

VIRTUAL OBJECT CONTROL METHOD AND APPARATUS, TERMINAL, AND STORAGE MEDIUM

RELATED APPLICATION

[0001] This application claims priority to Chinese Patent Application No. 202010507547.6, entitled "METHOD AND APPARATUS FOR CONTROLLING VIRTUAL OBJECT TO CAST SKILL, TERMINAL, AND STORAGE MEDIUM" and filed with the China National Intellectual Property Administration on June 5, 2020.

FIELD OF THE TECHNOLOGY

[0002] This application relates to the field of computer technologies, and in particular, to a virtual object control method and apparatus, a terminal, and a storage medium.

BACKGROUND OF THE DISCLOSURE

[0003] In an application based on a two-dimensional or three-dimensional (3D) virtual environment, for example, a multiplayer online battle arena (MOBA), a user may control a virtual object in a virtual scene to cast skills on other virtual objects, virtual buildings, and the like, and compete against each other using specific victory conditions as goals.

[0004] In the related art, when the user clicks/taps a plurality of skill controls in a short period, skills may be cast on different virtual objects. However, this skill casting effect does not meet the user's expectation (casting skills on the same virtual object in a short period), affecting accuracy of quick skill casting.

SUMMARY

[0005] Embodiments of this application provide a virtual object control method and apparatus, a terminal, and a storage medium, which may improve casting accuracy during skills continuous casting. The technical solutions are as follows.

[0006] According to an aspect, an embodiment of this application provides a virtual object control method, performed by a terminal, the method including:

[0007] displaying a virtual scene interface, the virtual scene interface including a first virtual object, at least two second virtual objects, and at least two skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills;

- [0008]** determining, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and controlling the first virtual object to cast a first skill on the target virtual object;
- [0009]** determining the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition, the second trigger operation being triggered during skill casting of the first skill; and
- [0010]** controlling the first virtual object to cast the second skill on the target virtual object.
- [0011]** According to another aspect, an embodiment of this application provides a virtual object control apparatus, including:
- [0012]** a display module, configured to display a virtual scene interface, the virtual scene interface including a first virtual object, at least two second virtual objects, and at least two skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills;
- [0013]** a first control module, configured to: determine, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and control the first virtual object to cast a first skill on the target virtual object;
- [0014]** a first determining module, configured to determine the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition, the second trigger operation being triggered during skill casting of the first skill; and
- [0015]** a second control module, configured to control the first virtual object to cast the second skill on the target virtual object.
- [0016]** According to another aspect, an embodiment of this application provides a terminal, including a processor and a memory, the memory storing at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction set being loaded and executed by the processor to implement the virtual object control method according to the foregoing aspect.
- [0017]** According to another aspect, an embodiment of this application provides a computer-readable storage medium, storing at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction

set being loaded and executed by a processor to implement the virtual object control method according to the foregoing aspect.

[0018] According to another aspect, an embodiment of this application provides a computer program product, the computer program product, when executed by a processor, being used for implementing the virtual object control method.

BRIEF DESCRIPTION OF THE DRAWINGS

[0019] FIG. 1 is a schematic implementation diagram of an skill continuous casting process in the related art.

[0020] FIG. 2 is a schematic diagram of an implementation environment according to an embodiment of this application.

[0021] FIG. 3 is a flowchart of a virtual object control method according to an exemplary embodiment of this application.

[0022] FIG. 4 is a schematic implementation diagram of an skill continuous casting process according to an exemplary embodiment of this application.

[0023] FIG. 5 is a schematic diagram of a control region division manner in an skill control according to an exemplary embodiment of this application.

[0024] FIG. 6 is a flowchart of a virtual object control method according to another exemplary embodiment of this application.

[0025] FIG. 7 is a diagram of a principle of an skill casting target inheritance mechanism according to an exemplary embodiment of this application.

[0026] FIG. 8 is a schematic diagram of an interface of controlling a virtual object to cast an skill according to an exemplary embodiment of this application.

[0027] FIG. 9 is a flowchart of a virtual object control method according to another exemplary embodiment of this application.

[0028] FIG. 10 is a structural block diagram of a virtual object control apparatus according to an exemplary embodiment of this application.

[0029] FIG. 11 is a structural block diagram of a terminal according to an exemplary embodiment of this application.

DESCRIPTION OF EMBODIMENTS

[0030] To make objectives, technical solutions, and advantages of this application clearer, the following further describes implementations of this application in detail with reference to the accompanying drawings.

[0031] The term "Plurality of" mentioned in the specification means two or more. "And/or" describes an association relationship for describing associated objects and represents that three relationships may exist. For example, A and/or B may represent the following three cases: only A exists, both A and B exist, and only B exists. The character "/" generally indicates an "or" relationship between the associated objects.

[0032] First, terms involved in the embodiments of this application are briefly introduced as follows:

[0033] Virtual scene is a virtual scene displayed (or provided) when an application is running on a terminal. The virtual scene may be a simulated world of the real world, or may be a semi-simulated semi-fictional 3D world, or may be an entirely fictional 3D world. The virtual scene may be any one of a two-dimensional virtual scene, a 2.5-dimensional virtual scene, and a 3D virtual scene.

[0034] Optionally, the virtual scene is further used for a virtual scene battle between at least two virtual objects, and virtual resources available to the at least two virtual objects are provided in the virtual scene. Optionally, the virtual scene includes a lower left corner region and an upper right corner region that are symmetrical. Virtual objects on two opposing camps occupy the regions respectively, and the objective of each side is to destroy a target building/fort/base/crystal deep in the opponent's region to win victory.

[0035] Virtual object: a movable object in a virtual scene. The movable object may be at least one of a virtual human, a virtual animal, and an animated human character. Optionally, when the virtual scene is a 3D virtual scene, the virtual object may be a 3D model. Each virtual object has a shape and a volume in the 3D virtual scene, and occupies some space in the 3D virtual scene.

[0036] Optionally, the virtual object is a 3D character constructed based on 3D human skeleton technology. The virtual object wears different skins to implement different appearances. In some implementations, the virtual object may also be implemented by using a 2.5-dimensional model or a two-dimensional model, which is not limited in the embodiments of this application.

[0037] Skill: an skill in the embodiments of this application is an skill cast by a virtual character and used for modifying an attribute value of a virtual object, another virtual object, or

both a virtual object and another virtual object. A virtual object has at least one skill, and different virtual objects correspond to different skills. An skill of a virtual character may be obtained and upgraded during level upgrading, and a virtual object may obtain an skill of another virtual object.

[0038] Optionally, according to skill effects, skills may be divided into: a damage skill (used to decrease a health point of a virtual object), a shield skill (used to add a shield for a virtual object), an acceleration skill (used to increase a moving speed of a virtual object), a deceleration skill (used to decrease a moving speed of a virtual object), an imprison skill (used to restrict a movement of a virtual object for a certain duration), a forced displacement skill (used to force a virtual object to move), a silence skill (used to restrict a virtual object from casting an skill for a certain duration), a restore skill (used to restore a health point or an energy value of a virtual object), a field of view skill (used to obtain/shield the field of view within a certain range or another virtual characters), a passive skill (skill that can be triggered when performing an normal attack), and the like. This is not limited in this embodiment.

