PET VIRTUAL CHARACTER CAPTURE METHOD AND APPARATUS, AND DEVICE, MEDIUM AND PRODUCT

RELATED APPLICATION

[0001] This application claims priority to Chinese Patent Application No. 202411295360.9, entitled "PET VIRTUAL CHARACTER CAPTURE METHOD AND APPARATUS, AND DEVICE, MEDIUM AND PRODUCT", and filed on September 13, 2024, which is incorporated herein by reference in its entirety.

FIELD OF THE TECHNOLOGY

[0002] Embodiments of this application relate to the field of human-computer interaction, and in particular, to a pet virtual character capture method and apparatus, a device, a medium, and a product.

BACKGROUND OF THE DISCLOSURE

[0003] A Role-Playing Game (RPG) is a game type. In this type of game, a player controls a main control virtual character to explore or perform activities in a virtual world. In some embodiments, the player controls the main control virtual character to capture (tame) a pet virtual character in the virtual world.

[0004] In a related technology, a capture mechanism is provided. When capturing a pet virtual character, a player controls a main control virtual character to attack the pet virtual character, throws a capture prop to the pet virtual character when an attribute value of the pet virtual character decreases, and captures the pet virtual character by using the capture prop.

[0005] However, in the related technology, capture performance for a pet virtual character is monotonous.

SUMMARY

[0006] This application provides a pet virtual character capture method and apparatus, a device, a medium, and a product. The technical solutions are as follows:

[0007] According to an aspect of this application, a pet virtual character capture method is provided. The method is performed by a terminal and includes:

displaying a main control virtual character located in a virtual world, and a pet virtual character in a wild state;

controlling, in response to a capture operation on the pet virtual character, the main control virtual character to release a capture prop or a capture skill for the pet virtual character; and

displaying a challenge capture effect for the pet virtual character, the challenge capture effect including a struggle capture effect or a struggle escape effect; and the struggle capture effect being configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles, and the struggle escape effect being configured for representing that the pet virtual character escapes from capture after the pet virtual character struggles.

[0008] According to an aspect of this application, a pet virtual character capture apparatus is provided. The apparatus includes:

a display module, configured to display a main control virtual character located in a virtual world, and a pet virtual character in a wild state; and a control module, configured to control, in response to a capture operation on the pet virtual character, the main control virtual character to release a capture prop or a capture skill for the pet virtual character,

the display module being configured to display a challenge capture effect for the pet virtual character, the challenge capture effect including a struggle capture effect or a struggle escape effect; and the struggle capture effect being configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles, and the struggle escape effect being configured for representing that the pet virtual character escapes from capture after the pet virtual character struggles.

[0009] According to another aspect of this application, a computer device is provided, the computer device including: a processor and a memory, the memory having at least one computer program stored therein, and the at least one computer program being loaded and executed by the processor to implement the pet virtual character capture method according to the foregoing aspects.

[0010] According to another aspect of this application, a computer storage medium is provided. The computer storage medium has at least one computer program stored therein, the at least one computer program being loaded and executed by a processor to implement the pet virtual character capture method according to the foregoing aspects.

[0011] According to another aspect of this application, a computer program product is provided. The computer program product includes a computer program. The computer program is stored in a computer-readable storage medium. The computer program is read and executed from

the computer-readable storage medium by a processor of a computer device, to enable the computer device to perform the pet virtual character capture method according to the foregoing aspects.

[0012] The technical solutions provided in the embodiments of this application produce at least the following beneficial effects:

[0013] After the main control virtual character releases a capture prop or a capture skill, a challenge capture effect of the pet virtual character in the virtual world is displayed. The challenge capture effect includes a struggle capture effect or a struggle escape effect. By displaying a virtual struggle behavior of the pet virtual character with the capture prop or on the capture skill, it is indicated that the pet virtual character is successfully captured or struggles to escape after the pet virtual character struggles, thereby visually presenting uncertainty and a challenge of capturing the pet virtual character. A struggle behavior in the real world is simulated, and more real capture experience is provided for the player.

BRIEF DESCRIPTION OF THE DRAWINGS

[0014] FIG. 1 is a schematic architectural diagram of a computer system according to an embodiment of this application.

[0015] FIG. 2 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0016] FIG. 3 is a flowchart of a pet virtual character capture method according to an embodiment of this application.

[0017] FIG. 4 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0018] FIG. 5 is a flowchart of a pet virtual character capture method according to an embodiment of this application.

[0019] FIG. 6 is a flowchart of a pet virtual character capture method according to an embodiment of this application.

[0020] FIG. 7 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0021] FIG. 8 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0022] FIG. 9 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0023] FIG. 10 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0024] FIG. 11 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0025] FIG. 12 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0026] FIG. 13 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0027] FIG. 14 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0028] FIG. 15 is an overall flowchart of a pet virtual character capture method according to an embodiment of this application.

[0029] FIG. 16 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0030] FIG. 17 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0031] FIG. 18 is a schematic diagram of a pet virtual character capture method according to an embodiment of this application.

[0032] FIG. 19 is a structural block diagram of a pet virtual character capture apparatus according to an embodiment of this application.

[0033] FIG. 20 is a schematic structural diagram of a computer device according to an embodiment of this application.

DESCRIPTION OF EMBODIMENTS

[0034] The terms used in this application are merely for the purpose of illustrating specific embodiments, and are not intended to limit this application. The terms "a", "said" and "the" of singular forms used in this application and the appended claims are also intended to include plural forms, unless otherwise specified in the context clearly. The term "and/or" used in this specification indicates and includes any or all possible combinations of one or more associated listed items.

Although terms "first", "second", and the like may be used in this application to describe various pieces of information, these pieces of information are not limited by the terms. These terms are merely configured for differentiating the same type of information. For example, without departing from the scope of this application, a first parameter may also be referred to as a second parameter, and similarly, a second parameter may also be referred to as a first parameter. Depending on the context, for example, a word "if" used herein may be explained as "while …" or "when …" or "in response to determining". In this application, before collection of relevant data of the user and during the collection of the relevant data of the user, a prompt interface or a pop-up window may be displayed, or audio prompt information may be outputted. The prompt interface, the pop-up window, or the audio prompt information is configured for prompting the user that the relevant data of the user is currently searched for collection. In this way, in this application, only after a confirmation operation transmitted by the user for the prompt interface or the pop-up window is obtained, a relevant operation of obtaining the relevant data of the user is started to be performed. Otherwise (in other words, the confirmation operation transmitted by the user for the prompt interface or the pop-up window is not obtained), the relevant operation of obtaining the relevant data of the user is ended, in other words, the relevant data of the user is not obtained. In other words, all user data collected in this application is collected with user consent and authorization, and collection, use, and processing of relevant user data need to comply with relevant laws, regulations, and standards of relevant countries and regions.

[0035] First, relevant terms involved in this application are introduced:

[0036] **Virtual world**: It is a virtual world displayed (or provided) when an application runs on a terminal. The virtual world may be a simulation world of the real world, a semi-simulation and semi-fictitious three-dimensional world, or a purely fictitious three-dimensional world. The virtual world can be any one of a two-dimensional virtual world, a 2.5-dimensional virtual world, and a three-dimensional virtual world. This is not limited in this application. In some embodiments, the virtual world is further configured for allowing for a virtual environment battle between at least two virtual characters, and there are virtual resources available for use by the at least two virtual characters in the virtual world.

[0037] **Main control virtual character**: It is a movable object played by a player in a virtual world. The main control virtual character can be a virtual person, a virtual animal, or an animation character, for example, a person or an animal displayed in a three-dimensional virtual world. In some embodiments, the main control virtual character is a three-dimensional model created based on an animation skeleton technology. Each main control virtual character has a shape

and a volume in the three-dimensional virtual world and occupies a part of the space in the three-dimensional virtual world.

[0038] **Pet virtual character**: It is a movable object controlled in a virtual world. In some embodiments, the pet virtual character is controlled by Artificial Intelligence (AI), or the pet virtual character is controlled by a state machine, or the pet virtual character is controlled by another player. This is not limited in this application. The pet virtual character may be a virtual creature, a virtual animal, a virtual monster, a virtual sprue, a virtual pet, or the like, for example, a movable object in an animal form or another form displayed in a three-dimensional virtual world. The main control virtual character can perform at least one of interactive operations of capturing, raising, upgrading, and the like on the pet virtual character. The pet virtual character can assist the main control virtual character in performing at least one of interactive operations of collecting, fighting, plot changing, and the like.

[0039] FIG. 1 is a structural block diagram of a computer system according to an exemplary embodiment of this application. The computer system 100 includes a first terminal 120, a server 140, a second terminal 160, and a third terminal 180.

[0040] A client of an application supporting a virtual world is installed and run in the first terminal 120. The application can be any one of a three-dimensional map program, a virtual reality (VR) application, an augmented reality (AR) program, an RPG program, a turn-based game program, and a turn-based RPG program. The first terminal 120 is a terminal used by a first user. The first user uses the first terminal 120 to control a first virtual character in the virtual world to perform activities. The first virtual character serves as a main control virtual character. The activities include but are not limited to: at least one of adjusting a body posture, walking, running, jumping, riding, driving, aiming, picking up, capturing a pet virtual character, controlling a pet virtual character, raising a pet virtual character, using a pet virtual character to pick, using a pet virtual character to fight, using a throwing prop, and attacking other virtual characters. For example, the first virtual character is a first virtual person, such as a simulated person object or an animation person object. For example, the first user controls the first virtual character through a UI control on a virtual world picture to perform activities, and the first user controls the first virtual character to throw a pet virtual character through the UI control on the virtual world picture.

[0041] The first terminal 120 is connected to the server 140 over a wireless network or a wired network.

[0042] The server 140 includes at least one of a server, a plurality of servers, a cloud computing platform, or a virtualization center. For example, the server 140 includes a processor 144

and a memory 142. The memory 142 further includes a receiving module 1421, a control module 1422, and a sending module 1423. The receiving module 1421 is configured to receive a request sent by a client, such as a request for viewing a position of an energy virtual character. The control module 1422 is configured to control rendering of a virtual world picture. The sending module 1423 is configured to send a response to the client, for example, send a position of a third virtual character to the client. The server 140 is configured to provide background services for applications that support the three-dimensional virtual world. In some embodiments, the server 140 undertakes main computing work, and the first terminal 120, the second terminal 160, and the third terminal 180 undertake secondary computing work. Alternatively, the server 140 undertakes secondary computing work, and the first terminal 120, the second terminal 160, and the third terminal 180 undertake main computing work. Alternatively, the server 140, the first terminal 120, the second terminal 160, and the third terminal 180 perform collaborative computing by using a distributed computing architecture.

[0043] A client of an application supporting a virtual world is installed and run in the second terminal 160. The second terminal 160 is a terminal used by a second user. The second user uses the second terminal 160 to control a second virtual character in the virtual world to perform activities. The second virtual character also serves as a main control virtual character. An application supporting a virtual world is installed and run in the third terminal 180. The third terminal 180 is a terminal used by a third user. The third user uses the third terminal 180 to control a third virtual character in the virtual world to perform activities. In some embodiments, the first virtual character, the second virtual character, and the third virtual character are in the same virtual world. The first virtual character and the second virtual character belong to different camps, and the second virtual character and the third virtual character belong to the same camp. In some embodiments, the applications installed on the first terminal 120, the second terminal 160, and the third terminal 180 are the same, or the applications installed on the three terminals are the same type of application on different operating system platforms (Android or IOS). The first terminal 120 may generally refer to one of a plurality of terminals, the second terminal 160 may generally refer to one of a plurality of terminals, and the third terminal 180 may generally refer to one of a plurality of terminals. This embodiment only uses the first terminal 120, the second terminal 160, and the third terminal 180 for illustration. Device types of the first terminal 120, the second terminal 160, and the third terminal 180 are the same or different. The device type includes at least one of a smartphone, a smart watch, a smart TV, a tablet computer, an e-book reader, an MP3 player, an MP4 player, a laptop computer, and a desktop computer. In the following embodiments, for example, the terminal includes a smartphone. A person skilled in the art knows that the number of the above terminals may be more

or less. For example, there may be only one terminal, or there may be dozens of or hundreds of terminals or more. The quantity and device types of terminals are not limited in the embodiments of this application.

[0044] For example, FIG. 2 is a schematic diagram of a pet virtual character capture method according to an exemplary embodiment of this application. An example in which the method is performed by a client of an application supporting a virtual world is configured for description. In a related technology, a capture mechanism is provided. When capturing a pet virtual character, a player controls a main control virtual character to attack the pet virtual character, throws a capture prop to the pet virtual character when an attribute value of the pet virtual character decreases, and captures the pet virtual character by using the capture prop. However, in the related technology, capture performance for a pet virtual character is monotonous.

[0045] For the foregoing problems, the embodiments of this application provide a pet virtual character capture method. The method may be performed by a terminal or a client on a terminal. In some embodiments, a player controls a main control virtual character to release a capture prop for a pet virtual character. Capture props are classified into different levels, and capture props of different levels correspond to different capture forces. The difficulty of obtaining a capture prop increases sequentially with an increase in level. After a game starts, a capture prop control is displayed on the client. By triggering the capture prop control, an exclusive crosshair is displayed, the aiming crosshair is aimed at a pet virtual character, and an aiming circle for the pet virtual character is displayed. In some embodiments, the aiming circle may be displayed in different colors, and the player may determine, according to color information displayed in the aiming circle, difficulty in capturing the pet virtual character, to determine a level of a used capture prop. After the capture prop hits the pet virtual character, capture determining is performed, a capture type (a successful capture type, a challenge capture type, and a failed capture type) is determined according to a capture value, and a corresponding capture effect is displayed on a game interface. The capture effect includes at least one of a no-struggle capture effect, a struggle capture effect, a struggle escape effect, and a no-struggle failure effect. For example, as shown in (a) of FIG. 2, a virtual world picture that a main control virtual character 10 is located in a virtual world is displayed on the client, and the virtual world picture further includes at least one pet virtual character 20 in a wild state. The pet virtual character 20 is a movable object in the virtual world. In some embodiments, the main control virtual character 10 can perform, but not limited to, at least one of interactive operations of capturing, raising, upgrading, and the like on the pet virtual character 20. This is not limited in this application. In some embodiments, the pet virtual character 20 can assist the main

control virtual character 10 in performing, but not limited to, at least one of interactive operations of collecting, fighting, plot changing, and the like. This is not limited in this application.

[0046] As shown in (a) of FIG. 2, a capture prop control 30 is further displayed on the virtual world picture, and the capture prop control is a control for controlling the captured pet virtual character 20 by the main control virtual character 10. In response to a touch operation on the capture prop control 30, an aiming crosshair 40 corresponding to the capture prop control 30 is displayed, and the aiming crosshair 40 is configured for assisting the main control virtual character 10 in aiming at a virtual aiming object. In some embodiments, the virtual aiming object may be any object, for example, a pet virtual character, terrain, or a building.

[0047] In some embodiments, in response to an aiming operation on the pet virtual character 20, that is, when the virtual aiming object is the pet virtual character 20 to be captured, the aiming crosshair 40 is aimed at the pet virtual character 20. In this case, an aiming circle 50 for the pet virtual character 20 is displayed, and a level of a capture prop released by the main control virtual character is determined based on prompt information of the aiming circle 50. Levels of capture props may be classified into different levels according to obtaining difficulties. In some embodiments, capture props of the same parent level include capture props of child levels. For example, levels of capture props may be divided into an ordinary level, an uncommon level, and a rare level according to obtaining difficulties. For example, capture props of the ordinary level may be further subdivided into a capture prop of an ordinary child level, a capture prop of an advanced child level, and a capture prop of an excellent child level. In some embodiments, capture props of different levels correspond to different capture capabilities. The foregoing is merely exemplary description, and does not constitute a limitation on the division of capture props. In some embodiments, the prompt information displayed based on the aiming circle 50 includes at least one of the following: color information displayed based on the aiming circle; style information displayed based on the aiming circle; icon information displayed based on a first peripheral side position of the aiming circle; and recommendation information displayed based on a second peripheral side position of the aiming circle.

