，若技能点集合中未使用的技能点的数量小于 3，客户端则判断是否满足恢复条件，在满足恢复条件的情况下，恢复技能点到额定最大数量。

本申请实施例对增加技能点集合中未使用的技能点的数量的方式不作限定，可以是一次性增加至额定最大数量，也可以是一个一个地增加技能点直至额定最大数量。

在一些实施例中，技能点恢复条件包括以下至少之一：距离上一次使用技能点达到第一时长、距离上一次恢复技能点达到第二时长、第一虚拟对象对敌方虚拟对象产生满足第一条件的伤害、第一虚拟对象所属阵营对敌方虚拟对象产生满足第二条件的伤害、第一虚拟对象获得第一增益效果、第一虚拟对象所属阵营获得第二增益效果。

可选地，在竞技对局的一开始即恢复技能点集合中技能点的数量至额定最大数量。

在一些实施例中，技能点恢复条件是距离上一次使用技能点达到第一时长。例如，第一时长是 10s，在上一次使用技能点之后的 10s，用户都没有再次使用技能点，则技能点的数量恢复至额定最大数量。

在一些实施例中，技能点恢复条件是距离上一次恢复技能点达到第二时长。例如，第二时长是 60s，在上一次恢复技能点之后，技能点的数量经过 60s 自动恢复至额定最大数量。

在一些实施例中，技能点恢复条件是第一虚拟对象对敌方虚拟对象产生满足第一条件的伤害。可选地，第一条件是第一虚拟对象对于敌方虚拟对象造成的伤害大于第一伤害阈值，或者第一虚拟对象战胜的敌方虚拟对象的数量大于第一数量阈值，若满足，则技能点的数量恢复至额定最大数量。例如，第一数量阈值是 5，若第一虚拟对象战胜的敌方虚拟对象的数量大于 5 个，则技能点的数量恢复至额定最大数量。

在一些实施例中，技能点恢复条件是第一虚拟对象所属阵营对敌方虚拟对象产生满足第二条件的伤害。可选地，第二条件是第一虚拟对象所属阵营对于敌方虚拟对象造成的伤害大于第二伤害阈值，或者第一虚拟对象所属阵营战胜的敌方虚拟对象的数量大于第二数量阈值，若满足，则技能点的数量恢复至额定最大数量。例如，第二数量阈值是 5，若第一虚拟对象所属阵营对战胜的敌方虚拟对象的数量大于 5 个，则技能点的数量恢复至额定最大数量。

在一些实施例中，技能点恢复条件是第一虚拟对象获得第一增益效果。可选地，第一增益效果是攻击效果或者防御效果中的至少一种，本申请实施例对此不作限定。例如，若第一增益效果是攻击效果，则当第一虚拟对象获得攻击效果时，技能点的数量恢复至额定最大数

量。

在一些实施例中，技能点恢复条件是第一虚拟对象所属阵营获得第二增益效果。可选地，第二增益效果是攻击效果或者防御效果中的至少一种，本申请实施例对此不作限定。例如，若第二增益效果是防御效果，则当第一虚拟对象获得防御效果时，则技能点的数量恢复至额定最大数量。

本申请实施例提供的技术方案，在技能点集合中未使用的技能点的数量小于额定最大数量的情况下，若满足技能点恢复条件，则增加技能点集合中未使用的技能点的数量。同时技能点恢复条件也是多样的，根据不同的需求，用户可以设置不同的技能点恢复方式，另外，技能点的快速恢复也有利于加快对局的进度，减少设备的处理开销。

请参考图 14，其示出了相关技术一个实施例提供的虚拟对象的控制方法的框图。该方法可以包括如下几个步骤（S1~S6）中的至少一个步骤：

步骤 S1，请求释放技能。

服务器请求终端设备的客户端释放技能，客户端接收到技能释放请求之后，会执行步骤 S2-S3，释放所选择的指定技能槽位的技能逻辑。

步骤 S2，选择指定技能槽位。

本申请实施例中的技能槽位也可认为是技能控件。

步骤 S3，执行技能逻辑。

在逻辑执行完毕后，客户端会检查该技能是否要切换到其他的技能。如果需要进行切换，客户端则会将该技能槽位里的具体技能对象进行切换，来保证下次释放时是不同的技能逻辑。

步骤 S4，是否切换技能。

若是，则执行步骤 S5，若否，则执行步骤 S6。

步骤 S5，切换技能至目标技能。目标技能可以是指当前技能在指定技能槽位中的下一技能。

步骤 S6，结束。

相关技术中的技能释放逻辑存在着目标技能需要明确，以及技能释放的顺序需要固定的问题，而本申请实施例提供的技术方案对于技能释放的顺序不作限定，在存在技能点的情况下，即可随机组合技能，对于技能释放顺序不作限定，丰富了虚拟对象释放技能的方式。