[0039] Optionally, according to an skill casting manner, skills may be divided into a targeted skill and a non-targeted skill. The targeted skill is an skill specified to an skill receiver, that is, after a targeted skill is specified to an skill casting target, the skill casting target is definitely affected by the skill. The non-targeted skill is an skill pointing to a specified direction, range, or region, and a virtual object located in the direction, range, or region is affected by the skill.

[0040] A MOBA game is an arena game in which different virtual teams on at least two opposing camps occupy respective map regions in a virtual scene, and compete against each other using specific victory conditions as goals. The victory condition includes, but is not limited to, at least one of occupying forts or destroy forts of the opposing camps, killing virtual objects in the opposing camps, surviving in a specified scenario and time, seizing a specific resource, and outscoring the opponent within a specified time. The battle arena game may take place in rounds. The same map or different maps may be used in different rounds of the battle arena game. Each virtual team includes one or more virtual objects, for example, 1 virtual object, 2 virtual objects, 3 virtual objects, or 5 virtual objects.

[0041] A MOBA game is a game in which several bases are provided in a virtual scene, and users on different camps control virtual characters to battle in the virtual scene, occupy bases or destroy the base of the opposing camp. For example, in the MOBA game, the users may be divided into two opposing camps. The virtual characters controlled by the users are scattered in the virtual scene to compete against each other, and the victory condition is to destroy or occupy all enemy

bases. The MOBA game takes place in rounds. A duration of a round of the MOBA game is from a time point at which the game starts to a time point at which the victory condition is met.

[0042] In the MOBA game, a user may control a virtual object to move in a virtual scene, and can trigger the virtual object to cast a corresponding skill by using an skill control. When a trigger operation on an skill control is received, the terminal determines, according to an skill casting range corresponding to an skill, an skill casting target from virtual objects located in the skill casting range, so as to control the virtual object to cast an skill on the skill casting target.

[0043] Because different skills correspond to different skill casting ranges, when a virtual object casts different skills at the same position, different skills may correspond to different skill casting targets. For example, as shown in FIG. 1, a second virtual object 102 and a third virtual object 103 are around a first virtual object 101. When a tap operation on an skill control corresponding to an skill A is received, the terminal determines, according to an skill casting range 104 corresponding to the skill A, the second virtual object 102 with the lowest health point within the range as an skill casting target. When a tap operation on an skill control corresponding to an skill B is received, the terminal determines, according to an skill casting range 105 corresponding to the skill B, the third virtual object 103 with the lowest health point within the range as an skill casting target.

[0044] By using the foregoing skill casting target determining methods, each skill casting of the terminal is regarded as an independent procedure. Correspondingly, there is no correlation between skill casting targets of skill casting. However, during actual applications, the user may quickly trigger different skills, to cause damage to the same virtual object in a short time period. If the foregoing method is adopted, when different skills indicate different skill casting targets, different skills may be cast on different objects, which affects accuracy of skill casting.

[0045] In this embodiment of this application, by introducing an skill casting target inheritance mechanism, when casting skills continuously, a current skill can inherit an skill casting target of a previous skill (during casting), and does not search an skill casting target according to a conventional searching rule. Therefore, different skills cast continuously can be cast on the same skill casting target, so as to cause damage to the same virtual object in a short period and to improve accuracy of skill casting.

[0046] FIG. 2 is a schematic diagram of an implementation environment according to an embodiment of this application. The implementation environment may include a first terminal 210, a server 220, and a second terminal 230.

[0047] An application 211 supporting a virtual scene is installed and run on the first terminal 210, and the application 211 may be a multiplayer online battle program. When the first terminal 210 runs the application 211, a user interface (UI) of the application 211 is displayed on a screen of the first terminal 210. The application 211 may be any one of a military simulation program, a MOBA game, a battle royale shooting game, and a simulation game (SLG).

[0048] In this embodiment, an example in which the application 211 is a MOBA game is used for description. The first terminal 210 is a terminal used by a first user 212. The first user 212 uses the first terminal 210 to control a first virtual object located in a virtual scene to perform activities, and the first virtual object may be referred to as a master virtual object of the first user 212. The activities of the first virtual object include, but are not limited to: at least one of body postures adjusting, crawling, walking, running, riding, flying, jumping, driving, picking, shooting, attacking, throwing, and skill casting. For example, the first virtual object is a first virtual character such as a simulated character or a cartoon character.

[0049] An application 231 supporting a virtual scene is installed and run on the second terminal 230, and the application 231 may be a multiplayer online battle program. When the second terminal 230 runs the application 231, a UI of the application 231 is displayed on a screen of the second terminal 230. The client may be any one of a military simulation program, a MOBA game, a battle royale shooting game, and an SLG.

[0050] In this embodiment, an example in which the application 231 is a MOBA game is used for description. The second terminal 230 is a terminal used by a second user 232. The second user 232 uses the second terminal 230 to control a second virtual object located in a virtual scene to perform activities, and the second virtual object may be referred to as a master virtual character of the second user 232. For example, the second virtual object is a second virtual character, such as a simulated character or a cartoon character.

[0051] Optionally, the first virtual object and the second virtual object are located in the same virtual scene. Optionally, the first virtual object and the second virtual object may belong to the same camp, the same team, or the same organization, may be friends of each other, or may have a temporary communication permission. Optionally, the first virtual object and the second virtual object may belong to different camps, different teams, different organizations, or may be enemies of each other.

[0052] Optionally, the applications installed on the first terminal 210 and the second terminal 230 are the same, or the applications installed on the two terminals are the same type of applications on different operating system platforms (Android system or iOS system). The first

terminal 210 may generally refer to one of a plurality of terminals, and the second terminal 230 may generally refer to another one of the plurality of terminals. In this embodiment, the first terminal 210 and the second terminal 230 are merely used as an example for description. The first terminal 210 and the second terminal 230 are of the same or different device types, and the device type includes at least one of a smartphone, a tablet computer, an e-book reader, a digital player, a laptop, and a desktop computer.

[0053] Although FIG. 2 shows only two terminals, a plurality of other terminals may access the server 220 in different embodiments. Optionally, one or more terminals are terminals corresponding to a developer. A developing and editing platform for the application supporting a virtual scene is installed on the terminal. The developer can edit and update the application on the terminal and transmit an updated application installation package to the server 220 through a wired or wireless network. The first terminal 210 and the second terminal 230 can download an application installation package from the server 220 to update the application.

[0054] The first terminal 210, the second terminal 230, and the another terminal are connected to the server 220 through a wireless network or a wired network.