[0048] In some embodiments, the prompt information of the aiming circle 50 includes color information. In some embodiments, the aiming circle 50 is displayed as different color information, and the different color information represents different capture difficulties. The player determines, according to color information displayed in the aiming circle 50, difficulty in capturing the pet virtual character 20, to determine a level of a used capture prop 60. When the color information of the aiming circle 50 is a first color, the main control virtual character 10 is guided in selecting a

capture prop of a first level, and the first level is a level of the current capture prop of the main control virtual character 10. When the color information of the aiming circle 50 is a second color, the main control virtual character 10 is guided in switching a capture prop of a second level to a capture prop of a third level, and a grade of the third level is higher than a grade of the second level. When the color information of the aiming circle 50 is a third color, the main control virtual character 10 is guided in switching a capture prop of a fourth level to a capture prop of a fifth level, and a grade of the fifth level is lower than a grade of the fourth level. When the color information of the aiming circle 50 is a fourth color, the main control virtual character 10 is guided in selecting a capture prop of a preset level, and the preset level is a level of a specified capture prop. For example, as shown in (b) of FIG. 2, after an aiming crosshair 40 is aimed at a pet virtual character 20, an aiming circle 50 is displayed. For example, color information of the aiming circle 50 is displayed as red, which represents that a capture difficulty of the pet virtual character 20 is high, that is, a possibility that the main control virtual character 10 successfully captures the pet virtual character by using a capture prop 60 is low. In some embodiments, the player controls the main control virtual character 10 to switch a level of the pre-thrown capture prop 60. For example, the initial level of the pre-thrown capture prop 60 is an ordinary level, and the player may choose to switch the level of the pre-thrown capture prop 60 from the ordinary level to an uncommon level, thereby improving a possibility of successful capture.

[0049] In a possible implementation, the main control virtual character 10 is controlled to release a capture skill, to capture the pet virtual character 20. In some embodiments, a capture skill control is displayed on the virtual world picture, and in response to a trigger operation on the capture skill control, that the main control virtual character 10 releases a capture skill for the pet virtual character 20 is displayed. In some embodiments, the capture skill is presented as a capture spell-casting effect. For example, the capture spell-casting effect is presented as at least one of an attack spell-casting effect, a feeding spell-casting effect, a curing spell-casting effect, a hypnotic spell-casting effect, an interaction spell-casting effect, and an attribute counteraction spell-casting effect. In some embodiments, after the pre-thrown capture prop 60 hits the pet virtual character 20, or after the capture spell-casting effect is released for the pet virtual character 20, capture determining is performed, a capture type is determined according to a capture value, and a corresponding capture effect is displayed on the game interface. The capture effect includes at least one of a not-struggle capture effect, a struggle capture effect, a struggle escape effect, and a no-struggle failure effect. The successful capture effect is configured for representing that the pet virtual character 20 does not struggle and is successfully captured. The struggle capture effect is configured for representing that the pet virtual character 20 is successfully captured after the pet

virtual character struggles. The struggle escape effect is configured for representing that the pet virtual character 20 escapes from capture after the pet virtual character struggles. The failed capture effect is configured for representing that the pet virtual character 20 does not struggle and fails to be captured. In some embodiments, in response to a throwing operation on the pre-thrown capture prop 60, after the pre-thrown capture prop 60 hits the pet virtual character 20, or after a capture spell-casting effect is released for the pet virtual character 20, a capture value for the pet virtual character 20 is calculated, and the capture value is configured for indicating a capture probability of capturing the pet virtual character 20. A probability interval to which the capture probability belongs is determined according to the capture probability, and capture effects for the pet virtual character 20 in different probability intervals are displayed.

[0050] In some embodiments, the probability interval is configured for indicating a capture type for the pet virtual character 20. The challenge capture effect for the pet virtual character 20 is displayed when a capture probability of the pet virtual character 20 falls within a target probability interval.

[0051] In some embodiments, when the capture probability of the pet virtual character 20 falls within a first probability interval, it is determined that a capture type for the pet virtual character 20 is a successful capture type and a capture effect for the pet virtual character 20 is displayed as a successful capture effect, the successful capture effect being configured for representing that the pet virtual character 20 does not struggle and is successfully captured; when the capture probability of the pet virtual character 20 falls within a second probability interval, it is determined that a capture type for the pet virtual character 20 is a successful capture type and a capture effect for the pet virtual character 20 is displayed as a struggle capture effect, the struggle capture effect being configured for representing that the pet virtual character 20 is successfully captured after the pet virtual character struggles; and when the capture probability of the pet virtual character 20 falls within a third probability interval, it is determined that a capture type for the pet virtual character 20 is a failed capture type and a capture effect for the pet virtual character 20 is displayed as a failed capture effect, the failed capture effect being configured for representing that the pet virtual character 20 does not struggle and fails to be captured. In some embodiments, a minimum value of the first probability interval is greater than a maximum value of the second probability interval, a minimum value of the second probability interval is greater than a maximum value of the target probability interval, and a minimum value of the target probability interval is greater than a maximum value of the third probability interval. In some embodiments, as the capture value decreases, a capture probability that the main control virtual character 10 captures the pet

virtual character 20 also decreases accordingly. In some embodiments, within a corresponding probability interval, a capture effect for the pet virtual character 20 is also reduced accordingly.

[0052] For example, when the capture probability falls within the first probability interval, the capture type for the pet virtual character 20 is a successful capture type, and a successful capture effect is presented as that the capture prop 60 directly succeeds in taming the pet virtual character 20, and the pet virtual character 20 has no struggle behavior. When the capture probability falls within the second probability interval, the capture type for the pet virtual character 20 is a successful capture type, and a struggle capture effect is presented as that the capture prop 60 succeeds in taming the pet virtual character 20, and the pet virtual character 20 has a struggle behavior. When the capture probability falls within the target probability interval, the capture type for the pet virtual character 20 is a challenge capture type, and the challenge capture type includes a struggle capture effect or a struggle escape effect. The struggle capture effect is presented as that the pet virtual character 20 has a struggle behavior, and is successfully captured after the pet virtual character struggles; or the struggle escape effect is presented as that the pet virtual character 20 has a struggle behavior, and escapes from capture after the pet virtual character struggles. In some embodiments, in this case, n random probabilities (or referred to as random numbers) are generated. A value range of the n random probabilities falls within the target probability interval. The capture probability is compared with the n random probabilities successively. A count of struggle of the pet virtual character 20 in the capture prop 60 is determined based on a comparison result. If the count of struggle is greater than a count threshold, it is determined that a capture effect of the pet virtual character 20 is a struggle escape effect; otherwise, the capture effect is a struggle capture effect. For example, the target probability interval is (60%, 40%)], and the capture probability is 50%. Using an example in which n is 4, the generated 4 random probabilities are respectively 58%, 55%, 52%, and 43%. The capture probability is compared with the 4 random probabilities successively. The count at which the capture probability is greater than the n random probabilities is 1, that is, the count at which the pet virtual character 20 struggles in the capture prop 60 is 1. Using an example in which the count threshold is 3, the count of struggle is less than the count threshold (1 is less than 3), that is, a capture effect of the pet virtual character 20 is presented as a struggle capture effect. When the capture probability falls within the third probability interval, a capture type for the pet virtual character 20 is a failed capture type, the failed capture effect being configured for representing that the pet virtual character does not struggle and fails to be captured. For example, the failed capture effect is presented as that the pet virtual character does not struggle, and the capture prop 60 directly ricochets. In some embodiments, in response to a throwing operation on the pre-thrown capture prop 60, after the pre-thrown capture prop 60 hits an illegal unit, a text prompt

of uncapturable is displayed on the game interface. The illegal unit refers to a non-pet virtual character, such as a stone, a tree, or a lawn.

[0053] For example, as shown in (c) of FIG. 2, the main control virtual character 10 captures the pet virtual character 20 based on the capture prop 60. When the capture probability falls within the first probability interval, a capture type for the pet virtual character 20 is a successful capture type, it is displayed that a successful capture effect is presented as that the pet virtual character 20 has no struggle behavior, the capture prop 60 successfully tames the pet virtual character 20, and the pet virtual character 20 is directly changed into a starlight and is inhaled by the capture prop 60.

[0054] In some embodiments, a basic capture value for the pet virtual character 20 is obtained, the basic capture value being configured for indicating a basic capture probability of capturing the pet virtual character 20. In some embodiments, in a process in which the main control virtual character 10 captures the pet virtual character 20, there are further some correction types that affect magnitude of a basic capture value, and further affect a basic capture probability. In some embodiments, a correction type is obtained, and the basic capture value is corrected based on content indicated by the correction type to obtain a corrected basic capture probability. The corrected basic capture probability is a capture probability, that is, the capture probability is obtained based on the content indicated by the correction type.

[0055] In some embodiments, the correction type includes at least one of the following: state correction, configured for indicating correction of the basic capture value by a state of the pet virtual character 20; gain correction, configured for indicating correction of the basic capture value by a positive gain effect and/or a negative gain effect; level suppression correction, configured for indicating correction of the basic capture value by a level difference between the main control virtual character 10 and the pet virtual character 20; continuous capture compensation correction, configured for indicating correction of the basic capture value by a continuous capture operation performed by the main control virtual character 10; and capture force correction, configured for indicating correction of the basic capture value by a capture capability of the main control virtual character 10. For example, the correction type is state correction. When the pet virtual character 20 is in a low health point state, a basic capture value of the pet virtual character 20 increases, that is, a capture probability of capturing the pet virtual character 20 increases. For another example, the correction type is gain correction. When the pet virtual character 20 is a fire pet virtual character, an attribute value becomes weaker when the pet virtual character 20 moves to a lake side. In this case, the main control virtual character 10 obtains a positive gain effect, that is, a capture probability of

capturing the pet virtual character 20 is increased. For another example, the correction type is level suppression correction. When the level of the main control virtual character 10 is higher than the level of the pet virtual character 20, the capture probability of capturing the pet virtual character 20 increases; otherwise, the capture probability decreases.

[0056] FIG. 3 is a flowchart of a pet virtual character capture method according to an exemplary embodiment of this application. That the method is applied to the first terminal 120 and/or the second terminal 160 and/or the third terminal 180 shown in FIG. 1, or to a client that supports a virtual world and that is installed on the first terminal 120 and/or the second terminal 160 and/or the third terminal 180 is used as an example for description. The method includes at least operation 210, operation 220, and operation 230 as follows:

[0057] **Operation 210**: Display a main control virtual character located in a virtual world, and a pet virtual character in a wild state.

[0058] In some embodiments, a client supporting a virtual environment is run on a terminal device, a virtual world picture that a main control virtual character is located in a virtual world is displayed on a game interface of the client, and the virtual world picture further includes at least one pet virtual character in a wild state. The main control virtual character is a virtual character playing an attacker character in a game. In some embodiments, the main control virtual character may be a virtual character controlled by a player, or the main control virtual character is a virtual character controlled by AI. This embodiment of this application is described by using an example in which the main control virtual character is a virtual character controlled by a player. In some embodiments, a player manipulates a main control virtual character to play a game or interact with a virtual environment. The pet virtual character in a wild state is a virtual object having a virtual mobility in a virtual world. The wild state refers to that a pet virtual character existing in a virtual world is free to move in the virtual world according to a behavior mode of voluntarily performing a virtual activity before the pet virtual character is captured and tamed by a player. In some embodiments, the pet virtual character in a wild state is at least one of a Non-Player Character (NPC), a character controlled by AI, and a to-be-captured pet virtual character. In some embodiments, the pet virtual character may be displayed in any possible form. For example, the form of the pet virtual character is related to a level, an environment (weather, time, and a district), a used prop, and the like of the pet virtual character. For example, at least one of activity information such as a type of a virtual activity executed when an activity is voluntarily executed and a scope in which the virtual activity is executed may be preset in a process of developing a virtual world.

[0059] In some embodiments, the pet virtual character may be a to-be-captured pet virtual character. In some embodiments, the main control virtual character can perform, but not limited to, at least one of interactive operations of capturing, raising, upgrading, and the like on the pet virtual character. This is not limited in this application. In some embodiments, the pet virtual character can assist the main control virtual character in performing, but not limited to, at least one of interactive operations of collecting, fighting, plot changing, and the like. This is not limited in this application.

[0060] In some embodiments, for a to-be-captured pet virtual character, the pet virtual character may be at any possible position in a virtual world. The position of the pet virtual character may be associated with a type, an attribute, and the like of the pet virtual character. For example, a pet virtual character is a grass character, and is mostly distributed near a lawn, a jungle, or the like in a virtual world. A pet virtual character is a water character, and is mostly distributed near a lake, a sea, and the like in a virtual world. A position of the pet virtual character may be randomly set.

[0061] In a possible implementation, the terminal displays the virtual environment by using the virtual environment picture. In some embodiments, the virtual environment picture is a picture of the virtual environment that is observed from a perspective of a main control virtual character. The perspective is an observation angle at which observation is performed in the virtual environment from a first-person perspective or a third-person perspective. In some embodiments, in this embodiment of this application, the perspective is an angle for observing the main control virtual character by using a camera model in the virtual environment. When the first-person perspective is used, the camera model is located near the head of the main control virtual character or at the position of the head of the main control virtual character. When the third-person perspective is used, the camera model may be located behind the main control virtual character and bound to the main control virtual character, or may be located at any position from the main control virtual character by a preset distance. The main control virtual character located in the virtual environment may be observed from different angles by using the camera model. In some embodiments, in addition to the first-person perspective and the third-person perspective, the perspective further includes another perspective, such as a top perspective. When the top perspective is used, the camera model may be located above the head of the main control virtual character, and the top perspective is a perspective in which the virtual environment is observed from a perspective of looking up in air. In some embodiments, the camera model is not actually displayed in a game picture. That is, the camera model is not displayed in the virtual environment displayed on the game interface.

[0062] **Operation 220**: Control, in response to a capture operation on the pet virtual character, the main control virtual character to release a capture prop or a capture skill for the pet virtual character, the capture operation being an operation that the player controls the main control virtual character to capture the pet virtual character. In some embodiments, the capture operation includes releasing a capture prop or a capture skill for the pet virtual character.

[0063] Capture manner I: The main control virtual character releases a capture prop for a pet virtual character.

[0064] In some embodiments, a capture prop control is displayed on the game interface in response to a first capture trigger operation. The first capture trigger operation is an operation that is triggered by a player to fight against a to-be-captured pet virtual character, to capture the pet virtual character by using a capture prop. The capture prop control is a control for controlling the captured pet virtual character by the main control virtual character. The capture prop control is an interface interaction element that is in the game interface and that is configured for triggering a capture operation. In some embodiments, the capture prop control is usually presented in an icon, a button, text, or another form. This is not limited in this application. For example, the capture prop control may be a capture prop button. In some embodiments, in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture prop for the pet virtual character. That is, in response to a trigger operation on the capture prop control, the main control virtual character is controlled to release a capture prop for the pet virtual character. Capture props may be classified into different levels. In some embodiments, the levels of the capture props are classified based on obtained difficulties. The difficulty of obtaining a capture prop increases sequentially with an increase in level. The levels of the capture props include a capture prop of a parent level type and a capture prop of a child level type. A parent-level type of a capture prop includes, but is not limited to, at least one of a common level, an uncommon level, and a rare level. A child level type of the common level includes, but is not limited to, at least one of a common child level, an advanced child level, and an excellent child level. A child level type of the uncommon level includes, but is not limited to, a legend child level and a master child level. A child level of the rare level includes, but is not limited to, a master child level. The foregoing is merely exemplary description, and does not constitute a limitation on the division of capture props.

[0065] In some embodiments, a player activates a selection mode by long-pressing (or continuously touching) the capture prop control. A plurality of candidate options of capture props of different levels is presented on the game interface, and a capture prop of a preset level may be selected from the plurality of available candidate options. For example, the player browses capture

props of different levels in the plurality of available candidate options through a slide operation, views detailed information or effects of the capture props of different levels, and selects a capture prop by sliding toward the capture prop of a preset level.