请参考图 15，其示出了本申请一个实施例提供的虚拟对象的控制方法的框图。该方法各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（P1~P6）中的至少一个步骤：

步骤 P1，请求释放技能。

步骤 P2，选择指定技能槽位。

步骤 P3，执行技能逻辑。

步骤 P4，发送技能切换事件。

步骤 P5，接收事件。

步骤 P6，执行技能切换逻辑。

本申请实施例提供的技能切换逻辑是一种根据具体情况下切换不同逻辑的功能，也即技能点使用（或某个技能的技能逻辑执行完）之后，客户端即可直接对各个技能控件所对应的技能进行切换。所以需要将技能切换逻辑脚本化，配置化。具体的技能切换逻辑从之前的单纯检查一个槽位是否需要切换技能，变成了检测条件是否满足，并且同时切换用户界面中的各个指定技能槽位的技能。

请参考图 16，其示出了本申请另一个实施例提供的虚拟对象的控制方法的框图。该方法

各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（Q1~Q3）中的至少一个步骤：

步骤 Q1，事件判断。

在一些实施例中，通过步骤 Q1 确定当前所使用的技能点是技能点集合中的具体哪一个。

步骤 Q2，执行技能切换逻辑。

根据确定出来的技能点来执行技能切换逻辑，即将技能控件对应的技能切换为当前技能点对应的技能。

步骤 Q3，输出切换结果。

本申请实施例提供的技术方案，可以将技能切换逻辑进行配置化，不同的虚拟对象在不同的情况下可以调整技能切换逻辑的具体内容。

请参考图 17，其示出了本申请一个实施例提供的技能切换逻辑的框图。该方法各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（W1~V5）中的至少一个步骤：

步骤 W1，选择第一技能槽位。

判断是否消耗技能点 1，若是，则执行步骤 W2，若不是，则判断是否消耗技能点 2。

本申请实施例中，使用技能点也可认为是消耗技能点。

步骤 W2，切换第一技能槽位对应的第一技能。

判断是否消耗技能点 2，若是，则执行步骤 W3，若不是，则判断是否消耗技能点 3。

步骤 W3，切换第一技能槽位对应的第二技能。

判断是否消耗技能点 3，若是，则执行步骤 W4，若不是，则执行步骤 W5。

步骤 W4，切换第一技能槽位对应的第三技能。

步骤 W5，无法释放第一技能槽位对应的技能。

步骤 V1，选择第二技能槽位。

判断是否消耗技能点 1，若是，则执行步骤 V2，若不是，则判断是否消耗技能点 2。

步骤 V2，切换第二技能槽位对应的第一技能。

判断是否消耗技能点 2，若是，则执行步骤 V3，若不是，则判断是否消耗技能点 3。

步骤 V3，切换第二技能槽位对应的第二技能。

判断是否消耗技能点 3，若是，则执行步骤 V4，若不是，则执行步骤 V5。

步骤 V4，切换第二技能槽位对应的第三技能。

步骤 V5，无法释放第二技能槽位对应的技能。

本申请实施例提供的技术方案，通过给不同的技能点以编号，可以设计不同的技能下对应的不同的技能，提供了一种新的技能切换逻辑，便于根据使用的技能点来确定需要切换的技能，也即当前技能点被消耗，技能槽位对应的技能则需要更新至下一技能点对应的技能。

请参考图 18，其示出了本申请一个实施例提供的 CD 切换逻辑的框图。该方法各步骤的执行主体可以是图 1 所示方案实施环境中的终端设备 10，如各步骤的执行主体可以是目标应用程序的客户端。在下文方法实施例中，为了便于描述，仅以各步骤的执行主体为“客户端”进行介绍说明。该方法可以包括如下几个步骤（U1~U3）中的至少一个步骤：