[0055] The server 220 includes at least one of one server, a server cluster including a plurality of servers, a cloud computing platform, and a virtualization center. The server 220 is configured to provide a backend service for an application supporting a 3D virtual scene. Optionally, the server 220 is responsible for primary computing work, and the terminal is responsible for secondary computing work; or the server 220 is responsible for secondary computing work, and the terminal is responsible for primary computing work; or the server 220 and the terminals perform collaborative computing by using a distributed computing architecture among each other.

[0056] In a schematic example, the server 220 includes a memory 221, a processor 222, a user account database 223, a battle service module 224, and a user-oriented input/output (I/O) interface 225. The processor 222 is configured to load instructions stored in the server 220, and process data in the user account database 223 and the battle service module 224. The user account database 223 is configured to store data of user accounts used by the first terminal 210, the second terminal 230, and the other terminals, for example, avatars of the user accounts, nicknames of the user accounts, battle effectiveness indexes of the user accounts, and service zones of the user accounts. The battle service module 224 is configured to provide a plurality of battle rooms for the users to battle, for example, a 1V1 battle room, a 3V3 battle room, a 5V5 battle room, and the like. The user-oriented I/O interface 225 is configured to establish communication between the first

terminal 210 and/or the second terminal 230 via a wireless network or a wired network for data exchange.

[0057] FIG. 3 is a flowchart of a virtual object control method according to an exemplary embodiment of this application. An example in which the method is used in the first terminal 210 or the second terminal 230 in the implementation environment shown in FIG. 2 or other terminals in the implementation environment is used to describe this embodiment. The method includes the following steps:

[0058] Step 301: Display a virtual scene interface, the virtual scene interface including a first virtual object, at least two second virtual objects, and at least two skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills.

[0059] The second virtual object may include a virtual object belonging to the same camp as the first virtual object, or may include a virtual object belonging to different camps from the first virtual object.

[0060] In some embodiments, both the first virtual object and the second virtual object are virtual objects controlled by users in a battle. Alternatively, the first virtual object is a virtual object controlled by a user in a battle, but the second virtual object is a virtual object controlled by a server, that is, an artificial intelligence (AI) virtual object.

[0061] In a possible implementation, the virtual scene interface includes a virtual scene picture and a control layer located on an upper layer of the virtual scene picture, and the skill control is located on the control layer. In addition to the skill control, a data bar, a movement control, a map control, a normal attack control, a virtual prop control, a message control, a record control, and other UI controls may alternatively be set on the control layer. This is not limited in this embodiment.

[0062] The virtual scene is a virtual scene with arbitrary boundary shapes, and the first virtual object is located in a visible range of the virtual scene interface. Optionally, the first virtual object is located at a visual center of the virtual scene picture, that is, located at a center of a virtual scene picture obtained by observing the virtual scene by using a third-person perspective.

[0063] Perspective refers to an observation angle at which observation is performed in a virtual scene from a first-person perspective or a third-person perspective of a virtual character. Optionally, in this embodiment of this application, the perspective is an angle for observing the virtual character through a camera model in the virtual scene.

[0064] Optionally, the camera model automatically follows the virtual object in the virtual scene. That is, when a position of the virtual object in the virtual scene changes, a position of the camera model following the virtual object in the virtual scene changes simultaneously, and the camera model is always within a preset distance range of the virtual object in the virtual scene.

[0065] Optionally, in the automatic following process, relative positions of the camera model and the virtual object remain unchanged. The third-person perspective is used as an example for description in this embodiment of this application. Optionally, the camera model is located behind the virtual object (for example, the head and the shoulders of the virtual character).

[0066] In some embodiments, the skill control adopts a button, and the first virtual object may be controlled to cast corresponding skills on other virtual objects by triggering the button.

[0067] Step 302: Determine, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and control the first virtual object to cast a first skill on the target virtual object.

[0068] Optionally, the first skill corresponding to the first skill control is an skill having an active casting function, and an skill casting target (that is, a virtual object affected by the skill) of the first skill is a virtual object belonging to the same camp as the first virtual object, or is a virtual object belonging to different camps from the first virtual object. In this embodiment of this application, the skill casting target of the first skill is the target virtual object in the second virtual objects. The target virtual object is at least one virtual object.

[0069] Optionally, the first trigger operation is at least one of a tap operation, a double-tap operation, a press operation, a drag operation, and a long press operation. An operation type of the first trigger operation is related to a casting manner of the first skill. For example, when the first skill is a targeted skill, the first trigger operation is a tap operation, and when the first skill is a non-targeted skill, the first trigger operation is a drag operation.

[0070] In some embodiments, the same skill control corresponds to at least two trigger operations. For example, an skill control corresponds to two trigger operations, which are respectively used for actively selecting an skill casting target (direction or range) or automatically selecting an skill casting target (direction or range).

[0071] Optionally, the first trigger operation is used for triggering the terminal to determine a target virtual object (automatic searching) from at least one second virtual object according to a target automatic searching rule. Alternatively, the first trigger operation is used for triggering the terminal to determine a specified virtual object in at least one second virtual object as a target

virtual object (actively aiming). The specified virtual object is indicated by the first trigger operation.

[0072] Optionally, when the first virtual object casts the first skill, the first virtual object presents a corresponding skill casting animation. Alternatively, the terminal controls the first virtual object to perform a corresponding skill casting action.

[0073] For example, as shown in FIG. 4, when a tap operation corresponding to an skill control 41 corresponding an skill A is received, the terminal controls the first virtual object 42 to cast the skill A on a mage 44 (that is, the target virtual object) in a first skill casting range 43.

[0074] Step 303: Determine the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition, the second trigger operation being triggered during skill casting of the first skill.

[0075] In a casting process of the first skill, if a second trigger operation on the second skill control is received, different from that in the related art, the terminal determines an skill casting target (direction or range) of the second skill according to the second trigger operation, an skill casting rule and an skill casting range corresponding to the second skill, and a relative orientation of each second virtual object and the first virtual object.

[0076] In this embodiment of this application, the terminal detects whether the second trigger operation meets the skill continuous casting condition, if the skill continuous casting condition is met, the terminal determines the skill casting target corresponding to the second skill as the target virtual object. If the skill continuous casting condition is not met, the terminal determines the skill casting target of the second skill according to the method provided in the related art.

[0077] In some embodiments, the skill continuous casting condition includes an skill trigger operation condition and an skill casting time condition. Correspondingly, when the second trigger operation meets the skill trigger operation condition and a trigger time corresponding to the second trigger operation meets the skill casting time condition, the second trigger operation is determined to meet the skill continuous casting condition.

[0078] Similar to the first skill, the second skill is also an skill having an active casting function.

[0079] For example, as shown in FIG. 4, when a tap operation corresponding to an skill control 45 corresponding to an skill B is received in a casting process of the skill A, because the

skill continuous casting condition is met, the terminal determines the mage 44 as an skill casting target of the skill B.

[0080] Step 304: Control the first virtual object to cast the second skill on the target virtual object.

[0081] Further, the terminal controls the first virtual object to cast the second skill on the target virtual object, so that under an skill continuous casting, and even if the continuously cast skills correspond to different skill casting ranges, the skills can be cast on the same virtual object.