[0066] In some embodiments, capture props of different levels correspond to different visual effects. The significance degree of the visual effect of the capture prop sequentially increases as the level increases. For example, a significance degree of a visual effect of a capture prop of a rare level is richer than that of a capture prop of an uncommon level, and a significance degree of a visual effect of a capture prop of an uncommon level is higher than that of a capture prop of an ordinary level. For example, a significance degree of a visual effect of a capture prop of an excellent child level is richer than that of a capture prop of an advanced child level, and a significance degree of a visual effect of a capture prop of an advanced child level is richer than that of a capture prop of an ordinary child level. In some embodiments, the visual effect of the capture prop includes, but is not limited to, at least one of a color, a style, a material, an animation effect, and the like.

[0067] In some embodiments, a display form of the capture prop includes but is not limited to at least one of a capture gourd, a capture ball, a capture net, a capture binding tape or rope, and a magic spell. The foregoing is merely exemplary description, and does not constitute a limitation on display forms of capture props. For example, as shown in FIG. 4, levels of capture props are classified based on obtained difficulties, and capture props of different levels have different visual effects. An example in which a display form of a capture prop is astral spirit gourd (a type of capture gourd) is configured for description. Part (a) in FIG. 4 belongs to an ordinary astral spirit gourd of an ordinary level. Part (b) in FIG. 4 belongs to an advanced astral spirit gourd of an ordinary level. Part (c) in FIG. 4 belongs to an excellent astral spirit gourd of an ordinary level. Part (d) in FIG. 4 belongs to a legend astral spirit gourd of an uncommon level. Part (e) in FIG. 4 belongs to a master astral spirit gourd of an uncommon level. Part (f) in FIG. 4 belongs to a master astral spirit gourd of a rare level.

[0068] In some embodiments, capture props of different levels correspond to different capture capabilities. Capture capabilities of the capture props sequentially increase as the level increases. For example, a capture prop of a rare level has a capture capability higher than that of a capture prop of an uncommon level, and a capture capability of a capture prop of an uncommon level is higher than that of a capture prop of a common level. For example, a capture prop of an excellent child level has a capture capability higher than a capture prop of an advanced child level, and a capture capability of a capture prop of an advanced child level is higher than a capture prop of an ordinary child level. In some embodiments, obtaining of a capture prop is related to a level of the

main control virtual character. In some embodiments, obtaining of a capture prop is in a positive correlation with the level of the main control virtual character. As the level of the main control virtual character increases, the level of the capture prop of the main control virtual character increases. That is, a higher level of the main control virtual character indicates an increased possibility that the main control virtual character obtains a capture prop of a higher level.

[0069] In some embodiments, a manner of obtaining a capture prop includes but is not limited to at least one of the following:

[0070] obtaining based on that the main control virtual character completes a specified task; obtaining based on that the main control virtual character collects fragments; obtaining based on that the main control virtual character reaches the preset level and performs unlocking; obtaining based on that the main control virtual character picks up in a virtual world; obtaining based on that a teammate virtual character performs giving; obtaining based on that the main control virtual character purchases in a store; obtaining based on that the main control virtual character interacts with a pet virtual character; obtaining based on that a pet virtual character defeated by the main control virtual character falls; obtaining based on that the main control virtual character obtains a raw material for manufacturing; and obtaining based on that a captured pet virtual character is manufactured. The foregoing is merely an exemplary description of a manner of obtaining a capture prop, and does not constitute a limitation on the capture prop.

[0071] Capture manner II: The main control virtual character releases a capture skill for a pet virtual character.

[0072] In some embodiments, a capture skill control is displayed on the game interface in response to a second capture trigger operation. The second capture trigger operation is an operation that is triggered by a player to fight against a to-be-captured pet virtual character, to capture the pet virtual character by using a capture skill. The capture skill control is a control for releasing a capture skill and controlling the captured pet virtual character by the main control virtual character. The capture skill control is an interface interaction element that is in the game interface and that is configured for triggering a capture operation. In some embodiments, the capture skill control is usually presented in an icon, a button, text, or another form. This is not limited in this application. For example, the capture skill control may be a capture skill button. In some embodiments, in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture skill for the pet virtual character. That is, in response to a trigger operation on the capture skill control, the main control virtual character is controlled to release a capture skill for the pet virtual character. In some embodiments, the capture skill includes, but is not

limited to, at least one of an attack capture skill, a feeding capture skill, a curing capture skill, a hypnotic capture skill, an interaction capture skill, and an attribute counteraction capture skill. In a possible implementation, the capture skill is presented as a capture spell-casting effect. In some embodiments, the capture spell-casting effect includes, but is not limited to, at least one of an attack spell-casting effect, a feeding spell-casting effect, a curing spell-casting effect, a hypnotic spell-casting effect, an interaction spell-casting effect, and an attribute counteraction spell-casting effect. In some embodiments, different types of capture spell-casting effects have different display forms. In some embodiments, a display form of the capture spell-casting effect includes at least one of a light effect, a sound effect, an animation effect, an energy fluctuation, a particle effect, a magic circle, and a light strip constraint, but is not limited thereto. This is not limited in this application.

[0073] In some embodiments, in response to a selection operation on the attack capture skill, an attack type spell casting effect released by the main control virtual character on the pet virtual character is displayed. The attack capture skill is a skill configured for damaging or applying a negative effect to a pet virtual character. The attack spell-casting effect is configured for damaging or applying a negative state to a pet virtual character, and is applicable to a pet virtual character that needs to be captured or controlled through force. In some embodiments, in response to a selection operation on the feeding capture skill, a feeding spell-casting effect released by the main control virtual character on the pet virtual character is displayed. The feeding capture skill is a skill configured for providing a gain effect for the pet virtual character, and the feeding spell-casting effect is configured for providing food or other resources for the pet virtual character, to solve its hunger problem or improve its friendliness. In some embodiments, in response to a selection operation on the curing capture skill, a curing type spell casting effect released by the main control virtual character on the pet virtual character is displayed. The curing capture skill is a skill configured for curing damage on a pet virtual character or eliminating a negative state. The curing spell-casting effect is configured for restoring health of the pet virtual character or relieving an injured state of the pet virtual character, and is applicable to a situation in which the pet virtual character is injured or in a poor state. In some embodiments, in response to a selection operation on the hypnotic capture skill, a hypnotic spell-casting effect released by the main control virtual character on the pet virtual character is displayed. The hypnotic capture skill is a skill configured for enabling a pet virtual character to fall into a sleeping state or reducing alertness of the pet virtual character. The hypnotic-type spell-casting effect is configured for enabling a pet virtual character to enter a sleeping state, reducing resistance and mobility of the pet virtual character, so that the pet virtual character is more easily captured. In some embodiments, in response to a selection operation on the interaction capture skill, an interaction spell-casting effect released by the main control

virtual character on the pet virtual character is displayed. The interaction capture skill is a skill configured for promoting interaction with or establishing a relationship with a target. The interaction-type spell-casting effect is configured for directly interacting with a pet, increasing intimacy or friendliness of a pet virtual character, and increasing acceptance of the pet virtual character for capture, such as stroking, communication, or training. In some embodiments, in response to a selection operation on the attribute counteraction capture skill, an attribute counteraction spell-casting effect released by the main control virtual character on the pet virtual character is displayed. The attribute counteraction capture skill is a skill configured for counteracting a specific attribute or reducing an attribute advantage of a pet virtual character. The attribute counteraction spell-casting effect uses an attribute counteraction relationship of the pet virtual character to cast a spell-casting effect having an attribute counteraction function, thereby enhancing a capture success possibility.

[0074] In some embodiments, the main control virtual character releases a capture skill to the pet virtual character according to the state information of the pet virtual character. In some embodiments, the state information of the pet virtual character includes at least one of a personality, a real-time state, and an attribute of the pet virtual character. For example, the state information includes the personality of the pet virtual character. If the personality of the pet virtual character is a fighting personality, the capture skill released by the main control virtual character on the pet virtual character is an attack capture skill. If the personality of the pet virtual character is a docile personality, the capture skill released by the main control virtual character on the pet virtual character is a feeding capture skill. If the personality of the pet virtual character is a timid personality, the capture skill released by the main control virtual character on the pet virtual character is a hypnotic capture skill.

[0075] For example, the state information includes the real-time state of the pet virtual character. If the state of the pet virtual character is a hungry state, the capture skill released by the main control virtual character on the pet virtual character is a feeding capture skill. If the state of the pet virtual character is a vigilant state, the capture skill released by the main control virtual character on the pet virtual character is a hypnotic capture skill. If the state of the pet virtual character is an injured state, the capture skill released by the main control virtual character on the pet virtual character is a curing capture skill.

[0076] For example, the state information includes the attribute of the pet virtual character. If the attribute of the pet virtual character is a fire attribute, the capture skill released by the main control virtual character on the pet virtual character is a water counteraction capture skill. If the

attribute of the pet virtual character is a water attribute, the capture skill released by the main control virtual character on the pet virtual character is an ice counteraction capture skill. If the attribute of the pet virtual character is a dark attribute, the capture skill released by the main control virtual character on the pet virtual character is a light counteraction capture skill. The foregoing is merely an exemplary description of a capture skill, and does not constitute a limitation on the capture skill.

[0077] **Operation 230**: Display a challenge capture effect for the pet virtual character, the challenge capture effect including a struggle capture effect or a struggle escape effect. The challenge capture effect is configured for representing a capture effect generated due to a struggle behavior of the pet virtual character in a capture process. In some embodiments, the challenge capture effect includes a struggle capture effect or a struggle escape effect. In some embodiments, in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture prop for the pet virtual character; or in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture skill for the pet virtual character, to capture the pet virtual character. In some embodiments, after the main control virtual character releases the capture prop or releases the capture skill, a challenge capture effect for the pet virtual character is displayed. In some embodiments, the challenge capture effect includes a struggle capture effect or a struggle escape effect. For example, the challenge capture effect is configured for indicating a capture result of the main control virtual character for the pet virtual character. For example, the struggle capture effect is configured for representing that the pet virtual character is successfully captured by the main control virtual character after the pet virtual character struggles. The struggle escape effect is configured for representing that the pet virtual character escapes from capture after struggling, and is not successfully captured by the main control virtual character.

[0078] In some embodiments, different challenge capture effects are displayed according to different capture conditions. a challenge capture effect for the pet virtual character is displayed as a struggle capture effect or a struggle escape effect when a preset capture condition is satisfied.

[0079] By displaying a challenge capture effect on the pet virtual character, a struggle performance of the pet virtual character is associated with a final capture result (such as capture or escape). The struggle of the pet virtual character not only visually presents feedback to a capture operation, but also actually indicates a capture success rate. For example, a more fiercely struggling pet virtual character indicates a higher escape possibility, and an association is established between a virtual behavior of the pet virtual character and a capture result.

[0080] To sum up, in the method provided in this embodiment of this application, after the main control virtual character releases a capture prop or a capture skill, a challenge capture effect of the pet virtual character in the virtual world is displayed. The challenge capture effect includes a struggle capture effect or a struggle escape effect. By displaying a virtual struggle behavior of the pet virtual character with the capture prop or on the capture skill, it is indicated that the pet virtual character is successfully captured or struggles and escapes after the pet virtual character struggles, thereby visually presenting uncertainty and a challenge of capturing the pet virtual character. A struggle behavior in the real world is simulated, and more real capture experience is provided for the player.

[0081] **Different challenge capture effects are displayed according to different capture conditions**

[0082] In some embodiments, different challenge capture effects are displayed according to different capture conditions. a challenge capture effect for the pet virtual character is displayed as a struggle capture effect or a struggle escape effect when a preset capture condition is satisfied.

[0083] FIG. 5 is a flowchart of a pet virtual character capture method according to an exemplary embodiment of this application. The foregoing operation 230 may be replaced with operation 231 or operation 232.

[0084] **Operation 231**: Display a challenge capture effect for the pet virtual character as a struggle capture effect when a first capture condition is satisfied, the struggle capture effect including that the pet virtual character is successfully captured after the pet virtual character struggles in a first form.

[0085] The struggle capture effect is configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles.

[0086] In some embodiments, the first capture condition is a preset condition or rule that determines whether the pet virtual character triggers a struggle capture effect in a capture process. In some embodiments, the first capture condition is a fixed condition, or the first capture condition is a dynamic condition. The fixed condition refers to a specific and fixed condition that is set in a capture process, and does not change with a game state or another factor. In some embodiments, the fixed condition is the same condition applicable to all pet virtual characters. The dynamic condition refers to a condition that dynamically changes according to a game state or another factor in a capture process. In some embodiments, the dynamic condition dynamically changes for different pet virtual characters. For example, the first capture condition is a fixed condition. When the main

control virtual character captures the pet virtual character by using the capture prop, the first capture condition is the count of struggle of the pet virtual character in the capture prop. For example, if the count of struggle of the pet virtual character in the capture prop is less than three, the first capture condition is satisfied. When different pet virtual characters are captured, if the count of struggle of the different pet virtual characters in the capture prop is less than three, it may be considered that the first capture condition is satisfied, and a challenge capture effect for the pet virtual characters is displayed as a struggle capture effect.

[0087] For example, the first capture condition is a dynamic condition. In some embodiments, the first capture condition is related to at least one of the following factors: a level of a capture prop released by the main control virtual character; a type of a capture prop released by the main control virtual character; a level of the pet virtual character that is being captured; a state of the pet virtual character that is being captured; interaction between the main control virtual character and the pet virtual character before capture; an attribute counteraction relationship between the main control virtual character and the pet virtual character; and the count of capturing the pet virtual character by the main control virtual character. Reference factors (a capture prop level, a capture prop type, an attribute counteraction relationship, the count of capture, and the like) corresponding to the first capture condition are dynamic conditions, and may change with a player operation and a game progress. For example, a level of a capture prop may increase as a game progresses, a type of the capture prop and an attribute counteraction relationship also affect a capture effect, and an increase in the count of capturing may change a capture success rate.

[0088] When the first capture condition is a dynamic condition, the first capture condition includes but is not limited to at least one of the following: a level of the capture prop is higher than a first level; a level of the capture prop is higher than a level of the pet virtual character; and a level of the main control virtual character is higher than a level of the pet virtual character. For example, when a state of the pet virtual character that is being captured is an injured state, it is considered that the first capture condition is satisfied, and a challenge capture effect for the pet virtual character is displayed as a struggle capture effect.

[0089] In some embodiments, when the first capture condition is satisfied, it is displayed that the pet virtual character is successfully captured after the pet virtual character struggles in a first form. The first form refers to a behavior or state presented by the pet virtual character being captured in the first capture condition. In some embodiments, the first form includes, but is not limited to, at least one of the count of struggle, motion amplitude, struggle duration, a struggle sound effect, and an animation effect. The count of struggle refers to the count of struggle motions

presented by the pet virtual character in a process of being captured. For example, a struggle performance motion of a virtual body of the pet virtual character in a process of being captured is counted as a struggle motion. The motion amplitude refers to a motion range or strength of the pet virtual character that is struggling. The struggle duration refers to a length of time for which the pet virtual character has struggled in a process of being captured. The struggle sound effect is a sound effect generated of the pet virtual character that is struggling, and is configured for enhancing a feeling of realism. The animation effect is a visual animation effect presented in a capture process. In some embodiments, the first form of struggle is configured for simulating a natural reaction of the pet virtual character that is being captured. Different pet virtual characters have different struggle performance motions in the first form. For example, the struggle performance motion of the pet virtual character in the first form includes, but is not limited to, at least one of jumping, twisting, flipping, rocking, and dancing. For example, a struggle performance motion of a worm pet virtual character of in the first form is jumping, to jump back and forth. For another example, a struggle performance motion of a plume pet virtual character in the first form is flipping up and down. The foregoing is merely exemplary description, and does not constitute a limitation on struggle performance motions.