步骤 U1，请求释放技能。

在技能释放之后，判断是否还有技能点，若是，则执行步骤 U2，若不是，则执行步骤 U3。

在每次释放完技能之后，都去检测技能点集合中是否还有技能点，如果还有技能点则该释放的技能不进入该技能的完整 CD（额定 CD）。

步骤 U2，重新设置第一技能槽位以及第二技能槽位的技能 CD。

可选地，将所有槽位对应的技能 CD 都调整为 0，或者调整到小于额定 CD。
步骤 U3，恢复第一技能槽位以及第二技能槽位对应技能的额定 CD。

下述为本申请装置实施例，可以用于执行本申请方法实施例。对于本申请装置实施例中未披露的细节，请参照本申请方法实施例。

请参考图 19，其示出了本申请一个实施例提供的虚拟对象的控制装置的框图。该装置具有实现上述方法示例的功能，所述功能可以由硬件实现，也可以由硬件执行相应的软件实现。该装置可以是上文介绍的终端设备，也可以设置在终端设备中。如图 19 所示，该装置 1800 可以包括：显示模块 1810、技能点确定模块 1820 和控制模块 1830。

所述显示模块 1810，用于在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，所述技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能。

所述技能点确定模块 1820，用于响应于针对所述至少两个技能控件中的第一技能控件的操作，从所述至少两个技能点中确定将使用的第一技能点。

所述控制模块 1830，用于控制第一虚拟对象释放所述第一技能控件对应的技能组中，与所述第一技能点相对应的技能。

在一些实施例中，所述至少两个技能控件对应的技能组中的第一技能组中包括至少两个技能，所述第一技能组中包括的不同技能对应于不同的技能点。

在一些实施例中，所述显示模块 1810，还用于显示所述至少两个技能点中，未使用的技能点。

在一些实施例中，所述显示模块 1810，还用于在所述技能控件的显示区域，显示与下一个使用的技能点相对应的技能的图标。

在一些实施例中，所述技能点确定模块 1820，用于根据技能点使用顺序，从所述至少两个技能点中确定将使用的所述第一技能点；其中，所述技能点使用顺序用于定义所述至少两个技能点的使用顺序。

在一些实施例中，所述技能点确定模块 1820，用于从所述至少两个技能点中，将技能点选择操作所选择的技能点确定为所述第一技能点；其中，所述技能点选择操作是指用户执行的用于选择技能点的操作。

在一些实施例中，所述技能点选择操作和针对所述第一技能控件的操作为一步滑动操作；其中，所述一步滑动操作的起始位置所对应的技能控件为所述第一技能控件，所述一步滑动操作的终止位置所对应的技能点为所述第一技能点；或者，所述一步滑动操作的起始位置所对应的技能点为所述第一技能点，所述一步滑动操作的终止位置所对应的技能控件为所述第一技能控件。

在一些实施例中，所述技能点确定模块 1820，用于根据实时对局数据，从所述至少两个技能点中确定将使用的所述第一技能点；其中，所述实时对局数据包括以下至少之一：所述第一虚拟对象最近 N 次释放的技能、所述第一虚拟对象的属性信息、所述第一虚拟对象正在对局的第二虚拟对象的属性信息、所述第一虚拟对象正在辅助的第三虚拟对象的属性信息，N 为正整数。

在一些实施例中，在使用所述技能点释放所述技能的情况下，所述技能的冷却时长小于所述技能的额定冷却时长。

在一些实施例中，如图 20 所示，所述装置还包括技能点增加模块 1840。

所述技能点增加模块 1840，用于在所述技能点集合中未使用的技能点的数量小于额定最大数量的情况下，若满足技能点恢复条件，则增加所述技能点集合中所述未使用的技能点的数量；其中，所述技能点恢复条件包括以下至少之一：距离上一次使用技能点达到第一时长、距离上一次恢复技能点达到第二时长、所述第一虚拟对象对敌方虚拟对象产生满足第一条件