[0082] For example, as shown in FIG. 4, the terminal controls the first virtual object 42 to cast the skill A and the skill B (the skill A is cast before the skill B) on the mage 44. If the method in the related art is adopted, because a second skill casting range 46 of the skill B is larger than the first skill casting range 43, and a health point of a warrior 47 in the second skill casting range 46 is lower than that of the mage 44, it is determined that the skill casting target of the skill B is the warrior 47, not the mage 44.

[0083] In some embodiments, if a third trigger operation on a third skill control (may be different from the first skill control and the second skill control, or may be the first skill control (the first skill has been cast and is in a status that can be cast)) is received in a casting process of the second skill, and the third trigger operation meets the skill continuous casting condition, the terminal further determines the target virtual object as an skill casting target corresponding to a third skill, and controls the first virtual object to cast the third skill. This is not repeated in this embodiment herein.

[0084] Based on the above, in this embodiment of this application, when a trigger operation on an skill control is received, if the trigger operation is triggered in an skill casting process of a previous skill and the trigger operation meets an skill continuous casting condition, an skill casting target indicated by the previous skill is determined as an skill casting target indicated by the current skill, so as to control a virtual object to cast an skill on the skill casting target. Through the skill casting target inheritance mechanism, a problem that continuously cast skills are cast on different objects because different skills correspond to different skill range can be avoided, and skills may be cast on the same object continuously in an skill continuous casting scenario, thereby improving accuracy of skill casting.

[0085] In a possible implementation, to enable the skill casting to meet quick casting requirements of the user and the active aiming requirements, a control region of an skill control is divided into two parts, which are respectively an automatic control region and an active control

region. The automatic control region is used for triggering a terminal to automatically determine an skill casting target, direction, or range, and the active control region is used for triggering a user to actively select an skill casting target, direction, or range. Correspondingly, the user may trigger to cast an skill quickly by performing a touch operation on the automatic control region of the skill control, and may actively select an skill casting target by performing a touch operation on the active control region of the skill control, so as to aim actively.

[0086] For example, as shown in FIG. 5, an skill control 50 includes an automatic control region 51 and an active control region 52. The automatic control region 51 is a circular region centered on a center of the skill control 50, and the active control region 52 is a ring region outside the automatic control region 51 and in the skill control 50. When a tap operation (an operation of quickly lifting after touching, as shown on the left side in FIG. 5) on the automatic control region 51 is received, the terminal automatically determines an skill casting target according to an skill casting range and a relative position between another virtual object and a current controlled virtual object, and cast an skill on the skill casting target. When a drag operation (touch and drag, as shown in the right side in FIG. 5) on the active control region 52 is received, the terminal determines an skill casting target (a target actively selected by the user may be marked in a special manner such as highlight) according to a mapping relationship between a touch position in the active control region 52 and a virtual object in a virtual scene, and controls the virtual object to cast an skill on the skill casting target after the drag operation.

[0087] The foregoing embodiment is merely used to describe a division manner of an automatic control region and an active control region in an skill control schematically, and the skill control may further adopt other possible control region division manners. This is not limited in this embodiment.

[0088] If the second skill control adopts the foregoing control region division manner, to avoid from wrongly replacing an skill casting object actively selected by the user by using the second skill control with the skill casting object of the first skill, the terminal needs to determine whether to inherit the skill casting target based on a control region in which the touch operation is located. A schematic embodiment is provided below for description.

[0089] FIG. 6 is a flowchart of a virtual object control method according to another exemplary embodiment of this application. An example in which the method is used in the first terminal 210 or the second terminal 230 in the implementation environment shown in FIG. 2 or other terminals in the implementation environment is used to describe this embodiment. The method includes the following steps:

[0090] Step 601: Display a virtual scene interface, the virtual scene interface including a first virtual object, at least two second virtual objects, and at least two skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills.

[0091] For the implementation of this step, reference may be made to step 301, and details are not described again in this embodiment.

[0092] Step 602: Determine, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and control the first virtual object to cast a first skill on the target virtual object.

[0093] To inherit the skill casting target subsequently, when controlling the first virtual object to cast the first skill, a specific target virtual object (that is, a specific skill casting target) is required.

[0094] In a possible implementation, the first skill control also includes an automatic control region and an active control region. The target virtual object may be specified by a first trigger operation located in the active control region, or may be determined according to a target automatic searching rule of the first skill after the terminal receives a first trigger operation located in the automatic control region.

[0095] Optionally, the terminal determines the target virtual object according to a target automatic searching rule corresponding to the first skill and in response to the first skill being a targeted skill or a non-targeted skill and a touch end position of the first trigger operation being located in the automatic control region, and casts the first skill, and

[0096] determines, in response to first skill being a targeted skill and a touch end position of the first trigger operation being located in the active control region, a second virtual object corresponding to the touch end position as the target virtual object, and casts the first skill.

[0097] In some embodiments, the target automatic searching rule includes at least one of the following: determining an enemy virtual object with a lowest health point (an absolute health point or a percentage of health point) in an skill casting range as an skill casting target (when the skill is cast on an enemy virtual object); determining a friend virtual object with a lowest health point in an skill casting range as an skill casting target (when the skill is cast on a friend virtual object); and determining an enemy virtual object closest to a current virtual object in an skill casting range as an skill casting object.

[0098] Optionally, when the first skill is a targeted skill and the touch end position of the first trigger operation is located in the automatic control region, the terminal controls the first virtual object to cast the first skill on the target virtual object directionally, which is not shielded or affected by another object between the target virtual object and the first virtual object.

[0099] When the first skill is a non-targeted skill and the touch end position of the first trigger operation is located in the automatic control region, the terminal controls, according to a relative direction between the target virtual object and the first virtual object, the first virtual object to cast the first skill according to the relative direction. For example, the first skill (a range-type skill) is cast in the skill casting range of the first skill by using the relative direction as an skill casting center, or the first skill (a ballistic trajectory type skill) is cast according to a relative orientation. When the target virtual object moves during skill casting, or there is another object between the target virtual object and the first virtual object, casting effect of the first skill is affected, and even the target virtual object cannot be hit.

[0100] Optionally, when the first skill is a targeted skill and the touch end position of the first trigger operation is located in the active control region, it represents that the user needs to aim actively. Therefore, the terminal determines the second virtual object corresponding to the touch end position as the target virtual object, and cast the first skill on the target virtual object directionally.

[0101] In other possible implementations, if the first skill is a non-targeted skill and the touch end position of the first trigger operation is located in the active control region, because there may be a plurality of virtual objects in an skill casting direction indicated by the touch end position, the terminal cannot specify the skill casting target, that is, the terminal cannot determine the target virtual object corresponding to the first skill. Correspondingly, even if a second trigger operation meeting the skill continuous casting condition is received subsequently, the skill casting target inheritance mechanism cannot be enabled.