[0090] In some embodiments, the virtual environment is changed when the pet virtual character is struggling in the first form. In some embodiments, in response to that the pet virtual character is struggling in the first form, the virtual environment of the pet virtual character is changed. In some embodiments, at least one of a state, a position, terrain, and weather of an object in the virtual environment is changed. For example, the pet virtual character struggling in the first form may crash into an object in the virtual environment, for example, the pet virtual character may crash into a tree in the virtual environment in a process of struggling in the first form. Alternatively, a position of an object is changed in struggling in the first form, for example, a pet virtual character crashes into a stone in the virtual environment in a process of struggling in the first form, and the stone is crashed into a position B from a position A. Alternatively, surrounding terrain is changed in struggling in the first form, for example, in a process in which a pet virtual character struggles in the first form, a surrounding plot starts to become unstable, and the plot has cracks and gradually becomes muddy. Alternatively, the weather in the virtual environment may be changed in struggle in the first form. For example, in a process in which a pet virtual character struggles in the first form, a thunderstorm is caused, and the sky changes from sunny to densely cloudy. The foregoing is merely exemplary description, and does not constitute a limitation on changing the virtual environment.

[0091] **Operation 232**: Display a challenge capture effect for the pet virtual character as a struggle escape effect when a second capture condition is satisfied, the struggle capture effect including that the pet virtual character escapes from capture after the pet virtual character struggles in a second form.

[0092] The struggle escape effect is configured for representing that the pet virtual character escapes from capture after the pet virtual character struggles.

[0093] In some embodiments, the second capture condition is a preset condition or rule that determines whether the pet virtual character triggers a struggle escape effect in a capture process. In some embodiments, the second capture condition is a fixed condition, or the second capture condition is a dynamic condition. The fixed condition refers to a specific and fixed condition that is set in a capture process, and does not change with a game state or another factor. In some embodiments, the fixed condition is the same condition applicable to all pet virtual characters. The dynamic condition refers to a condition that dynamically changes according to a game state or another factor in a capture process. In some embodiments, the dynamic condition dynamically changes for different pet virtual characters. For example, the second capture condition is a fixed condition. When the main control virtual character captures the pet virtual character by using the capture prop, the second capture condition is the count of struggle of the pet virtual character in the capture prop. For example, if the count of struggle of the pet virtual character in the capture prop is greater than three, the second capture condition is satisfied. When different pet virtual characters are captured, if the count of struggle of the different pet virtual characters in the capture prop is greater than three, it may be considered that the second capture condition is satisfied, and a challenge capture effect for the pet virtual characters is displayed as a struggle escape effect.

[0094] For example, the second capture condition is a dynamic condition. In some embodiments, the second capture condition is related to at least one of the following factors: a level of a capture prop released by the main control virtual character; a type of a capture prop released by the main control virtual character; a level of the pet virtual character that is being captured; a state of the pet virtual character that is being captured; interaction between the main control virtual character and the pet virtual character before capture; an attribute counteraction relationship between the main control virtual character and the pet virtual character; and the count of capturing the pet virtual character by the main control virtual character. Reference factors (a capture prop level, a capture prop type, an attribute counteraction relationship, the count of capture, and the like) corresponding to the second capture condition are dynamic conditions, and may change with a player operation and a game progress. For example, a level of a capture prop may increase as a

game progresses, a type of the capture prop and an attribute counteraction relationship also affect a capture effect, and an increase in the count of capturing may change a capture success rate.

[0095] When the second capture condition is a dynamic condition, the second capture condition includes but is not limited to at least one of the following: a level of the capture prop is lower than a first level; a level of the capture prop is lower than a level of the pet virtual character; and a level of the main control virtual character is lower than a level of the pet virtual character. For example, when a level of a pet virtual character that is being captured is higher than a level of the main control virtual character, it is considered that the second capture condition is satisfied, and a challenge capture effect for the pet virtual character is displayed as a struggle escape effect.

[0096] In some embodiments, when the second capture condition is satisfied, it is displayed that the pet virtual character escapes from capture after the pet virtual character struggles in a second form. The second form refers to another behavior or state presented by the pet virtual character being captured in the second capture condition. In some embodiments, the second form includes, but is not limited to, at least one of the count of struggle, motion amplitude, struggle duration, a struggle sound effect, and an animation effect. The count of struggle refers to the count of struggle motions presented by the pet virtual character in a process of being captured. The motion amplitude refers to a motion range or strength of the pet virtual character that is struggling. The struggle duration refers to a length of time for which the pet virtual character has struggled in a process of being captured. The struggle sound effect is a sound effect generated of the pet virtual character that is struggling, and is configured for enhancing a feeling of realism. The animation effect is a visual animation effect presented in a capture process.

[0097] In some embodiments, in the second form of struggle, the pet virtual character also struggles, but a struggle performance motion in this case may be different from that in the first form, to distinguish from that in the first form of struggle. In some embodiments, a difference between a struggle performance motion of the pet virtual character in the second form and a struggle performance motion of the pet virtual character in the first form includes at least one of the following: the count of struggle of the pet virtual character in the second form is higher than the count of struggle of the pet virtual character in the first form; a motion amplitude of the pet virtual character in the second form is stronger than a motion amplitude of the pet virtual character in the first form; a struggle duration of the pet virtual character in the second form is greater than a struggle duration of the pet virtual character in the first form; a struggle sound effect of the pet virtual character in the second form is higher than a struggle sound effect of the pet virtual character

in the first form; and an animation effect of the pet virtual character in the second form is richer than an animation effect of the pet virtual character in the first form.

[0098] For example, the count of struggle of the pet virtual character in the second form is higher than the count of struggle of the pet virtual character in the first form. The count of struggle refers to the count of struggle motions presented by the pet virtual character in a process of being captured. For example, the count of struggle of the pet virtual character in the first form is three, and the count of struggle of the pet virtual character in the second form is four. For example, a motion amplitude of the pet virtual character in the second form is stronger than a motion amplitude of the pet virtual character in the first form. The motion amplitude refers to a motion range or strength of the pet virtual character that is struggling. For example, a struggle motion of the pet virtual character in the first form may be temperate and has a small motion amplitude; and a struggle motion of the pet virtual character in the second form may be more violent and has a large motion amplitude.

[0099] For example, a struggle duration of the pet virtual character in the second form is greater than a struggle duration of the pet virtual character in the first form. The struggle duration refers to a length of time for which the pet virtual character has struggled in a process of being captured. For example, the struggle duration of the pet virtual character in the first form is short, for example, 3 seconds, indicating that the resistance of the pet virtual character is not strong; and the struggle duration of the pet virtual character in the second form is long, for example, 5 seconds, indicating that the pet virtual character strives to escape.

[00100] For example, a struggle sound effect of the pet virtual character in the second form is higher than a struggle sound effect of the pet virtual character in the first form. The struggle sound effect is a sound effect generated of the pet virtual character that is struggling, and is configured for enhancing a feeling of realism. For example, a struggle sound effect of the pet virtual character in the first form is gentle, such as a low grunt sound or a slight struggle sound; and a struggle sound effect of the pet virtual character in the second form is resounding, for example, a scream sound or a strong striking sound.

[00101] For example, an animation effect of the pet virtual character in the second form is richer than an animation effect of the pet virtual character in the first form. The animation effect is a visual animation effect presented in a capture process. For example, an animation effect of the pet virtual character in the first form is simple, and involves few motions, such as slight shaking or swinging; and an animation effect of the pet virtual character in the second form is complex, and involves more motions such as jumping, rolling over, or quick moving.

[00102] In some embodiments, a struggle performance motion of the pet virtual character affects a physical energy value of the main control virtual character. For example, a struggle performance motion of the pet virtual character consumes a physical energy value of the main control virtual character. In some embodiments, the physical energy value consumption of the main control virtual character may be in a positive correlation with strength of a struggle performance motion of the pet virtual character. For example, a larger count of struggle of the pet virtual character indicates greater physical energy value consumption of the main control virtual character. For another example, a larger motion amplitude of the pet virtual character indicates faster physical energy value consumption of the main control virtual character. For another example, a longer struggle duration of the pet virtual character indicates more physical energy value consumption of the main control virtual character. For another example, a larger struggle sound effect of the pet virtual character indicates faster physical energy value consumption of the main control virtual character. In some embodiments, the physical energy value of the main control virtual character is further related to a state of the main control virtual character before the main control virtual character captures the pet virtual character. For example, if the main control virtual character is in a de-buffing state (for example, just going through a battle) before the pet virtual character is captured, the physical energy value of the main control virtual character decreases. In some embodiments, the physical energy value of the main control virtual character is associated with performance of the capture prop. For example, when the physical energy value of the main control virtual character is at a high level, the performance of the capture prop is good, and in this case, the capture success rate is high. As the physical energy value of the main control virtual character is consumed, the performance of the capture prop decreases, and the capture success rate is low. In some embodiments, in response to that the physical energy value of the main control virtual character decreases to a physical energy threshold, it is displayed that the main control virtual character loses a capability of controlling the capture prop. The physical energy threshold is a minimum physical energy threshold at which the main control virtual character can control use of the capture prop. For example, when the physical energy value of the main control virtual character is consumed until 20% of a full physical energy value, it is displayed that the main control virtual character loses a capability of controlling the capture prop, for example, the capture prop is broken or invalidated. In this case, a possibility that the pet virtual character struggles and escapes increases accordingly. In some embodiments, a struggle performance motion of the pet virtual character affects the performance of the capture prop. The performance of the capture prop includes, but is not limited to, at least one of the endurance corresponding to the capture prop, the magnitude of a capture force corresponding to the capture prop, and the completeness corresponding to the capture

prop. In some embodiments, an example in which a struggle performance motion of the pet virtual character reduces the endurance of the capture prop is configured for description. In some embodiments, the strength of the struggle performance motion of the pet virtual character is in a negative correlation with the endurance of the capture prop. For example, an increased count of struggle of the pet virtual character indicates decreased endurance of the capture prop. For another example, an increased motion amplitude of the pet virtual character indicates decreased endurance of the capture prop. For another example, an increased struggle duration of the pet virtual character indicates decreased endurance of the capture prop. In some embodiments, capture props of different levels correspond to different endurance. The endurance corresponding to a capture prop of a first level is higher than the endurance corresponding to a capture prop of a second level, and the grade of the first level is higher than the grade of the second level. In some embodiments, when struggle performance motions of pet virtual characters are the same (or strengths of the struggle performance motions are the same), reduction amounts of endurance corresponding to different levels of capture props are different. In response to that the endurance corresponding to the capture prop decreases to an endurance threshold, it is displayed that the capture prop loses a capability of capturing the pet virtual character. The endurance threshold is a minimum endurance threshold at which the capture prop can be normally used. For example, the endurance corresponding to the capture prop of the first level is 10, the endurance corresponding to the capture prop of the second level is 5, and the endurance threshold is 50%, that is, when the endurance corresponding to the capture prop decreases to 50% of a full endurance value (initial endurance), the capture prop loses a capability of capturing the pet virtual character. For example, the count of struggle of the pet virtual character in the capture prop of the first level is 3, and the reduction amount of endurance corresponding to the capture prop of the first level is 2, that is, the endurance corresponding to the capture prop of the first level is reduced from 10 to 8. For example, the count of struggle of the pet virtual character in the capture prop of the second level is 3, and the reduction amount of endurance corresponding to the capture prop of the second level is 3, that is, the endurance corresponding to the capture prop of the second level is reduced from 5 to 2. In this case, the endurance corresponding to the capture prop of the second level decreases to the endurance threshold. That is, when the count of struggle of the pet virtual character in the capture prop of the second level is 3, the capture prop of the second level loses a capability of capturing the pet virtual character. In some embodiments, the struggle performance motion of the pet virtual character reduces magnitude of a capture force of the capture prop, or the struggle behavior of the pet virtual character consumes completeness corresponding to the capture prop, whose implementation principle is similar to that the struggle performance motion of the pet virtual character reduces the endurance corresponding to the capture prop. Details are not

described herein again. To sum up, in the method provided in this embodiment of this application, the first capture condition and the second capture condition are set, and when different capture conditions are satisfied, different challenge capture effects may be displayed. When the first capture condition is satisfied, the pet virtual character is successfully captured, and the player sees a struggle capture effect. When the first capture condition is satisfied, the pet virtual character escapes, and the player sees a struggle escape effect. This design enhances interaction and challenge of a game, so that the player experiences a richer game process.

[00103] **The capture condition is the count of struggle of the pet virtual character in the capture prop.**

[00104] In some embodiments, the capture condition is a fixed condition. The fixed condition refers to a specific and fixed condition that is set in a capture process, and does not change with a game state or another factor. In some embodiments, the fixed condition is the same condition applicable to all pet virtual characters. For example, the fixed condition is the count of struggle of the pet virtual character in the capture prop.

[00105] FIG. 6 is a flowchart of a pet virtual character capture method according to an exemplary embodiment of this application. Operation 231 may be replaced with operation 231a, and operation 232 may be replaced with operation 232a.

[00106] **Operation 231a**: Display the challenge capture effect for the pet virtual character as the struggle capture effect in response to that a count of struggle of the pet virtual character in the capture prop is less than or equal to a count threshold.

[00107] The count of struggle refers to the count of struggle motions presented by the pet virtual character in a process of being captured. The count threshold is configured for indicating a maximum count of struggle of the pet virtual character that can be tamed in the capture prop. In some embodiments, the challenge capture effect for the pet virtual character is displayed as the struggle capture effect in response to that a count of struggle of the pet virtual character in the capture prop is less than or equal to a count threshold, that is, the count of struggle motions presented by the pet virtual character in the capture prop is less than or equal to the maximum count of struggle of the pet virtual character that can be tamed. For example, as shown in part (a) of FIG. 7, a main control virtual character 301 is displayed in a virtual world picture 300, and in response to a capture operation on a pet virtual character 302, the main control virtual character 301 is controlled to release a capture prop 303 for the pet virtual character 302. As shown in part (b) of FIG. 7, an example in which the count threshold is 3 is configured for description. In response to that a count of struggle of the pet virtual character 302 in the capture prop 303 is less than or equal

to a count threshold, for example, the count of struggle of the pet virtual character 302 in the capture prop 303 is 2, a challenge capture effect for the pet virtual character 302 is displayed as a struggle capture effect, that is, it is displayed that the pet virtual character 302 is successfully captured after the pet virtual character struggles.

[00108] In a possible implementation, in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture prop for the pet virtual character. In some embodiments, after the main control virtual character releases the capture prop, the pet virtual character enters the capture prop, and a capture probability is displayed at a peripheral side position of the capture prop. The capture probability is configured for indicating a probability that the main control virtual character successfully captures the pet virtual character. In some embodiments, the challenge capture effect for the pet virtual character is displayed when a capture probability of the pet virtual character falls within a target probability interval. In some embodiments, the challenge capture effect includes a struggle capture effect and a struggle escape effect. When the capture probability of the pet virtual character is within the target probability interval, the challenge capture effect is displayed, to enable the user to feel feedback of a capture process, thereby enhancing a dynamic response capability of the game. In some embodiments, n random numbers are generated within the target probability interval according to a preset rule, n being a positive integer. The preset rule refers to a series of standards or conditions that are set before the random numbers are generated, and defines a constraint condition that is to be complied with in a random number generation process. In some embodiments, the preset rule includes at least one of a range of random numbers, distribution of random numbers, and a quantity of random numbers. For example, the preset rule may specify that the random numbers are to be generated in a particular value range. In some embodiments, the count of struggle of the pet virtual character in the capture prop is determined based on a magnitude relationship between the n random numbers and the capture probability. In some embodiments, a count in the n random numbers that is greater than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. Alternatively, a count in the n random numbers that is less than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. The count of struggle is determined by combining the random numbers and the capture probability, so that the capture process has fairness and variability. The use of random numbers increases uncertainty, ensures uniqueness of a result of each capture attempt, and enriches strategies for capturing a pet virtual character. In some embodiments, a count in the n random numbers that is greater than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. In some embodiments, the capture effect for the pet virtual character is determined as the

struggle capture effect if a count of struggle of the pet virtual character in the capture prop is less than or equal to a count threshold. For example, the server randomly generates n random numbers (or referred to as random probabilities). A value range of the n random numbers falls within the target probability interval. The capture probability is compared with the n random numbers successively. A count in the n random numbers that is greater than the capture probability is determined as a count of struggle of the pet virtual character in the capture prop. If the count of struggle is less than or equal to a count threshold, it is determined that a capture effect of the pet virtual character is a struggle capture effect. For example, the target probability interval is (60%, 40%)], and the capture probability is 50%. Using an example in which n is 4, the randomly generated 4 random numbers are respectively 58%, 55%, 48%, and 43%. The capture probability is compared with the 4 random numbers successively. The count at which the 4 random numbers are greater than the capture probability is 2, that is, the count at which the pet virtual character struggles in the capture prop is 2. Using an example in which the count threshold is 3, the count of struggle is less than the count threshold (2 is less than 3), a challenge capture effect for the pet virtual character displayed on the terminal is presented as a struggle capture effect. In some embodiments, a capture progress bar is displayed at a peripheral side position of the capture prop in response to that the pet virtual character starts to struggle for the first time in the capture prop. In some embodiments, the capture progress bar increases to a target capture progress (for example, the target capture progress is 100% progress) at a preset rate, and before the capture progress bar reaches the target capture progress, the count of struggle of the pet virtual character in the capture prop is determined. The challenge capture effect for the pet virtual character is displayed as the struggle capture effect if the count of struggle is less than or equal to a count threshold.