的伤害、所述第一虚拟对象所属阵营对敌方虚拟对象产生满足第二条件的伤害、所述第一虚拟对象获得第一增益效果、所述第一虚拟对象所属阵营获得第二增益效果。

本申请实施例提供的技术方案，通过支持至少两个技能控件共用同一技能点集合，且当用户执行针对至少两个技能控件中的第一个技能控件的操作时，从技能点集合中确定出将使用的第一技能点，同时控制第一虚拟对象释放第一技能控件对应的技能组中，与第一技能点相对应的技能，实现了通过技能点来确定当前所触发的技能控件对应的当前技能。因此，在有至少两个技能点、至少两个技能控件的情况下，根据不同的技能点选择和不同技能控件的选择，技能的释放有多种可能的组合方式，从而实现技能释放的组合多样化，进而提升技能释放的组合灵活性和多样性。

另外，通过采用本申请实施例提供的技术方案，仅需对技能组中的技能数量或技能点集合中的技能点数量进行调整，即可在不增加多余实体（如技能控件）的情况下，提升技能释放的组合多样性，从而提高了本申请实施例提供的技术方案的可扩展性。另外，通过采用本申请实施例提供的技术方案，也可对用户界面进行优化，如利用技能点减少技能控件数量，从而有利于提高用户界面的利用率。

需要说明的是，上述实施例提供的装置，在实现其功能时，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，上述实施例提供的装置与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。

请参考图 21，其示出了本申请一个实施例提供的终端设备 2100 的结构框图。该终端设备 2100 可以是图 1 所示实施环境中的终端设备 10，用于实施上述实施例中提供的虚拟对象的控制方法。具体来讲：

通常，终端设备 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计算机程序，所述计算机程序经配置以由一个或者一个以上处理器执行，以实现上述虚拟对象的控制方法。

在一些实施例中，终端设备 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。具体地，外围设备包括：射频电路 2104、显示屏 2105、音频电路 2107 和电源 2108 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设备 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。

在示例性实施例中，还提供了一种计算机可读存储介质，所述存储介质中存储有计算机

程序，所述计算机程序在被处理器执行时以实现上虚拟对象的控制方法。

可选地，该计算机可读存储介质可以包括：ROM（Read-Only Memory，只读存储器）、RAM（Random Access Memory，随机存取存储器）、SSD（Solid State Drives，固态硬盘）或光盘等。其中，随机存取存储器可以包括 ReRAM（Resistance Random Access Memory，电阻式随机存取存储器）和 DRAM（Dynamic Random Access Memory，动态随机存取存储器）。

在示例性实施例中，还提供了一种计算机程序产品，所述计算机程序产品包括计算机程序，所述计算机程序存储在计算机可读存储介质中。终端设备的处理器从所述计算机可读存储介质中读取所述计算机程序，所述处理器执行所述计算机程序，使得所述终端设备执行上述虚拟对象的控制方法。

需要说明的是，本申请实施例在收集用户的相关数据之前以及在收集用户的相关数据的过程中，都可以显示提示界面、弹窗或输出语音提示信息，该提示界面、弹窗或语音提示信息用于提示用户当前正在搜集其相关数据，使得本申请仅仅在获取到用户对该提示界面或者弹窗发出的确认操作后，才开始执行获取用户相关数据的相关步骤，否则（即未获取到用户对该提示界面或者弹窗发出的确认操作时），结束获取用户相关数据的相关步骤，即不获取用户的相关数据。换句话说，本申请所采集的所有用户数据，处理严格根据相关国家法律法规的要求，获取个人信息主体的知情同意或单独同意都是在用户同意并授权的情况下进行采集的，并在法律法规及个人信息主体的授权范围内，开展后续数据使用及处理行为且相关用户数据的收集、使用和处理需要遵守相关国家和地区的相关法律法规和标准。例如，本申请中涉及到的操作、用户界面、虚拟环境、虚拟对象等都是在充分授权的情况下获取的。

应当理解的是，在本文中提及的“多个”是指两个或两个以上。“和/或”，描述关联对象的关联关系，表示可以存在三种关系，例如，A 和/或 B，可以表示：单独存在 A，同时存在 A 和 B，单独存在 B 这三种情况。字符“/”一般表示前后关联对象是一种“或”的关系。另外，本文中描述的步骤编号，仅示例性示出了步骤间的一种可能的执行先后顺序，在一些其它实施例中，上述步骤也可以不按照编号顺序来执行，如两个不同编号的步骤同时执行，或者两个不同编号的步骤按照与图示相反的顺序执行，本申请实施例对此不作限定。

以上所述仅为本申请的示例性实施例，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，所述方法由终端设备执行，所述方法包括：