[0102] Step 603: Determine a target inheritance period corresponding to the target virtual object according to an skill flow corresponding to the first skill and in response to the first trigger operation, the skill flow being used for controlling a cast procedure of the first skill, and the target inheritance period being a valid period for the second skill to inherit the target virtual object.

[0103] Skill casting is not instantaneous, and requires a series of skill procedures, that is, it takes a certain amount of time from the beginning of skill casting to the end of skill casting. The skill flow, that is, a logic layer, is configured to control a control logic of an skill cast procedure. Different skills correspond to different skill flows.

[0104] In a possible implementation, the inheritance for the target virtual object has a valid period, that is, not all second trigger operations on the second skill control received at any time in a casting process of the first skill meet the skill continuous casting condition. Only second trigger operations (that is, there is a requirement on a touch time of the second trigger operation) received in a valid inheritance period in a casting process of the first skill can meet the skill continuous casting condition.

[0105] Optionally, because the valid period is related to the cast procedure of the first skill, the terminal obtains an skill flow corresponding to the first skill from the logic layer after receiving the first trigger operation on the first skill control, further to determine a target inheritance period corresponding to the target virtual object according to a first skill cast procedure indicated by the skill flow. The target inheritance period is a middle and early period of the first skill cast procedure.

[0106] In some embodiments, the cast procedure controlled by the skill flow includes an skill casting preparation stage, an skill casting stage, and an skill casting ending stage. The skill casting preparation stage is also referred to as an skill anticipation stage, and in this stage, the virtual object presents an skill casting preparation action. The skill casting ending stage is also referred to as an skill post-shake stage, and in this stage, the virtual object presents an skill casting ending action.

[0107] In a possible implementation, if a trigger operation on a current skill is received in an skill casting preparation stage or skill casting stage of a previous skill, it represents that there is an skill continuous casting requirement, so as to determine that an skill continuous casting condition is met. If a trigger operation on a current skill is received in an skill casting ending stage of a previous skill, because the previous skill has been cast (additional effects may not be cast to the skill casting target in the skill casting ending stage), there is no skill continuous casting requirement, so as to determine that the skill continuous casting condition is not met. Optionally, the process of determining a target inheritance period corresponding to the target virtual object according to the skill flow may include the following steps:

[0108] 1. Determine a duration of the target inheritance period according to a first duration of the skill casting preparation stage and a second duration of the skill casting stage.

[0109] In a possible implementation, the terminal determines a trigger operation whose triggered time is in an skill casting preparation stage or skill casting stage corresponding to a previous skill as a trigger operation meeting the skill continuous casting condition. Correspondingly, after obtaining an skill flow of the first skill, the terminal obtains a first duration of the skill casting preparation stage and a second duration of an skill casting stage, further to

determine a sum of the first duration and the second duration as the duration of the target inheritance period.

[0110] In an exemplary example, in the skill flow corresponding to the first skill, if the first duration of the skill casting preparation stage is 100 ms, and the second duration of the skill casting stage is 125 ms, then the terminal determines the duration of a target inheritance period as 225 ms.

[0111] In other possible implementations, the terminal may alternatively determine a duration shorter than the sum of the first duration and the second duration as the duration of the target inheritance period. This is not limited in this embodiment.

[0112] Because different skills correspond to different durations of an skill casting preparation stage and skill casting stage, different skills correspond to different durations of a target inheritance period.

[0113] 2. Determine the target inheritance period according to a start execution time and the duration of the skill flow.

[0114] Further, the terminal determines a start time of the target inheritance period and an end time according to a start execution time of the skill flow and a duration of the target inheritance period. The start time is the start execution time of the skill flow, and the end time is determined according to the duration and the start time.

[0115] In a possible implementation, as shown in FIG. 7, when a trigger operation on the first skill control is received, the terminal performs an skill flow corresponding to the first skill, and sets a target inheritance node between the beginning and the end of the skill flow, and a time between the beginning of the skill flow and the target inheritance node may be used as the target inheritance period.

[0116] There is no strict sequence between step 602 and step 603, that is, the terminal may perform step 602 and step 603 synchronously after the first trigger operation is received. This is not limited in this embodiment.

[0117] Step 604: Receive a trigger operation on the second skill control.

[0118] In a casting process of the first skill, the terminal receives a trigger operation on the second skill control.

[0119] Step 605: Determine, in response to a touch end position corresponding to the trigger operation on the second skill control being located in the automatic control region, that the trigger operation on the second skill control belongs to the second trigger operation.

[0120] In order to avoid the failure of active aiming caused by enabling the skill casting target inheritance mechanism, the user may select an skill casting target by using an active aiming function provided by the second skill control. In this embodiment, the terminal obtains a touch end position corresponding to a trigger operation, and determines that the trigger operation belongs to the second trigger operation when the touch end position is located in the automatic control region (it represents that the user does not aim actively, that is, the user does not specify an skill casting target), that is, the trigger operation meets a trigger operation condition in the skill continuous casting condition.

[0121] For example, as shown in FIG. 5, when the touch end position is located in the automatic control region 51, the terminal determines that a trigger operation on the skill control 50 meets the trigger operation condition in the skill continuous casting condition.

[0122] Step 606: Determine, in response to the trigger operation on the second skill control belonging to the second trigger operation, and a triggered time of the second trigger operation is in the target inheritance period, that the second trigger operation meets the skill continuous casting condition, and obtain an skill casting range and an skill casting rule of the second skill.

[0123] Further, when the trigger operation condition is met, the terminal further detects whether the triggered time of the second trigger operation is in the target inheritance period; determines that the second trigger operation meets a triggered time condition if the triggered time of the second trigger operation is in the target inheritance period, and further determines that the second trigger operation meets the skill continuous casting condition.

[0124] For example, as shown in FIG. 8, the user first taps a first skill control 41, and controls the first virtual object 42 to cast an skill A to the mage 44. When the skill A is in a cooling state, the user quickly taps an automatic control region 451 of a second skill control 45. Because a time for tapping the second skill control 45 belongs to the target inheritance period, the terminal determines the mage 44 as the skill casting target of an skill B.

[0125] Optionally, the terminal determines an skill casting target (may be the same as or different from the target virtual object) of a second skill from at least two second virtual objects according to the second trigger operation and an skill casting rule and an skill casting range of the second skill, in response to the trigger operation on the second skill control not belonging to the second trigger operation, or the triggered time of the second trigger operation being outside the target inheritance period.

[0126] Because different skills have different skill casting ranges and skill casting rules, if the inherited target virtual object does not meet the skill casting range or the skill casting rule of the second skill, invalid skill casting may occur. Therefore, in some embodiments, when the second trigger operation meets the skill continuous casting condition, the terminal further obtains the skill casting range and the skill casting rule of the second skill, and detects whether the target virtual object meets the skill casting range and the skill casting rule.

[0127] Optionally, the skill casting rule includes a casting target rule. The casting target rule indicates that: only can be cast on a friend virtual object, only can be cast on an enemy virtual object, only can be cast on an enemy hero, or only can be cast on an enemy virtual object in a predetermined duration.