[00109] For example, as shown in FIG. 8, in a virtual world picture 400, a main control virtual character 401 is displayed. After the main control virtual character 401 hits a pet virtual character by using a capture prop 402, the pet virtual character enters the capture prop 402. If the pet virtual character starts to struggle for the first time in the capture prop 402, a capture progress bar 403 is displayed at a peripheral side position of the capture prop 402. For example, the capture progress bar 403 at this time is displayed as 51%. Before the capture progress bar 403 reaches 100% progress, the count of struggle of the pet virtual character in the capture prop is determined. The challenge capture effect is displayed as the struggle capture effect if the count of struggle is less than or equal to a count threshold. That is, the pet virtual character is successfully captured.

[00110] In this embodiment of this application, the count of struggle of the virtual character in the capture prop can be fairly determined by comparing the random number with the capture

probability. Randomness is introduced to this method, so that the game is more dynamic and interesting.

[00111] **Operation 232a**: Display the challenge capture effect for the pet virtual character as the struggle escape effect in response to that a count of struggle of the pet virtual character in the capture prop is greater than a count threshold.

[00112] The count of struggle refers to the count of struggle motions presented by the pet virtual character in a process of being captured. The count threshold is configured for indicating a maximum count of struggle of the pet virtual character that can be tamed in the capture prop. In some embodiments, the challenge capture effect for the pet virtual character is displayed as the struggle escape effect in response to that a count of struggle of the pet virtual character in the capture prop is greater than a count threshold, that is, the count of struggle motions presented by the pet virtual character in the capture prop is greater than the maximum count of struggle of the pet virtual character that can be tamed. For example, as shown in part (a) of FIG. 9, a main control virtual character 501 is displayed in a virtual world picture 500, and in response to a capture operation on a pet virtual character 502, the main control virtual character 501 is controlled to release a capture prop 503 for the pet virtual character 502. As shown in part (b) of FIG. 9, an example in which the count threshold is 3 is configured for description. In response to that a count of struggle of the pet virtual character 502 in the capture prop 503 is greater than a count threshold, for example, the count of struggle of the pet virtual character 502 in the capture prop 503 is 4, a challenge capture effect for the pet virtual character 502 is displayed as a struggle capture effect, that is, it is displayed that the pet virtual character 502 is successfully captured after the pet virtual character struggles.

[00113] In a possible implementation, in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture prop for the pet virtual character. In some embodiments, after the main control virtual character releases the capture prop, the pet virtual character enters the capture prop, and a capture probability is displayed at a peripheral side position of the capture prop. The capture probability is configured for indicating a probability that the main control virtual character successfully captures the pet virtual character. In some embodiments, the challenge capture effect for the pet virtual character is displayed when a capture probability of the pet virtual character falls within a target probability interval. In some embodiments, the challenge capture effect includes a struggle capture effect and a struggle escape effect. In some embodiments, n random numbers are generated within the target probability interval according to a preset rule, n being a positive integer. the count of struggle of the pet virtual

character in the capture prop is determined based on a magnitude relationship between the n random numbers and the capture probability. In some embodiments, a count in the n random numbers that is greater than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. Alternatively, a count in the n random numbers that is less than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. In some embodiments, a count in the n random numbers that is greater than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. In some embodiments, the capture effect for the pet virtual character is determined as the struggle escape effect if a count of struggle of the pet virtual character in the capture prop is greater than a count threshold. For example, n random numbers (or referred to as random probabilities) are randomly generated. A value range of the n random numbers falls within the target probability interval. The capture probability is compared with the n random numbers successively. A count in the n random numbers that is greater than the capture probability is determined as a count of struggle of the pet virtual character in the capture prop. If the count of struggle is greater than a count threshold, it is determined that a capture effect of the pet virtual character is a struggle escape effect. For example, the target probability interval is (60%, 40%)], and the capture probability is 50%. Using an example in which n is 4, the randomly generated 4 random numbers are respectively 58%, 55%, 53%, and 51%. The capture probability is compared with the 4 random numbers successively. The count at which the 4 random numbers are greater than the capture probability is 4, that is, the count at which the pet virtual character struggles in the capture prop is 4. Using an example in which the count threshold is 3, the count of struggle is greater than the count threshold (4 is greater than 3), that is, a challenge capture effect of the pet virtual character is presented as a struggle escape effect.

[00114] In some embodiments, n random numbers are generated based on a random number algorithm. The random number algorithm usually needs random number seeds as input, to initialize a state of the random number algorithm, and values of the random number seeds determine a generated random number sequence. In a possible implementation, n types of attribute values of the pet virtual character are used as random number seeds, and the random number algorithm generates n random numbers based on the random number seeds. The n types of attribute values include, but are not limited to, at least one of a life value, a mood value, a magic value, an attack value, and a defense value of the pet virtual character. Each type of attribute value may generate one random number, and the n types of attribute values may generate n random numbers. The n types of attribute values are used as random number seeds to generate n different random numbers, and the generated n random numbers may be configured for determining a capture success rate of the pet virtual character. In some embodiments, current states of the n types of attribute values directly

affect the random number sequence. For example, if an attack value of the pet virtual character is very high, it is more likely that a generated random number is greater than the capture probability. For another example, if a life value of the pet virtual character is low, it is more likely that a generated random number is less than the capture probability. In some embodiments, a capture progress bar is displayed at a peripheral side position of the capture prop in response to that the pet virtual character starts to struggle for the first time in the capture prop. In some embodiments, the capture progress bar increases to a target capture progress (for example, the target capture progress is 100% progress) at a preset rate, and before the capture progress bar reaches the target capture progress, the count of struggle of the pet virtual character in the capture prop is determined. The challenge capture effect for the pet virtual character is displayed as the struggle escape effect if the count of struggle is greater than a count threshold. To sum up, according to the method provided in this embodiment of this application, a count threshold is set, and a displayed challenge capture effect is determined according to the count of struggle of a pet virtual character, thereby improving interaction and a feeling of challenge of a game. In some embodiments, two different challenge capture effects (a struggle capture effect and a struggle escape effect) are presented according to different counts of struggle of a pet virtual character in capture props, so that a player can intuitively see a capture progress, and can experience a more real and dynamic capture process, thereby enhancing immersion and fun of a game.

[00115] **Different capture effects are displayed based on probability intervals to which capture probabilities belongs**

[00116] In some embodiments, the challenge capture effect for the pet virtual character is displayed when a capture probability of the pet virtual character falls within a target probability interval. In some embodiments, a successful capture effect or failed capture effect for the pet virtual character is further displayed when a capture probability of the pet virtual character falls within a non-target probability interval. The successful capture effect is configured for representing that the pet virtual character does not struggle and is successfully captured, and the failed capture effect is configured for representing that the pet virtual character does not struggle and fails to be captured.

[00117] In some embodiments, when the capture probability of the pet virtual character falls within a first probability interval, it is determined that a capture type for the pet virtual character is a successful capture type and a capture effect for the pet virtual character is displayed as a successful capture effect, the successful capture effect being configured for representing that the pet virtual character does not struggle and is successfully captured; when the capture probability of the pet virtual character falls within a second probability interval, it is determined that a capture type for

the pet virtual character is a successful capture type and a capture effect for the pet virtual character is displayed as a struggle capture effect; when the capture probability of the pet virtual character falls within a target probability interval, it is determined that a capture type for the pet virtual character is a challenge capture type and a challenge capture effect (including a struggle capture effect or a struggle escape effect) for the pet virtual character is displayed; and when the capture probability of the pet virtual character falls within a third probability interval, it is determined that a capture type for the pet virtual character is a failed capture type and a capture effect for the pet virtual character is displayed as a failed capture effect, the failed capture effect being configured for representing that the pet virtual character does not struggle and fails to be captured. A minimum value of the first probability interval is greater than a maximum value of the second probability interval, a minimum value of the second probability interval is greater than a maximum value of the target probability interval, and a minimum value of the target probability interval is greater than a maximum value of the third probability interval.

[00118] In some embodiments, in response to a throwing operation on the capture prop, after the capture prop hits the pet virtual character, or after the main control virtual character is controlled to release a capture skill (a capture spell-casting effect) for the pet virtual character, a capture value for the pet virtual character is calculated, and the capture value is configured for indicating a capture probability of capturing the pet virtual character. A probability interval to which the capture probability belongs is determined according to the capture probability, and capture effects for the pet virtual character in different probability intervals are displayed. In some embodiments, a capture probability for the pet virtual character is calculated based on a capture value. When the capture value falls within a first range interval, the capture probability of the pet virtual character falls within the first probability interval. When the capture value falls within a second range interval, the capture probability of the pet virtual character falls within the second probability interval. When the capture value falls within a target range interval, the capture probability of the pet virtual character falls within the target probability interval. When the capture value falls within a third range interval, the capture probability of the pet virtual character falls within the third probability interval. A minimum value of the first range interval is greater than a maximum value of the second range interval, a minimum value of the second range interval is greater than a maximum value of the target range interval, and a minimum value of the target range interval is greater than a maximum value of the third range interval.

[00119] For example, a capture probability for the pet virtual character is calculated based on a capture value. When the capture value falls within the first range interval, the first range interval is a range interval less than the maximum capture value and greater than or equal to the first capture

value, and the first probability interval is a probability interval less than or equal to 1 and greater than or equal to the first capture probability. For example, the first range interval is [max, A]. In this case, the first probability interval is [1, a%]. When the capture value falls within the second range interval, the second range interval is a range interval less than the first capture value and greater than or equal to the second capture value, and the second probability interval is a probability interval less than the first capture probability and greater than or equal to the second capture probability. For example, the second range interval is (A, B]. In this case, the second probability interval is (a%, b%]. When the capture value belongs to the target range interval, the target range interval is a range interval less than the maximum target capture value (the second capture value) and greater than or equal to the minimum target capture value (the third capture value), and the target probability interval is a probability interval less than a maximum target capture probability (the second capture probability) and greater than or equal to a minimum target capture probability (the third capture probability). For example, the target range is (B, C]. In this case, the target probability interval is (b%, c%]. When the capture value falls within the third range interval, the third range interval is a range interval less than the third capture value and greater than or equal to the minimum capture value, and the third probability interval is a probability interval less than the third capture probability and greater than or equal to 0. For example, the third range interval is (c, 0]. In this case, the second probability interval is (c%, 0]. The first capture value is greater than the second capture value, and the second capture value is greater than the third capture value. That is, A is greater than B, and B is greater than C. The first capture probability is greater than the second capture probability, the second capture probability is greater than the third capture probability, the first capture probability is less than or equal to 1, and the third capture probability is greater than or equal to 0, that is, $1 \geq a\% > b\% > c\% \geq 0$.

[00120] In some embodiments, as the capture value decreases, a capture probability of the main control virtual character for the pet virtual character also decreases accordingly. In some embodiments, within a corresponding probability interval, a capture effect for the pet virtual character is also reduced accordingly. For example, when the capture probability falls within the first probability interval (the capture value falls within the first range interval), the capture type for the pet virtual character is a successful capture type, and it is displayed that a successful capture effect is presented as that the capture prop directly succeeds in taming the pet virtual character, and the pet virtual character has no struggle behavior. When the capture probability falls within the second probability interval (the capture value falls within the second range interval), the capture type for the pet virtual character is a successful capture type, and it is displayed that a struggle capture effect is presented as that the capture prop succeeds in taming the pet virtual character, and

the pet virtual character has a struggle behavior in the capture prop. When the capture probability falls within the target probability interval (the capture value belongs to the target range interval), the capture type for the pet virtual character is a challenge capture type, and the challenge capture type includes a struggle capture effect or a struggle escape effect. The struggle capture effect is presented as that the pet virtual character has a struggle behavior in the capture prop, and is successfully captured after the pet virtual character struggles; or the struggle escape effect is presented as that the pet virtual character has a struggle behavior in the capture prop, and escapes from capture after the pet virtual character struggles. When the capture probability falls within the third probability interval (the capture value falls within the third range interval), the capture type for the pet virtual character is a failed capture type, and it is displayed that a failed capture effect is presented as that the capture prop directly ricochets.

[00121] In some embodiments, in response to a throwing operation on the capture prop, after the capture prop hits an illegal unit, a text prompt of uncapturable is displayed on the game interface. The illegal unit refers to a non-pet virtual character, such as a stone, a tree, or a lawn. For example, as shown in FIG. 10, a main control virtual character 601 is displayed in a virtual world picture 600, and the main control virtual character 601 captures the pet virtual character 602 based on the capture prop 603. When a capture value falls within a first range interval, the capture probability falls within the first probability interval, the capture type for the pet virtual character 602 is a successful capture type, and it is displayed that a successful capture effect is presented as that the capture prop 603 directly succeeds in taming the pet virtual character 602. For example, the pet virtual character 602 is directly changed into a starlight and is inhaled by the capture prop 603, and the pet virtual character 602 has no struggle behavior in the capture prop 603.

[00122] To sum up, in the method provided in this embodiment of this application, capture effects for a pet virtual character are displayed at different capture probability intervals, thereby providing a clear capture mechanism. Capture probabilities are divided into different probability intervals, so that corresponding capture effects (a successful capture effect, a struggle capture effect, a struggle escape effect, or a failed capture effect) in different probability intervals can be more clearly presented to a player, and the player can adjust a strategy according to a capture type and a capture effect, to improve strategy and fun of a game.

[00123] **A capture probability for the pet virtual character is calculated based on a capture value**

[00124] In some embodiments, a capture probability for the pet virtual character is calculated based on a capture value. In some embodiments, when the capture probability for the pet virtual

character is calculated, a capture value for the pet virtual character is first obtained, and then the capture probability for the pet virtual character is calculated according to the capture value. In some embodiments, a basic capture value for the pet virtual character is obtained, the basic capture value being configured for indicating a basic capture probability of capturing the pet virtual character, and configured for indicating a capture capability of a capture prop or a capture skill for the pet virtual character. For example, as a virtual level of a capture prop or a capture skill increases, a basic capture probability increases. In some embodiments, in a process in which the main control virtual character captures the pet virtual character, there are further some correction types that affect magnitude of a basic capture value, and further affect magnitude of a basic capture probability. In some embodiments, a correction type is obtained, and the basic capture value is corrected based on content indicated by the correction type to obtain a corrected basic capture probability. The corrected basic capture probability is a capture probability, that is, the capture probability is obtained based on the content indicated by the correction type.