在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，所述技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能；

响应于针对所述至少两个技能控件中的第一技能控件的操作，从所述至少两个技能点中确定将使用的第一技能点；

控制第一虚拟对象释放所述第一技能控件对应的技能组中，与所述第一技能点相对应的技能。

2.根据权利要求1所述的方法，所述至少两个技能控件对应的技能组中的第一技能组包括至少两个技能，所述第一技能组中包括的不同技能对应于不同的技能点。

3.根据权利要求1所述的方法，所述方法还包括：

显示所述至少两个技能点中，未使用的技能点。

4.根据权利要求1所述的方法，所述方法还包括：

在所述技能控件的显示区域，显示与下一个使用的技能点相对应的技能的图标。

5.根据权利要求1所述的方法，所述从所述至少两个技能点中确定将使用的第一技能点，包括：

根据技能点使用顺序，从所述至少两个技能点中确定将使用的所述第一技能点；其中，所述技能点使用顺序用于定义所述至少两个技能点的使用顺序。

6.根据权利要求1所述的方法，所述从所述至少两个技能点中确定将使用的第一技能点，包括：

从所述至少两个技能点中，将技能点选择操作所选择的技能点确定为所述第一技能点；其中，所述技能点选择操作是指用户执行的用于选择技能点的操作。

7.根据权利要求6所述的方法，所述技能点选择操作和针对所述第一技能控件的操作为一步滑动操作；其中，

所述一步滑动操作的起始位置所对应的技能控件为所述第一技能控件，所述一步滑动操作的终止位置所对应的技能点为所述第一技能点；

或者，

所述一步滑动操作的起始位置所对应的技能点为所述第一技能点，所述一步滑动操作的终止位置所对应的技能控件为所述第一技能控件。

8.根据权利要求1所述的方法，所述从所述至少两个技能点中确定将使用的第一技能点，包括：

根据实时对局数据，从所述至少两个技能点中确定将使用的所述第一技能点；

其中，所述实时对局数据包括以下至少之一：所述第一虚拟对象最近N次释放的技能、所述第一虚拟对象的属性信息、所述第一虚拟对象正在对战的第二虚拟对象的属性信息、所述第一虚拟对象正在辅助的第三虚拟对象的属性信息，N为正整数。

9.根据权利要求1所述的方法，在使用所述技能点释放所述技能的情况下，所述技能的冷却时长小于所述技能的额定冷却时长。

10.根据权利要求1所述的方法，所述方法还包括：

在所述技能点集合中未使用的技能点的数量小于额定最大数量的情况下，若满足技能点恢复条件，则增加所述技能点集合中所述未使用的技能点的数量；其中，所述技能点恢复条件包括以下至少之一：距离上一次使用技能点达到第一时长、距离上一次恢复技能点达到第二时长、所述第一虚拟对象对敌方虚拟对象产生满足第一条件的伤害、所述第一虚拟对象所属阵营对敌方虚拟对象产生满足第二条件的伤害、所述第一虚拟对象获得第一增益效果、所述第一虚拟对象所属阵营获得第二增益效果。

11.一种虚拟对象的控制装置，所述装置包括：

控件显示模块，用于在用户界面中显示共用同一技能点集合的至少两个技能控件；其中，所述技能点集合中包括至少两个技能点，每个技能控件对应于一个技能组，每个技能组中包括至少一个技能；

技能点确定模块，用于响应于针对所述至少两个技能控件中的第一技能控件的操作，从所述至少两个技能点中确定将使用的第一技能点；

控制模块，用于控制第一虚拟对象释放所述第一技能控件对应的技能组中，与所述第一技能点相对应的技能。

12.一种终端设备，所述终端设备包括处理器和存储器，所述存储器中存储有计算机程序，所述计算机程序由所述处理器加载并执行以实现如权利要求1至10任一项所述的方法。

13.一种计算机可读存储介质，所述计算机可读存储介质中存储有计算机程序，所述计算机程序由处理器加载并执行以实现如上述权利要求1至10任一项所述的方法。

14.一种计算机程序产品，所述计算机程序产品包括计算机指令，所述计算机指令存储在计算机可读存储介质中，处理器从所述计算机可读存储介质读取并执行所述计算机指令，以实现如权利要求1至10任一项所述的方法。