[0128] In a possible implementation, the terminal obtains position information of the target virtual object in the virtual scene and attribute information of the target virtual object, so as to detect whether the target virtual object is in the skill casting range of the second skill according to the position information and detect whether the target virtual object meets the skill casting rule according to the attribute information.

[0129] The attribute information may include at least one of the following: a camp to which the virtual object belongs, a health point, an energy value, and an skill cast in a predetermined duration.

[0130] Step 607: Determine the target virtual object as the skill casting target corresponding to the second skill in response to the target virtual object being located in the skill casting range and the target virtual object meeting the skill casting rule.

[0131] When the target virtual object is located in the skill casting range and the target virtual object meets the skill casting rule, the terminal determines that the second skill can be cast on the target virtual object, so as to determine an skill casting target corresponding to the second skill is the target virtual object.

[0132] For example, as shown in FIG. 7, when the trigger operation on the second skill control is received in the target inheritance period, the second skill inherits the skill casting target of the first skill. If the trigger operation on the second skill control is received outside the target inheritance period, the terminal dynamically determines the skill casting target of the second skill according to the skill casting range and the skill casting rule of the second skill.

[0133] In other possible implementations, if the target virtual object is located outside the skill casting range, and/or, the target virtual object does not meet the skill casting rule, the terminal

determines that the second virtual object belongs to the skill casting target of the second skill determined according to the skill casting range and the skill casting rule of the second skill.

[0134] In some embodiments, since continuously cast skills are usually a combined skill, to further improve the accuracy of skill casting, when an skill casting combination formed by the first skill and the second skill belongs to an skill combination list of the first virtual object, the terminal determines the target virtual object as the skill casting target corresponding to the second skill. The skill combination list includes at least one skill casting combination corresponding to the first virtual object, and the skill casting combination includes at least two skills and a casting sequence between the skills.

[0135] In a possible implementation, the terminal stores skill combination lists corresponding to different virtual objects, and each skill combination in the skill combination list is formed by at least two skills, and there is a specified skill casting sequence. For example, an skill combination list may be shown as Table 1:

Table 1

Skill combination	First skill	Second skill
1	Skill A	Skill B
2	Skill B	Skill C

[0136] The skill combination 1 includes an skill A and an skill B, where the skill A is cast before the skill B. The skill combination 2 includes an skill B and an skill C, and the skill B is cast before the skill C.

[0137] When the first skill and the second skill belong to an skill combination in the skill combination list, and an skill sequence meets an skill casting sequence in the skill combination, it is determined that an skill casting combination formed by the first skill and the second skill belongs to the skill combination list.

[0138] For example, when trigger operations on the skill A and the skill B are received consecutively, because the skill A and the skill B belong to the skill combination 1, and a sequence for receiving the trigger operations meets the skill casting sequence of the skill combination, the terminal determines an skill casting target of the skill A as an skill casting target of the skill B. If trigger operations on the skill A and the skill C are received consecutively, because an skill casting

combination formed by the skill A and the skill C does not belong to the skill combination list, the skill C cannot inherit the skill casting target of the skill A directly.

[0139] Step 608: Control the first virtual object to cast the second skill on the target virtual object in response to the second skill being a targeted skill.

[0140] For different types of second skills, manners for controlling the first virtual object to cast the second skills after the skill casting target is inherited may also be different.

[0141] Optionally, when the second skill is a targeted skill (the targeted skill is an skill with a specified skill casting target), the terminal controls the first virtual object to cast the second skill on the target virtual object directly. No matter whether the target virtual object moves or whether there is another object between the target virtual object and the first virtual object, the second skill can affect the target virtual object directly. For example, the second skill is a targeted damage skill, and the second skill is to change a health point of the target virtual object targeted.

[0142] Step 609: Determine a relative direction of the target virtual object relative to the first virtual object in response to the second skill being a non-targeted skill; and control the first virtual object to cast the second skill according to the relative direction.

[0143] Optionally, when the second skill is a non-targeted skill, the terminal controls, according to a relative direction of a current position of the target virtual object and the first virtual object, the first virtual object to cast the second skill according to the relative direction.

[0144] When the target virtual object is displaced in a casting process of the second skill, or there is another virtual object between the target virtual object and the first virtual object, the second skill may not hit the target virtual object, or the skill effect of the second skill may be affected.

[0145] In an exemplary example, the virtual object control method shown in FIG. 9 is applied in the first terminal 210 or the second terminal 230 in the implementation environment or another terminal in the implementation environment shown in FIG. 2. When a trigger operation on an skill control is received, an skill casting target is determined by using the following steps:

[0146] Step 901: Receive an skill casting operation.

[0147] Step 902: Detect whether there is an skill being cast. Step 903 is performed if there is an skill being cast, and step 907 is performed if there is not an skill being cast.

[0148] Step 903: Detect whether the skill being cast has an skill casting target. Step 904 is performed if the skill being cast has an skill casting target, and step 907 is performed if the skill being cast does not have an skill casting target.

[0149] Step 904: Detect whether a second trigger operation meets an skill continuous casting condition. Step 905 is performed if the second trigger operation meets the skill continuous casting condition, and step 907 is performed if the second trigger operation does not meet an skill continuous casting condition.

[0150] Step 905: Detect whether an skill casting target of the skill being cast meets an skill casting condition of a current skill. Step 906 is performed if the skill casting target of the skill being cast meets the skill casting condition of the current skill, and step 907 is performed if the skill casting target of the skill being cast does not meet the skill casting condition of the current skill.

[0151] Step 906: Inherit the skill casting target of the skill being cast.

[0152] Step 907: Perform a default procedure for determining the skill casting target.

[0153] In conclusion, in this embodiment, for a second skill control having an automatic control region and an active control region, only when a trigger operation on the second skill control is located in the automatic control region, the terminal enables an skill casting object inheritance mechanism, thereby avoiding affecting an active aiming operation of the user, and further improving accuracy of continuous skill casting.

[0154] In addition, in this embodiment, the terminal determines a valid inheritance period of an skill casting target according to durations of an skill casting preparation stage and an skill casting stage in an skill flow, and enables an skill casting object inheritance mechanism when a triggered time of the trigger operation on the second skill control is within an skill valid period, to avoid a problem that an skill casting target of a current skill is still an skill casting target of a previous cast skill, thereby improving accuracy of continuous skill casting.

[0155] In addition, in this embodiment, the terminal determines whether to inherit an skill casting target by detecting whether a continuously cast skill belongs to an skill combination, which helps improve accuracy of skill combination casting.

[0156] FIG. 10 is a structural block diagram of a virtual object control apparatus according to an exemplary embodiment of this application. The apparatus includes:

[0157] a display module 1001, configured to display a virtual scene interface, the virtual scene interface including a first virtual object, at least two second virtual objects, and at least two

skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills;

[0158] a first control module 1002, configured to: determine, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and control the first virtual object to cast a first skill on the target virtual object;

[0159] a first determining module 1003, configured to determine the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition, the second trigger operation being triggered during skill casting of the first skill; and

[0160] a second control module 1004, configured to control the first virtual object to cast the second skill on the target virtual object.