[00125] In some embodiments, the correction type includes at least one of the following:

[00126] state correction, configured for indicating correction of the basic capture value by a state of the pet virtual character; gain correction, configured for indicating correction of the basic capture value by a positive gain effect and/or a negative gain effect; level suppression correction, configured for indicating correction of the basic capture value by a level difference between the main control virtual character and the pet virtual character; continuous capture compensation correction, configured for indicating correction of the basic capture value by a continuous capture operation performed by the main control virtual character; and capture force correction, configured for indicating correction of the basic capture value by a capture capability of the main control virtual character. For example, the correction type includes state correction. The state correction refers to impact of a current state of the pet virtual character on a basic capture value. The state is configured for indicating a state of the pet virtual character that is being captured. In some embodiments, a pet virtual character in a special state (such as sleeping, frozen, or poisoned) affects a capture probability. The pet virtual character in the special state performs correction on a basic capture probability based on the state information, to obtain a capture probability. The capture probability may be greater than or less than the basic capture probability. For example, when the pet virtual character is in a sleeping state, the corrected capture probability is greater than the basic capture value, that is, a capture probability of capturing the pet virtual character in the sleeping state is increased compared with a capture probability of capturing the pet virtual character in an awake non-sleeping state. For another example, when the pet virtual character is in a high health point state, the corrected capture probability is less than the basic capture value. That is, when the pet

virtual character is in the high health point state, a survival state of the pet virtual character is good, and the difficulty of capturing the pet virtual character is increased. For another example, when the pet virtual character is in a frozen state, the corrected capture probability is greater than the basic capture value.

[00127] For example, the correction type includes gain correction. Gain correction refers to impact of a positive gain effect (for example, a buffing state) or a negative gain effect (for example, a de-buffing state) on the basic capture value. The positive gain effect may improve the basic capture value and increase a capture success opportunity. The negative gain effect reduces the basic capture value, making capture more difficult. For example, the gain correction may be determined based on a positive gain relationship or a negative gain relationship between an element attribute of the pet virtual character and an element attribute of a plot in the position of the pet virtual character. For example, when the pet virtual character is a fire pet virtual character, a fighting capability becomes weaker when the pet virtual character moves to a lake side, and a basic capture value for the pet virtual character increases. In this case, the main control virtual character obtains a positive gain effect, that is, a capture probability of capturing the pet virtual character is increased. For another example, when the pet virtual character is a stone pet virtual character, and when the pet virtual character is hidden near a stone, a basic capture value for the pet virtual character is reduced. In this case, the main control virtual character obtains a negative gain effect, that is, a capture probability of capturing the pet virtual character is reduced. For example, the gain correction may also be provided by a gain prop or a gain skill used by the main control virtual character. In an example, a main control virtual character uses a capture buffing skill. Correspondingly, a capture probability of capturing a pet virtual character increases, and the main control virtual character obtains a positive gain effect. In another example, a main control virtual character uses a life value buffing skill. Based on increasing a virtual life value, the foregoing life value buffing skill has an attached effect of reducing a capture probability of a pet virtual character, and the main control virtual character obtains a negative gain effect about capturing the pet virtual character.

[00128] For example, the correction type includes level suppression correction. The level suppression correction refers to impact of a level difference between a main control virtual character and a pet virtual character on a basic capture value. If the level of the main control virtual character is significantly higher than that of the pet virtual character, the capture difficulty is reduced, and the basic capture value is increased, that is, a capture probability for the pet virtual character is increased. On the contrary, if the level of the main control virtual character is lower than that of the pet virtual character, the basic capture value is reduced, that is, a capture probability for the pet virtual character decreases. For example, a level of a main control virtual character is

level 30, a level of a captured pet virtual character is level 20, and a difference between the level of the main control virtual character and the level of the pet virtual character is 10 levels. It may be set that each time the level of the main control virtual character is increased by 1 level, the capture probability for the pet virtual character is increased by 1%.

[00129] For example, the correction type includes continuous capture compensation correction. The continuous capture compensation correction is to adjust a capture operation continuously performed by the main control virtual character, to compensate for impact of continuous capture on a basic capture value. In some embodiments, continuous capture performed by the main control virtual character improves a basic capture value due to an increase in the count of capturing, thereby increasing a capture probability for the pet virtual character. As the count of continuous attempts increases, a player is given positive feedback, thereby increasing a capture success opportunity. In some embodiments, continuous capture performed by the main control virtual character reduces a basic capture value due to an increase in the count of capturing, thereby reducing a capture probability for the pet virtual character, and increasing a challenge of the game. In some embodiments, continuous capture compensation rates may be fixed values, or continuous capture compensation rate may be progressively increasing values. For example, an example in which a main control virtual character performs continuous capture to improve a basic capture value is configured for description. When the main control virtual character captures the pet virtual character for the first time, a capture probability is 50%. If the first time of capture fails, the main control virtual character immediately attempts again. The capture probability for the pet virtual character increases by 5% (a continuous capture compensation rate is 5%) during the second time of capture, that is, a capture probability during the second time of capture is 55%. The capture probability for the pet virtual character increases by 10% (a continuous capture compensation rate is 10%) during the third time of capture, that is, a capture probability for the pet virtual character is 65% during the third time of capture.

[00130] In an example, when a count of continuous capture performed by the main control virtual character is greater than a first value, a basic capture probability is corrected based on continuous capture compensation correction, so that an obtained capture probability is greater than the basic capture probability. A player is given positive feedback, thereby increasing a capture success opportunity. When a count of continuous capture performed by the main control virtual character is greater than a second value, a basic capture probability is corrected based on continuous capture compensation correction, so that an obtained capture probability is less than the basic capture probability. A state in which physical energy is consumed and capture accuracy is reduced is simulated when a main control virtual character performs continuous capture. The first value is less

than the second value, and the first value and the second value are both positive integers. For example, the correction type includes capture force correction. The capture force correction refers to impact of a capture capability of the main control virtual character on a basic capture value. The capture force of the main control virtual character is related to a current level of the main control virtual character, a level of a capture prop used, a capture skill used, and an empirical value of the capture prop. In some embodiments, a stronger capture force of the main control virtual character indicates an increased basic capture value for the pet virtual character, that is, an increased capture probability for the pet virtual character, and a larger capture success opportunity. On the contrary, a weaker capture force of the main control virtual character indicates a reduced basic capture value for the pet virtual character, that is, a reduced capture probability for the pet virtual character, and a smaller capture success opportunity. For example, if a main control virtual character uses an advanced capture prop or improves a capture skill, a capture probability for a pet virtual character is increased. For example, an empirical value of a capture skill or a capture prop used by the main control virtual character is also referred to as proficiency, and increases as at least one of a count of using a capture skill or a capture prop by the main control virtual character and a time length of using the capture skill or the capture prop by the main control virtual character increases. For example, the foregoing proficiency is independent of a level of a capture skill or a capture prop.

[00131] In a possible implementation, various correction types are coefficients configured for adjusting the basic capture value, to reflect impact of different conditions and behaviors on the capture probability. In some embodiments, a capture formula may be expressed as:

Capture value = basic capture value * (1 + capture correction rate) * character capture force * state correction * level suppression correction * (1 + capture admission limit probability) * (3 * maximum life value − 2 * current life value)/maximum life value

[00132] The basic capture value is related to a level of the capture prop. The character capture force is configured for indicating a capability of the main control virtual character to capture the pet virtual character, the capture correction rate is configured for indicating correction of the capture force of the main control virtual character, and when the character capture force is not corrected, the capture correction rate is 0. The state correction is configured for indicating correction of the basic capture value by the state of the pet virtual character. When the pet virtual character has no special state (such as sleeping, poisoned, or frozen), the state correction is 1 by default. If the pet virtual character has a plurality of special states, a maximum value in the plurality of special states is used as a coefficient by default. The level suppression correction is configured for indicating correction of the basic capture value by a level difference between the main control

virtual character and the pet virtual character. A value of the level suppression correction is usually a maximum value max ((1 – level suppression attenuation), 5%) of the level difference, and 5% is a configuration constant. The capture admission limit probability is configured for increasing a success rate at which a player continuously attempts to capture the same pet virtual character. When the player captures a pet virtual character, the server searches for a capture admission limit probability according to the count at which the player previously attempts to capture such type of pet virtual character. If a player attempts to capture a pet virtual character for the first time, an admission limit probability starts from 0, and starts after the player fails for the first time, and a capture admission limit probability gradually increases. Each time an attempt fails, a capture admission limit probability increases, until the capture succeeds. Once the player successfully captures the pet virtual character, the count of attempts accumulated previously is zeroed out, and a capture admission limit probability starts to be calculated again in the next capture. The maximum life value is configured for indicating an upper limit of the life value that the pet virtual character has in an optimal state. The current life value refers to a current actual life value of the pet virtual character. The current life value changes with a game process and activities of the pet virtual character. When the pet virtual character is injured, the current life value decreases, and when the pet virtual character uses a life value restoring skill, the current life value increases.

[00133] In some embodiments, after obtaining the capture value of the pet virtual character, the server calculates a capture probability for the pet virtual character based on the capture value, and returns the calculated capture probability to the terminal for displaying.

[00134] In a possible implementation, a capture probability is expressed as:

$$\text{Capture probability} = \min[3.78/B * (\text{capture value})^{(1/4N)} * 100\%, 99\%]$$

[00135] When 3.78/B * (capture value) ^ (1/4N) is greater than 1, 100% is displayed, B is a denominator constant 60, and N is an exponent constant 0.8. For example, when the capture value is 5, the capture probability is approximately 69.70%; when the capture value is 10, the capture probability is approximately 86.56%; and when the capture value is 20, the capture probability is approximately 99%. 3.78 is a first correction parameter for the denominator constant B, and may be implemented as different values in different examples. 1/4 is a second correction parameter for the exponent constant N, and may be implemented as different values in different examples. In this embodiment of this application, by obtaining the basic capture value and the correction type, the capture probability may be controlled and adjusted, so that the game system can dynamically adjust a capture difficulty according to different conditions and correction types, thereby improving experience of the player.

[00136] In this embodiment of this application, the basic capture value is adjusted by using a plurality of correction types, and the server can calculate a final capture probability by considering a plurality of factors (for example, a state of the pet virtual character, a gain effect, a level difference, continuous capture, and a capture capability of the main control virtual character). Such a multi-dimensional correction mechanism can improve capture flexibility and accuracy, and increase strategy of a game and a sense of participation of a player.

[00137] **Prompt information is displayed based on an aiming circle**

[00138] In some embodiments, an aiming circle for the pet virtual character is displayed in response to a pre-aiming operation on the pet virtual character, and prompt information is displayed based on the aiming circle, the prompt information being configured for guiding the main control virtual character in selecting the capture prop.

[00139] In a possible implementation, a pre-aiming operation on the pet virtual character is implemented based on an aiming joystick. In some embodiments, a capture prop control is displayed on the virtual world picture, and the capture prop control is a control for controlling and capturing the pet virtual character by the main control virtual character. In response to a touch operation on the capture prop control, an aiming joystick corresponding to the capture prop control is displayed and an aiming crosshair is displayed, the aiming joystick being configured for controlling an aiming position indicated by the aiming crosshair, and the aiming crosshair being configured for assisting the virtual character in aiming at the aiming position. For example, an example in which the capture prop is a virtual throwing prop is used. The main control virtual character aims an aiming crosshair at a position A. After being thrown, the capture prop falls at the position A, or the capture prop falls near the position A.

[00140] In some embodiments, the terminal displays, in response to the touch aiming operation on the aiming joystick, the aiming crosshair having the aiming position changed, that is, the aiming position of the aiming crosshair is changed by controlling the aiming joystick, to implement the pre-aiming operation on the pet virtual character. The touch aiming operation on the aiming joystick includes at least one of dragging the aiming joystick, tapping the aiming joystick, and double-tapping the aiming joystick, but is not limited thereto. This is not specifically limited in the implementation of this application.

[00141] In this embodiment of this application, the aiming circle and the corresponding prompt information are displayed to guide the player in selecting a suitable capture prop, thereby improving operation accuracy and game interaction. This design can help the player to perform a capture operation more easily, thereby improving user experience of the game.

[00142] In some embodiments, in response to an aiming operation on the pet virtual character, that is, the aiming crosshair is aimed at the pet virtual character. In this case, an aiming circle for the pet virtual character is displayed, and a level of a capture prop released by the main control virtual character is determined based on prompt information of the aiming circle. In some embodiments, capture props of different levels correspond to different capture capabilities.

[00143] The prompt information is configured for guiding the main control virtual character in selecting a capture prop. In some embodiments, the prompt information displayed based on the aiming circle includes at least one of the following: color information of the aiming circle; style information of the aiming circle; icon information in a first peripheral side position of the aiming circle; and recommendation information in a second peripheral side position of the aiming circle.

[00144] In some embodiments, the aiming circle is displayed as different color information. The different color information represents different capture difficulties. The player determines, according to color information displayed in the aiming circle, difficulty in capturing the pet virtual character, to determine a level of a used capture prop. In some embodiments, the color information may be implemented as at least one of first color information to fourth color information. The first color information is configured for guiding the main control virtual character in selecting a capture prop of a first level, and the first level is a level of a current capture prop of the main control virtual character; the second color information is configured for guiding the main control virtual character in switching a capture prop of a second level to a capture prop of a third level, and a grade of the third level is higher than a grade of the second level; the third color information is configured for guiding the main control virtual character in switching a capture prop of a fourth level to a capture prop of a fifth level, and a grade of the fifth level is lower than a grade of the fourth level; and the fourth color information is configured for guiding the main control virtual character in selecting a capture prop of a first preset level, and the first preset level is a level of a specified capture prop.

[00145] For example, as shown in FIG. 11, a main control virtual character 701 is displayed in a virtual world picture 700, and in response to an aiming operation on a pet virtual character 702, an aiming circle 704 for the pet virtual character 702 is displayed. For example, color information of the aiming circle 704 is displayed as red, which represents that a capture difficulty of the pet virtual character 702 is high, that is, a possibility that the main control virtual character 701 successfully captures the pet virtual character by using a capture prop 703 is low. In some embodiments, the player controls the main control virtual character 701 to switch a level of the capture prop 703. For example, the initial level of the capture prop 703 is an ordinary level, and the player may choose to switch the level of the pre-thrown capture prop 703 from the ordinary level to

an uncommon level, thereby improving a possibility of successful capture. In some embodiments, the pet virtual character 702 is captured based on the switched capture prop 703, and the color information displayed in the aiming circle 704 may change to green, representing a high possibility of successfully capturing the pet virtual character 702.

[00146] In this embodiment of this application, the color information provides a clear visual prompt, and different color information may clearly guide the player in controlling the main control virtual character to select or switch a capture prop of a particular level, so that the player can perform operations accurately and optimize a game strategy and a decision process. In some embodiments, the aiming circle is displayed as different style information. The different style information represents different capture difficulties. The player determines, according to style information displayed in the aiming circle, difficulty in capturing the pet virtual character, to determine a level of a used capture prop. In some embodiments, first style information is displayed based on the aiming circle, the first style information is configured for guiding the main control virtual character in selecting a capture prop of a sixth level, and the sixth level is a level of the current capture prop of the main control virtual character; second style information is displayed based on the aiming circle, the second style information is configured for guiding the main control virtual character in switching a capture prop of a seventh level to a capture prop of an eighth level, and a grade of the eighth level is higher than a grade of the seventh level; third style information is displayed based on the aiming circle, the third style information is configured for guiding the main control virtual character in switching a capture prop of a ninth level to a capture prop of a tenth level, and a grade of the tenth level is lower than a grade of the ninth level; and fourth style information is displayed based on the aiming circle, the fourth style information is configured for guiding the main control virtual character in selecting a capture prop of a second preset level, and the second preset level is a level of a specified capture prop.

[00147] For example, as shown in FIG. 12, a main control virtual character 801 is displayed in a virtual world picture 800, and in response to an aiming operation on a pet virtual character 802, an aiming circle 804 for the pet virtual character 802 is displayed. For example, style information of the aiming circle 804 is displayed as dashed line information, which represents that a capture difficulty of the pet virtual character 802 is low, that is, a possibility that the main control virtual character 801 successfully captures the pet virtual character by using a capture prop 803 is large. The player determines, according to the dashed line information displayed in the aiming circle 804, that the main control virtual character 801 selects to use the current capture prop 803.

[00148] In some embodiments, a first peripheral side position of the aiming circle is displayed as different icon information. The different icon information represents different capture difficulties. The player determines, according to icon information displayed in the aiming circle, difficulty in capturing the pet virtual character, to determine a level of a used capture prop. In some embodiments, first icon information is displayed based on the aiming circle, the first icon information is configured for guiding the main control virtual character in selecting a capture prop of an eleventh level, and the eleventh level is a level of the current capture prop of the main control virtual character; second icon information is displayed based on the aiming circle, the second icon information is configured for guiding the main control virtual character in switching a capture prop of a twelfth level to a capture prop of a thirteenth level, and a grade of the thirteenth level is higher than a grade of the twelfth level; third icon information is displayed based on the aiming circle, the third icon information is configured for guiding the main control virtual character in switching a capture prop of a fourteenth level to a capture prop of a fifteenth level, and a grade of the fifteenth level is lower than a grade of the fourteenth level; and fourth icon information is displayed based on the aiming circle, the fourth icon information is configured for guiding the main control virtual character in selecting a capture prop of a third preset level, and the third preset level is a level of a specified capture prop.

[00149] For example, a difficulty difference between capture difficulty of the pet virtual character indicated by the icon information and real capture difficulty of the pet virtual character has at least one of the following association relationships. There is a negative correlation between the difficulty difference and an observation time of the pet virtual character. For example, as the observation time of the pet virtual character increases, observation details of a behavior manner, an appearance feature, and the like of the pet virtual character are increased, accuracy of indicating capture difficulty of the pet virtual character is increased based on the icon information, and the difficulty difference is reduced. There is a negative correlation between the difficulty difference and the number of historical captures of the pet virtual character. For example, as the number of historical captures of the pet virtual character increases, a degree of understanding of the pet virtual character is increased, accuracy of indicating capture difficulty of the pet virtual character is increased based on the icon information, and the difficulty difference is reduced.

[00150] For example, as shown in FIG. 13, a main control virtual character 901 is displayed in a virtual world picture 900, and in response to an aiming operation on a pet virtual character 902, an aiming circle 904 for the pet virtual character 902 is displayed. Icon information displayed in a first peripheral side position of the aiming circle 904. For example, the icon information displayed above the aiming circle 904 is star level information, and the capture difficulty is quickly

determined according to the number of star levels. For example, three stars indicate high capture difficulty, and the player controls the main control virtual character 901 to switch a level of a capture prop 903. For example, the capture prop 903 is switched from a common level to a master level.

[00151] In some embodiments, a second peripheral side position of the aiming circle is displayed as different recommendation information. The different recommendation information represents different capture difficulties. The player determines, according to recommendation information displayed in the aiming circle, difficulty in capturing the pet virtual character, to determine a level of a used capture prop. In some embodiments, the recommendation information is presented as a level of a recommended specific capture prop.

[00152] For example, as shown in FIG. 14, a main control virtual character 1001 is displayed in a virtual world picture 1000, and in response to an aiming operation on a pet virtual character 1002, an aiming circle 1004 for the pet virtual character 1002 is displayed. Icon information displayed in a second peripheral side position of the aiming circle 1004. For example, the recommendation information is displayed above the aiming circle 1004. In some embodiments, the recommendation information is presented as a level of a recommended specific capture prop. For example, the recommendation information is displayed as "recommend a prop of level XX", and the player may select a level of a capture prop 1003 according to the recommendation information. The level of the capture prop 1003 is a level of a specified capture prop. In this embodiment of this application, visualization and functionality of the aiming circle are enhanced by using prompt information in various forms (a color, a style, an icon, and recommendation information). Display of the prompt information can effectively instruct a player to select a proper capture prop, thereby improving operation accuracy of a game and decision-making efficiency of the player.

[00153] FIG. 15 is an overall flowchart of a pet virtual character capture method according to an exemplary embodiment of this application. Operations of the method may be performed by the first terminal 120 and/or the second terminal 160 and/or the third terminal 180 in the computer system shown in FIG. 1, or applied to a client that supports a virtual world and that is installed on the first terminal 120 and/or the second terminal 160 and/or the third terminal 180, and overall operations are as follows:

[00154] **Operation 1**: Aim at a pet virtual character after the pet virtual character is found.

[00155] In some embodiments, in response to a pre-aiming operation on a pet virtual character, auxiliary aiming is enabled during aiming, and ray detection is performed. The ray detection is configured for detecting whether a straight line (a ray) sent from a point intersects with

an object in a virtual world. During capture and aiming, a ray is sent from an aiming position from the main control virtual character, and it is detected whether the ray intersects with any capturable pet virtual character. If yes, a server calculates a capture value, and displays an aiming crosshair on a virtual world picture.

[00156] During capture and aiming, with reference to FIG. 16, for proper program design and reduction of a degree of coupling between modules, a layered design architecture is used, which is presented as a logic layer (a logic state machine), a data layer (an animation state component), and a performance layer (an animation instance). This design architecture lowers a degree of coupling between modules and avoids cross-module access. A change of the logic layer affects data in the data layer, and the performance layer takes out data from the data layer for animation performance. For example, data is written into the animation state component by using the logic state machine, a player operation drives a logic state to change, the logic state drives data to change, and the data drives an animation to change. The logic state machine is a layered finite state machine with a network synchronization capability, and may implement separation and synchronization of movement and behavior logic states. In some embodiments, the logic state machine is divided into two layers, one layer is a movement state layer, and the other layer is a behavior state layer, and a state of each layer can only have one state node at the same time. Logic of the behavior state layer is independent of logic of the movement state layer. For example, a person aiming state is a behavior state. For example, a player may be in an aiming state (a behavior state) and be capable of running (a movement state) or riding (a movement state). Logic of the aiming state is independent of logic of running and riding, and is implemented without considering other associated movement states.

[00157] In the aiming state, with reference to FIG. 17, a behavior and an animation of the main control virtual character in the aiming state are controlled by using a person aiming logic state machine, and a capture prop logic state machine is configured for managing a behavior and an animation of a capture prop in a hand of the main control virtual character. The person aiming logic state machine defines different states (such as standing, walking, and aiming) that the main control virtual character may be in, and how to switch from a state (such as walking) to another state (such as aiming) according to different events. The capture prop logic state machine is configured to process various states of a capture prop, such as static, moving, and thrown. Data of the capture prop is set by using the capture prop logic state machine, for example, a speed of the capture prop is set, and an animation instance of the capture prop is displayed based on the set data. The animation instance of the capture prop refers to a specific animation effect presented by the capture prop in the game, such as throwing or rotating. In some embodiments, a binding notification is performed on

the person aiming logic state machine and the capture prop logic state machine. When the state of the main control virtual character changes, the binding notification ensures that the state of the capture prop is updated correspondingly, to keep synchronization between the two states. For example, when the main control virtual character is ready to throw, the animation of the capture prop is displayed in a state of being clenched or ready to be thrown.

[00158] In some embodiments, the capture prop is used as a projectile created by both the client and the server. In some embodiments, after the capture prop is thrown by the main control virtual character, there are three synchronization strategies according to configurations: One synchronization strategy is synchronizing the client and the server, where the server creates a moving track of a capture prop and then synchronizes the moving track to the client. One synchronization strategy is performing separate creation by two ends (the client and the server), where each of the server and the client creates a moving track of a capture prop, and the two capture props are not associated from a network level, and separately move. One synchronization strategy is performing creation by only the client and not performing creation by the server, and is usually configured for presenting a special effect animation of a capture prop.

[00159] In some embodiments, the capture prop has a parabolic movement capability and an ejection capability. In some embodiments, the capture prop further has a capability of dynamically adjusting the size of collision detection according to a configuration curve. The configuration curve is a mathematical tool, configured for defining a relationship of one or more variables changing with time or another parameter. During collision detection, the configuration curve may be configured for defining how the size of collision detection changes with a game process, an object state, or another factor. Collision detection is an important link in a game, and is configured for determining whether two or more objects touch or pass through each other. For example, collision detection is configured for determining whether a capture prop touches a pet virtual character, that is, whether a capture prop hits a pet virtual character.

[00160] **Operation 2**: Present an aiming circle after the aiming is locked.

[00161] In some embodiments, in response to a capture operation on the pet virtual character, the main control virtual character is controlled to release a capture prop or a capture skill for the pet virtual character. In some embodiments, the main control virtual character is controlled to release a capture prop for the pet virtual character. A capture prop control is displayed on the virtual world picture. In response to a touch operation on the capture prop control, an aiming crosshair corresponding to the capture prop control is displayed, the aiming crosshair is aimed at the pet virtual character, an aiming circle for the pet virtual character and prompt information based on the

aiming circle are displayed, and the prompt information is configured for guiding the main control virtual character in selecting a capture prop.

[00162] **Operation 3**: Hit the pet virtual character, and enter capture determining.

[00163] In some embodiments, after the main control virtual character hits the pet virtual character by using the capture prop, or after the main control virtual character is controlled to release a capture skill (a capture spell-casting effect) for the pet virtual character, a capture determining stage is entered. The server calculates a capture value for the pet virtual character according to the hit information and each piece of data that are reported by the client and according to a capture formula, obtains a capture probability of capturing the pet virtual character according to the capture value, determines a probability interval to which the capture probability belongs according to the capture probability, determines a corresponding capture effect, and instructs the client to display capture effects for the pet virtual character within different probability intervals.

[00164] With reference to FIG. 18, a capture main control end refers to behaviors and logic that are seen by Player A on a local client. A capture simulation end refers to behaviors and logic that are seen by Player A on a client of another player (for example, teammate Player B). After Player A initiates a capture request on the local client, the local client sends the capture request to the server, and the server processes the capture request of the player and encapsulates the capture request into an executable command. Then, the server instructs the client to perform capture performance, that is, instructs the client to display capture effects for the pet virtual character within different probability intervals. In some embodiments, the server notifies the local client to display a capture effect for the pet virtual character, or the server notifies a client of another player to display a capture effect for the pet virtual character. The server calculates the time required for the capture performance, and writes a capture result into backpack data of the player, for example, including information such as whether the pet is successfully captured and a state of the pet. A User Interface (UI) of the player displays the capture performance after receiving the notification of the server, and the player can see an animation of the capture process, for example, the count of shaking of the capture prop and shaking intensity. In some embodiments, the server sends, in a form of the Reliable Remote Procedure Call (RPC), the capture result to a player (such as Player A) of the capture main control end for performance, to ensure accurate data transmission; and sends, in a form of Unreliable Multicast, the capture result to a player (for example, teammate Player B) of the capture simulation end for performance, to save bandwidth.

[00165] **Operation 4a**: Directly determine, when a capture probability falls within a first probability interval, that the capture succeeds. In some embodiments, a capture probability for the

pet virtual character is calculated based on a capture value. When the capture value falls within a first range interval, the capture probability of the pet virtual character falls within the first probability interval. When the capture probability of the pet virtual character falls within a first probability interval, it is determined that a capture type for the pet virtual character is a successful capture type and a capture effect for the pet virtual character is displayed as a successful capture effect, the successful capture effect being configured for representing that the pet virtual character does not struggle and is successfully captured.

[00166] **Operation 4b**: Determine, when a capture probability falls within a second probability interval, that the capture succeeds, and shake a capture prop for three times. In some embodiments, when the capture value falls within a second range interval, the capture probability of the pet virtual character falls within the second probability interval. When the capture probability of the pet virtual character falls within a second probability interval, it is determined that a capture type for the pet virtual character is a successful capture type and a capture effect for the pet virtual character is displayed as a struggle capture effect, the struggle capture effect being configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles. For example, the pet virtual character is successfully captured after being shaken in the capture prop for three times.

[00167] **Operation 4c**: Generate four random numbers when the capture probability falls within a target probability interval, and compare the four random numbers with the capture probability. In some embodiments, when the capture value falls within a target range interval, the capture probability of the pet virtual character falls within the target probability interval. When the capture probability of the pet virtual character falls within a target probability interval, it is determined that a capture type for the pet virtual character is a challenge capture type and a challenge capture effect (including a struggle capture effect or a struggle escape effect) for the pet virtual character is displayed. In some embodiments, n random numbers are generated within the target probability interval according to a preset rule, n being a positive integer. the count of struggle of the pet virtual character in the capture prop is determined based on a magnitude relationship between the n random numbers and the capture probability. In some embodiments, a count in the n random numbers that is greater than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. Alternatively, a count in the n random numbers that is less than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. In some embodiments, a count in the n random numbers that is greater than the capture probability is determined as the count of struggle of the pet virtual character in the capture prop. In some embodiments, the capture effect for the pet virtual character is determined as the

struggle capture effect if a count of struggle of the pet virtual character in the capture prop is less than or equal to a count threshold; and the capture effect for the pet virtual character is determined as the struggle escape effect if a count of struggle of the pet virtual character in the capture prop is greater than a count threshold.

[00168] **Operation 4d**: Determine, when a capture probability falls within a third probability interval, that the direct capture fails, and the capture prop ricochets. In some embodiments, when the capture value falls within a third range interval, the capture probability of the pet virtual character falls within the third probability interval. When the capture probability of the pet virtual character falls within a third probability interval, it is determined that a capture type for the pet virtual character is a failed capture type and a capture effect for the pet virtual character is displayed as a failed capture effect, the failed capture effect being configured for representing that the pet virtual character does not struggle and fails to be captured. For example, the failed capture effect is displayed as that the capture prop ricochets.

[00169] **Operation 4e**: Illegal unit, prompting that the illegal unit is uncapturable. In some embodiments, in response to a throwing operation on the capture prop, after the capture prop hits an illegal unit, a text prompt of uncapturable is displayed on the game interface. The illegal unit refers to a non-pet virtual character, such as a stone, a tree, or a lawn.

[00170] **Operation 5**: Shake the capture prop for the first time, and display a capture progress bar. In some embodiments, a capture progress bar is displayed at a peripheral side position of the capture prop in response to that the pet virtual character starts to struggle for the first time in the capture prop.

[00171] **Operation 6**: Reach a target capture progress, where the count of shaking is less than or equal to a count threshold, and the capture succeeds.

[00172] In some embodiments, the capture progress bar increases to a target capture progress (for example, the target capture progress is 100% progress) at a preset rate, and before the capture progress bar reaches the target capture progress, the count of struggle of the pet virtual character in the capture prop is determined. The challenge capture effect for the pet virtual character is displayed as the struggle capture effect if the count of struggle is less than or equal to a count threshold.

[00173] **Operation 7**: Fail to reach the target capture progress, where the count of shaking is greater than the count threshold, and the capture fails.

[00174] In some embodiments, the capture progress bar increases to a target capture progress (for example, the target capture progress is 100% progress) at a preset rate, and before the capture

progress bar reaches the target capture progress, the count of struggle of the pet virtual character in the capture prop is determined. The challenge capture effect for the pet virtual character is displayed as the struggle escape effect if the count of struggle is greater than a count threshold.

[00175] FIG. 19 is a structural block diagram of a pet virtual character capture apparatus according to an embodiment of this application. The apparatus has a function of performing the foregoing pet virtual character capture method embodiments. The function may be implemented by hardware or may be implemented by hardware executing corresponding software. The apparatus may be the foregoing described server, or may be disposed in the server. As shown in FIG. 19, the apparatus 1700 includes a display module 1710 and a control module 1720. The display module 1710 is configured to execute operation 210 in the embodiment of FIG. 3. The control module 1720 is configured to perform operation 220 in the embodiment in FIG. 3. The display module 1710 is configured to execute operation 230 in the embodiment of FIG. 3. The struggle capture effect is configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles, and the struggle escape effect is configured for representing that the pet virtual character escapes from capture after the pet virtual character struggles.

[00176] In some embodiments, the display module 1710 further includes a display submodule. In an exemplary embodiment, the display submodule is configured to perform operation 231 and operation 232 in the embodiment of FIG. 5.

[00177] In an exemplary embodiment, the display submodule is configured to perform operation 231a and operation 232a in the embodiment of FIG. 6.

[00178] In an exemplary embodiment, the display submodule is configured to display the challenge capture effect for the pet virtual character as the struggle escape effect in response to that a count of struggle of the pet virtual character in the capture prop is greater than a count threshold.

[00179] In an exemplary embodiment, the display submodule is configured to display the challenge capture effect for the pet virtual character when a capture probability of the pet virtual character falls within a target probability interval.

[00180] In some embodiments, the apparatus 1700 further includes a generation module and a determining module.