[0161] Optionally, the apparatus further includes:

[0162] an inheritance period determining module, configured to determine a target inheritance period corresponding to the target virtual object according to an skill flow corresponding to the first skill and in response to the first trigger operation on the first skill control, the skill flow being used for controlling a cast procedure of the first skill, and the target inheritance period being a valid period for the second skill to inherit the target virtual object; and

[0163] The first determining module 1003 includes:

[0164] an operation receiving unit, configured to receive a trigger operation on the second skill control; and

[0165] a determining unit, configured to: determine, in response to the trigger operation on the second skill control belonging to the second trigger operation, and a triggered time of the second trigger operation is in the target inheritance period, that the second trigger operation meets the skill continuous casting condition, and determine the target virtual object as the skill casting target corresponding to the second skill.

[0166] Optionally, the cast procedure controlled by the skill flow includes an skill casting preparation stage, an skill casting stage, and an skill casting ending stage.

[0167] The inheritance period determining module is configured to:

[0168] determine a duration of the target inheritance period according to a first duration of the skill casting preparation stage and a second duration of the skill casting stage; and

[0169] determine the target inheritance period according to a start execution time and the duration of the skill flow.

[0170] Optionally, the second skill control includes an automatic control region and an active control region, the automatic control region is used for triggering a terminal to determine an skill casting target, direction, or range, and the active control region is used for triggering a user to select an skill casting target, direction, or range.

[0171] The apparatus includes:

[0172] an operation determining module, configured to determine, in response to a touch end position corresponding to the trigger operation on the second skill control being located in the automatic control region, that the trigger operation on the second skill control belongs to the second trigger operation.

[0173] Optionally, the first determining module 1003 further includes:

[0174] a rule obtaining unit, configured to obtain an skill casting range and an skill casting rule of the second skill,

[0175] the determining unit being further configured to determine the target virtual object as the skill casting target corresponding to the second skill in response to the target virtual object being located in the skill casting range and the target virtual object meeting the skill casting rule.

[0176] Optionally, the first determining module 1003 is configured to:

[0177] determine the target virtual object as the skill casting target corresponding to the second skill in response to the target virtual object being located in the skill casting range, the target virtual object meeting the skill casting rule, and an skill casting combination formed by the first skill and the second skill belonging to an skill combination list of the first virtual object, the skill combination list including at least one skill casting combination corresponding to the first virtual object, and the skill casting combination including at least two skills and a casting sequence between the skills.

[0178] Optionally, the apparatus further includes:

[0179] a second determining module, configured to determine the skill casting target of the second skill from the at least two second virtual objects according to the second trigger operation and an skill casting rule and an skill casting range of the second skill in response to the trigger operation on the second skill control not belonging to the second trigger operation, or, the triggered time of the second trigger operation being outside the target inheritance period.

- [0180]** Optionally, the second control module 1004 is configured to:
- [0181]** control the first virtual object to cast the second skill on the target virtual object in response to the second skill being a targeted skill, the targeted skill being an skill with a specified skill casting target; and
- [0182]** determine a relative direction of the target virtual object relative to the first virtual object in response to the second skill being a non-targeted skill; and control the first virtual object to cast the second skill according to the relative direction.
- [0183]** Optionally, the first skill control includes an automatic control region and an active control region, the automatic control region is used for triggering a terminal to determine an skill casting target, direction, or range, and the active control region is used for triggering a user to select an skill casting target, direction, or range.
- [0184]** The first control module 1002 is configured to:
- [0185]** determine the target virtual object according to a target automatic searching rule corresponding to the first skill and in response to the first skill being a targeted skill or a non-targeted skill and a touch end position of the first trigger operation is located in the automatic control region, and cast the first skill; and
- [0186]** determine a second virtual object corresponding to the touch end position as the target virtual object in response to the first skill being a targeted skill and a touch end position of the first trigger operation being located in the active control region, and cast the first skill.
- [0187]** Based on the above, in this embodiment of this application, when a trigger operation on an skill control is received, if the trigger operation is triggered in an skill casting process of a previous skill and the trigger operation meets an skill continuous casting condition, an skill casting target indicated by the previous skill is determined as an skill casting target indicated by the current skill, so as to control a virtual object to cast an skill on the skill casting target. Through the skill casting target inheritance mechanism, a problem that continuously cast skills are cast on different objects because different skills correspond to different skill range can be avoided, and skills may be cast on the same object continuously in an skill continuous casting scenario, thereby improving accuracy of skill casting.
- [0188]** FIG. 11 is a structural block diagram of a terminal according to an embodiment of this application. Generally, a terminal 1100 includes a processor 1101 and a memory 1102.
- [0189]** The processor 1101 may include one or more processing cores. For example, the processor may be a 4-core processor or an 8-core processor. The processor 1101 may be

implemented by using at least one hardware form of a digital signal processor (DSP), a field-programmable gate array (FPGA), and a programmable logic array (PLA). The processor 1101 may alternatively include a main processor and a coprocessor. The main processor is configured to process data in an active state, also referred to as a central processing unit (CPU). The coprocessor is a low-power processor configured to process data in a standby state. In some embodiments, the processor 1101 may be integrated with a graphics processing unit (GPU). The GPU is configured to render and draw content that needs to be displayed on a display screen. In some embodiments, the processor 1101 may further include an artificial intelligence (AI) processor. The AI processor is configured to process a computing operation related to machine learning.

[0190] The memory 1102 may include one or more computer-readable storage media. The computer-readable storage medium may be non-transient. The memory 1102 may further include a high-speed random access memory, and a non-volatile memory such as one or more magnetic disk storage devices and a flash memory device. In some embodiments, a non-transitory computer-readable storage medium in the memory 1102 is configured to store at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction set being configured to be executed by the processor 1101 to implement the method provided in the method embodiments of this application.

[0191] In some embodiments, the terminal 1100 may optionally include a peripheral interface 1103 and at least one peripheral. The processor 1101, the memory 1102, and the peripheral interface 1103 may be connected through a bus or a signal cable. Each peripheral may be connected to the peripheral interface 1103 through a bus, a signal cable, or a circuit board. Specifically, the peripheral may include at least one of a communication interface 1104, a display screen 1105, an audio circuit 1106, a camera component 1107, a positioning component 1108, and a power supply 1109.

[0192] A person skilled in the art may understand that the structure shown in FIG. 11 constitutes no limitation on the terminal 1100, and the terminal may include more or fewer components than those shown in the figure, or some components may be combined, or a different component deployment may be used.

[0193] An embodiment of this application further provides a computer-readable storage medium, storing at least one instruction, the at least one instruction being loaded and executed by a processor to implement the virtual object control method according to the foregoing embodiments.

[0194] An embodiment of this application further provides a computer program product, storing at least one instruction, the at least one instruction being loaded and executed by a processor to implement the virtual object control method according to the foregoing embodiments.

[0195] A person skilled in the art should be aware of that in the one or more examples, the functions described in the embodiments of this application may be implemented by hardware, software, firmware, or a combination of the above. When implemented by using software, the functions can be stored in a computer-readable storage medium or can be used as one or more instructions or code in a computer-readable storage medium for transferring. The computer-readable storage medium includes a computer storage medium and a communication medium, where the communication medium includes any medium that enables a computer program to be transmitted from one place to another. The storage medium may be any available medium accessible to a general-purpose or dedicated computer.

[0196] The foregoing descriptions are merely optional embodiments of this application, but are not intended to limit this application. Any modification, equivalent replacement, or improvement made within the spirit and principle of this application shall fall within the protection scope of this application.

CLAIMS

What is claimed is:

1. A virtual object control method, performed by a terminal, the method comprising:

displaying a virtual scene interface, the virtual scene interface comprising a first virtual object, at least two second virtual objects, and at least two skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills;

determining, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and controlling the first virtual object to cast a first skill on the target virtual object;

determining the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition, the second trigger operation being triggered during skill casting of the first skill; and

controlling the first virtual object to cast the second skill on the target virtual object.

2. The method according to claim 1, further comprising:

determining a target inheritance period corresponding to the target virtual object according to an skill flow corresponding to the first skill and in response to the first trigger operation, the skill flow being used for controlling a cast procedure of the first skill, and the target inheritance period being a valid period for the second skill to inherit the target virtual object; and

the determining the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition comprises:

receiving a trigger operation on the second skill control; and

determining, in response to the trigger operation on the second skill control belonging to the second trigger operation, and a triggered time of the second trigger operation being in the target inheritance period, that the second trigger operation meets the skill continuous casting condition, and determining the target virtual object as the skill casting target corresponding to the second skill.

3. The method according to claim 2, wherein the cast procedure controlled by the skill flow comprises an skill casting preparation stage, an skill casting stage, and an skill casting ending stage; and

the determining a target inheritance period corresponding to the target virtual object according to an skill flow corresponding to the first skill comprises:

determining a duration of the target inheritance period according to a first duration of the skill casting preparation stage and a second duration of the skill casting stage; and

determining the target inheritance period according to a start execution time and the duration of the skill flow.

4. The method according to claim 2, wherein the second skill control comprises an automatic control region and an active control region, the automatic control region being used for triggering a terminal to determine an skill casting target, direction, or range, and the active control region being used for triggering a user to select an skill casting target, direction, or range; and

after the receiving a trigger operation on the second skill control, the method further comprises:

determining, in response to a touch end position corresponding to the trigger operation on the second skill control being located in the automatic control region, that the trigger operation on the second skill control belongs to the second trigger operation.

5. The method according to any one of claims 2 to 4, wherein the determining the target virtual object as an skill casting target corresponding to a second skill comprises:

obtaining an skill casting range and an skill casting rule of the second skill; and

determining the target virtual object as the skill casting target corresponding to the second skill in response to the target virtual object being located in the skill casting range and the target virtual object meeting the skill casting rule.

6. The method according to claim 5, wherein the determining, the target virtual object as the skill casting target corresponding to the second skill in response to the target virtual object being located in the skill casting range and the target virtual object meets the skill casting rule comprises:

determining the target virtual object as the skill casting target corresponding to the second skill in response to the target virtual object being located in the skill casting range, the target virtual object meeting the skill casting rule, and an skill casting combination formed by the first skill and the second skill belonging to an skill combination list of the first virtual object, the skill combination list comprising at least one skill casting combination corresponding to the first virtual object, and the skill casting combination comprising at least two skills and a casting sequence between the skills.

7. The method according to any one of claims 2 to 4, further comprising:

determining the skill casting target of the second skill from the at least two second virtual objects according to the second trigger operation and an skill casting rule and an skill casting range of the second skill in response to the trigger operation on the second skill control not belonging to the second trigger operation.

8. The method according to any one of claims 2 to 4, further comprising:

determining the skill casting target of the second skill from the at least two second virtual objects according to the second trigger operation and an skill casting rule and an skill casting range of the second skill in response to the triggered time of the second trigger operation being outside the target inheritance period.

9. The method according to any one of claims 1 to 4, wherein the controlling the first virtual object to cast the second skill on the target virtual object comprises:

controlling the first virtual object to cast the second skill on the target virtual object in response to the second skill being a targeted skill, the targeted skill being an skill with a specified skill casting target.

10. The method according to any one of claims 1 to 4, wherein the controlling the first virtual object to cast the second skill on the target virtual object comprises:

determining a relative direction of the target virtual object relative to the first virtual object in response to the second skill being a non-targeted skill; and controlling the first virtual object to cast the second skill according to the relative direction.

11. The method according to any one of claims 1 to 4, wherein the first skill control comprises an automatic control region and an active control region, the automatic control region being used for triggering a terminal to determine an skill casting target, direction, or range, and the active control region being used for triggering a user to select an skill casting target, direction, or range; and

the controlling the first virtual object to cast a first skill on the target virtual object comprises:

determining the target virtual object according to a target automatic searching rule corresponding to the first skill and in response to the first skill being a targeted skill or a non-targeted skill and a touch end position of the first trigger operation being located in the automatic control region, and casting the first skill.

12. The method according to any one of claims 1 to 4, wherein the first skill control comprises

an automatic control region and an active control region, the automatic control region being used for triggering a terminal to determine an skill casting target, direction, or range, and the active control region being used for triggering a user to select an skill casting target, direction, or range; and

the controlling the first virtual object to cast a first skill on the target virtual object comprises:

determining a second virtual object corresponding to the touch end position as the target virtual object in response to the first skill being a targeted skill and a touch end position of the first trigger operation being located in the active control region, and casting the first skill.

13. A virtual object control apparatus, comprising:

a display module, configured to display a virtual scene interface, the virtual scene interface comprising a first virtual object, at least two second virtual objects, and at least two skill controls, the first virtual object and the second virtual objects being located in a virtual scene, and the skill controls being configured to control the first virtual object to cast skills;

a first control module, configured to: determine, in response to a first trigger operation on a first skill control, a target virtual object from the at least two second virtual objects and according to the first trigger operation, and control the first virtual object to cast a first skill on the target virtual object;

a first determining module, configured to determine the target virtual object as an skill casting target corresponding to a second skill in response to a second trigger operation on a second skill control meeting an skill continuous casting condition, the second trigger operation being triggered during skill casting of the first skill; and

a second control module, configured to control the first virtual object to cast the second skill on the target virtual object.

14. A terminal, comprising a processor and a memory, the memory storing at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction set being loaded and executed by the processor to implement the virtual object control method according to any one of claims 1 to 12.

15. A computer-readable storage medium, storing at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction set being loaded and executed by a processor to implement the virtual object control method according to any one of claims 1 to 12.