[00181] In an exemplary embodiment, the generation module is configured to generate n random numbers according to a preset rule, n being a positive integer; and

[00182] the determining module is configured to determine, based on a magnitude relationship between the n random numbers and the capture probability, the count of struggle of the pet virtual character in the capture prop.

[00183] In an exemplary embodiment, the determining module is configured to determine a count in the n random numbers that is greater than the capture probability as the count of struggle of the pet virtual character in the capture prop.

[00184] In an exemplary embodiment, the determining module is configured to determine, when the capture probability of the pet virtual character falls within a first probability interval, that a capture type for the pet virtual character is a successful capture type and display a capture effect for the pet virtual character as a successful capture effect, the successful capture effect being configured for representing that the pet virtual character does not struggle and is successfully captured; the determining module is configured to determine, when the capture probability of the pet virtual character falls within a second probability interval, that a capture type for the pet virtual character is a successful capture type and display a capture effect for the pet virtual character as a struggle capture effect; and the determining module is configured to determine, when the capture probability of the pet virtual character falls within a third probability interval, that a capture type for the pet virtual character is a failed capture type and displaying a capture effect for the pet virtual character as a failed capture effect, the failed capture effect being configured for representing that the pet virtual character does not struggle and fails to be captured; and a minimum value of the first probability interval being greater than a maximum value of the second probability interval, a minimum value of the second probability interval being greater than a maximum value of the target probability interval, and a minimum value of the target probability interval being greater than a maximum value of the third probability interval.

[00185] In some embodiments, the apparatus 1700 further includes an obtaining module. In an exemplary embodiment, the obtaining module is configured to obtain a basic capture value for the pet virtual character, the basic capture value being configured for indicating a basic capture probability of capturing the pet virtual character; and the obtaining module is configured to obtain a correction type, and correcting the basic capture probability based on content indicated by the correction type to obtain the capture probability. In an exemplary embodiment, the correction type includes at least one of the following: state correction, configured for indicating correction of the basic capture value by a state of the pet virtual character; gain correction, configured for indicating correction of the basic capture value by a positive gain effect and/or a negative gain effect; level suppression correction, configured for indicating correction of the basic capture value by a level

difference between the main control virtual character and the pet virtual character; continuous capture compensation correction, configured for indicating correction of the basic capture value by a continuous capture operation performed by the main control virtual character; and capture force correction, configured for indicating correction of the basic capture value by a capture capability of the main control virtual character.

[00186] In an exemplary embodiment, the display submodule is configured to display an aiming circle for the pet virtual character in response to a pre-aiming operation on the pet virtual character, and display prompt information based on the aiming circle, the prompt information being configured for guiding the main control virtual character in selecting the capture prop. In an exemplary embodiment, the prompt information displayed based on the aiming circle includes at least one of the following: color information of the aiming circle; style information of the aiming circle; icon information in a first peripheral side position of the aiming circle; and recommendation information in a second peripheral side position of the aiming circle. In an exemplary embodiment, the color information includes at least one of the following: first color information, the first color information being configured for guiding the main control virtual character in selecting a capture prop of a first level, and the first level being a level of a current capture prop of the main control virtual character; second color information, the second color information being configured for guiding the main control virtual character in switching a capture prop of a second level to a capture prop of a third level, and a grade of the third level being higher than a grade of the second level; third color information, the third color information being configured for guiding the main control virtual character in switching a capture prop of a fourth level to a capture prop of a fifth level, and a grade of the fifth level being lower than a grade of the fourth level; and fourth color information, the fourth color information being configured for guiding the main control virtual character in selecting a capture prop of a first preset level, and the first preset level being a level of a specified capture prop.

[00187] For specific limitations in one or more embodiments of the foregoing provided pet virtual character capture apparatus, refer to the foregoing limitations on the pet virtual character capture method. Details are not described herein again. The modules in the foregoing apparatus may be implemented entirely or partially by software, hardware, or a combination thereof. The foregoing modules may be built in or independent of a processor of a computer device in a hardware form, or may be stored in a memory of the computer device in a software form, so that the processor invokes and performs an operation corresponding to each of the foregoing modules. The sequence numbers of the foregoing embodiments of this application are merely for description, but do not indicate the preference among the embodiments.

[00188] An embodiment of this application further provides a computer device, including a processor and a memory, the memory having at least one computer program stored therein, and the at least one computer program being loaded and executed by the processor to implement the pet virtual character capture method according to the foregoing method embodiments.

[00189] FIG. 20 is a structural block diagram of a computer device 1800 according to an exemplary embodiment of this application. The computer device 1800 may be a portable mobile terminal, for example, a smartphone, a tablet computer, a moving picture experts group audio layer III (MP3) player, or a moving picture experts group audio layer IV (MP4) player. The computer device 1800 may also be referred to as another name such as user equipment or a portable terminal. Generally, the computer device 1800 includes a processor 1801 and a memory 1802.

[00190] The processor 1801 may include one or more processing cores, for example, the processor may be a 4-core processor or an 8-core processor. The processor 1801 may be implemented in at least one hardware form of digital signal processing (DSP), a field programmable gate array (FPGA), or a programmable logic array (PLA). The processor 1801 may also include a main processor and a coprocessor. The main processor is a processor configured to process data in an awake state, and is also referred to as a central processing unit (CPU). The coprocessor is a low power consumption processor configured to process data in a standby state. In some embodiments, the processor 1801 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. In some embodiments, the processors 1801 may further include an artificial intelligence (AI) processor. The AI processor is configured to process computing operations related to machine learning.

[00191] The memory 1802 may include one or more computer-readable storage media. The computer-readable storage medium may be tangible and non-transitory. The memory 1802 may also include a high-speed random-access memory, as well as non-volatile memory, such as one or more disk storage devices and flash storage devices. In some embodiments, a non-transitory computer-readable storage medium in the memory 1802 is configured to store at least one instruction, the at least one instruction being configured to be executed by the processor 1801 to implement the pet virtual character capture method provided in the embodiments of this application.

[00192] In some embodiments, the computer device 1800 further includes a peripheral device interface 1803 and at least one peripheral device. The processor 1801, the memory 1802, and the peripheral device interface 1803 may be connected via a bus or signal line. Each peripheral device may be connected to the peripheral interface 1803 through the bus, the signal cable, or a circuit board. Specifically, the peripheral device includes at least one of a radio frequency (RF) circuit

1804, a touch display screen 1805, a camera assembly 1806, an audio circuit 1807, and a power supply 1808.

[00193] The peripheral device interface 1803 may be configured to connect at least one peripheral device related to input/output (I/O) to the processor 1801 and the memory 1802. In some embodiments, the processor 1801, the memory 1802, and the peripheral device interface 1803 are integrated on the same chip or the same circuit board. In some other embodiments, any one or two of the processor 1801, the memory 1802, and the peripheral device interface 1803 may be implemented on an independent chip or circuit board, which is not limited in the embodiments. The RF circuit 1804 is configured to receive and transmit RF signals, which are also referred to as electromagnetic signals. The RF circuit 1804 communicates with a communication network and other communication devices via electromagnetic signals. The RF circuit 1804 converts an electric signal into an electromagnetic signal for sending, or converts a received electromagnetic signal into an electric signal. In some embodiments, the radio frequency circuit 1804 includes: an antenna system, an RF transceiver, one or more amplifiers, a tuner, an oscillator, a digital signal processor, a codec chip set, a subscriber identity module card, and the like. The radio frequency circuit 1804 may communicate with other terminals through at least one wireless communication protocol. The wireless communication protocol includes, but is not limited to, a world wide web, a metropolitan area network, an intranet, various generations of mobile communication networks (2G, 3G, 4G, and 5G), a wireless local area network, and/or a wireless fidelity (Wi-Fi) network. In some embodiments, the radio frequency circuit 1804 may further include a circuit related to near field communication (NFC), which is not limited in this application. The touch display screen 1805 is configured to display a UI. The UI may include graphics, text, icons, videos, and any combination thereof. The touch display screen 1805 also has a capability of acquiring a touch signal on or above a surface of the touch display screen 1805. The touch signal may be inputted, as a control signal, into the processor 1801 for processing. The touch display screen 1805 is configured to provide a virtual button and/or a virtual keyboard, which is also referred to as a soft button and/or a soft keyboard. In some embodiments, there may be one touch display screen 1805, arranged on a front panel of the computer device 1800. In some other embodiments, there are at least two touch display screens 1805, arranged on different surfaces of the computer device 1800 respectively or in a folded design. In some embodiments, the touch display screen 1805 may be a flexible display screen, arranged on a curved surface or a folded surface of the computer device 1800. The touch display screen 1805 may be further arranged as a non-rectangular irregular pattern, i.e., a special-shaped screen. The touch display screen 1805 may be made of a material such as a liquid crystal display (LCD) or an organic light-emitting diode (OLED). The camera assembly 1806 is configured to

collect images or videos. In some embodiments, the camera assembly 1806 includes a front-facing camera and a rear-facing camera. Generally, the front-facing camera is configured to implement a video call or self-portrait. The rear-facing camera is configured to shooting a picture or a video. In some embodiments, there are at least two rear-facing cameras, each of which is any one of a main camera, a depth of field camera and a wide-angle camera, to implement a background blurring function by fusing the main camera and the depth of field camera, and panoramic shooting and VR shooting functions by fusing the main camera and the wide-angle camera. In some embodiments, the camera component 1806 may further include a flash. The flash may be a single-color temperature flash or a double-color temperature flash. The double color temperature flash is a combination of a warm flash and a cold flash, and may be configured to perform light ray compensation at different color temperatures. The audio circuit 1807 is configured to provide an audio interface between a user and the computer device 1800. The audio circuit 1807 may include a microphone and a speaker. The microphone is configured to acquire sound waves of users and surroundings, and convert the sound waves into electrical signals and input the signals to the processor 1801 for processing, or input the signals to the RF circuit 1804 to implement voice communication. For the purpose of stereo sound acquisition or noise reduction, there may be a plurality of microphones, respectively arranged at different parts of the computer device 1800. The microphone may be further an array microphone or an omnidirectional microphone. The speaker is configured to convert electric signals from the processor 1801 or the radio frequency circuit 1804 into sound waves. The speaker may be a conventional thin-film speaker or a piezoelectric ceramic speaker. When the speaker is the piezoelectric ceramic speaker, the speaker can not only convert an electrical signal into sound waves audible to a human being, but also convert an electrical signal into sound waves inaudible to the human being for ranging and other purposes. In some embodiments, the audio circuit 1807 may also include an earphone jack. The power supply 1808 is configured to supply power to components in the computer device 1800. The power supply 1808 may be an alternating-current power supply, a direct-current power supply, a disposable battery, or a rechargeable battery. When the power supply 1808 includes the rechargeable battery, the rechargeable battery may be a wired rechargeable battery or a wireless rechargeable battery. The wired rechargeable battery is a battery charged through a wired line, and the wireless rechargeable battery is a battery charged through a wireless coil. The rechargeable battery may be further configured to support a fast-charging technology.

[00194] In some embodiments, the computer device 1800 may further include one or more sensors 1809. The one or more sensors 1809 include, but not limited to, an acceleration sensor

1810, a gyro sensor 1811, a pressure sensor 1812, an optical sensor 1813, and a proximity sensor 1814.

[00195] The acceleration sensor 1810 may detect acceleration on three coordinate axes of a coordinate system established by the computer device 1800. For example, the acceleration sensor 1810 is configured to detect components of gravity acceleration on the three coordinate axes. The processor 1801 may control, according to a gravity acceleration signal acquired by the acceleration sensor 1810, the touch display screen 1805 to display the user interface in a transverse view or a longitudinal view. The acceleration sensor 1810 may be further configured to collect data of a game or a user movement. The gyro sensor 1811 may detect a body direction and a rotation angle of the computer device 1800. The gyro sensor 1811 may cooperate with the acceleration sensor 1810 to acquire a 3D motion by the user on the computer device 1800. The processor 1801 may implement the following functions according to the data acquired by the gyro sensor 1811: motion sensing (such as changing the UI according to a tilt operation of the user), image stabilization at shooting, game control, and inertial navigation. The pressure sensor 1812 may be disposed at a side frame of the computer device 1800 and/or a lower layer of the touch display screen 1805, when the pressure sensor 1812 is arranged at the side frame of the computer device 1800, a holding signal of the user on the computer device 1800 may be detected, and left/right hand identification and a quick motion may be performed according to the holding signal. When the pressure sensor 1812 is arranged at the lower layer of the touch display screen 1805, an operable control on the UI interface can be controlled according to a pressure operation of the user on the touch display screen 1805. The operable control includes at least one of a button control, a scroll bar control, an icon control, and a menu control. The optical sensor 1813 is configured to collect the ambient light intensity. In an embodiment, the processor 1801 may control display brightness of the touch display screen 1805 according to the ambient light intensity acquired by the optical sensor 1813. Specifically, when the ambient light intensity is high, the display brightness of the touch display screen 1805 is increased; and when the ambient light intensity is low, the display brightness of the touch display screen 1805 is reduced. In another embodiment, the processor 1801 may further dynamically adjust a camera parameter of the camera assembly 1806 based on the ambient light intensity collected by the optical sensor 1813. The proximity sensor 1814, also referred to as a distance sensor, is generally disposed on the front surface of the computer device 1800. The proximity sensor 1814 is configured to collect a distance between the user and a front surface of the computer device 1800. In an embodiment, when the proximity sensor 1814 detects that the distance between the front face of the user and the front face of the computer device 1800 gradually becomes small, the touch display screen 1805 is controlled by the processor 1801 to switch from a screen-on state to a screen-off

state. When the proximity sensor 1814 detects that the distance between the user and the front surface of the computer device 1800 gradually becomes large, the touch display screen 1805 is controlled by the processor 1801 to switch from the screen-off state to the screen-on state. A person skilled in the art may understand that the foregoing structure does not constitute any limitation on the computer device 1800, and the computer device may include more components or fewer components than those shown in the figure, or some components may be combined, or a different component deployment may be used. An embodiment of this application further provides a computer-readable storage medium, having at least one computer program stored therein, the at least one computer program being loaded and executed by a processor to implement the pet virtual character capture method according to the foregoing method embodiments. An embodiment of this application further provides a computer program product, including at least one computer program, the at least one computer program being stored in a computer-readable storage medium, and a processor of a computer device reading the at least one computer program from the computer-readable storage medium and executing the at least one computer program, to cause the computer device to perform the pet virtual character capture method according to the foregoing method embodiments.

[00196] That "plurality of" mentioned in this specification means two or more. "And/or" describes an association relationship between associated objects and indicates that three relationships may exist. For example, A and/or B may indicate the following three cases: Only A exists, both A and B exist, and only B exists. The character "/" generally represents that the association objects before and after the character are in an "or" relationship. In addition, the operation numbers described herein merely exemplarily show a possible performance sequence of the operations. In some other embodiments, the operations may not be performed according to the sequence of the numbers. For example, two operations with different numbers may be performed simultaneously, or two operations with different numbers may be performed according to a sequence contrary to the sequence shown in the figure. This is not limited in the embodiments of this application.

[00197] A person of ordinary skill in the art may understand that all or part of the operations of implementing the foregoing embodiments may be implemented by hardware, or may be implemented by a program instructing related hardware. The program may be stored in a computer-readable storage medium. The foregoing storage medium may be a read-only memory, a magnetic disk, an optical disc, or the like.

CLAIMS

1. A pet virtual character capture method, performed by a terminal, and comprising:

displaying a main control virtual character located in a virtual world, and a pet virtual character in a wild state;

controlling, in response to a capture operation on the pet virtual character, the main control virtual character to release a capture prop or a capture skill for the pet virtual character; and

displaying a challenge capture effect for the pet virtual character, the challenge capture effect comprising a struggle capture effect or a struggle escape effect; and the struggle capture effect being configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles, and the struggle escape effect being configured for representing that the pet virtual character escapes from capture after the pet virtual character struggles.

2. The method according to claim 1, wherein the displaying a challenge capture effect for the pet virtual character comprises:

displaying a challenge capture effect for the pet virtual character as a struggle capture effect when a first capture condition is satisfied, the struggle capture effect comprising that the pet virtual character is successfully captured after the pet virtual character struggles in a first form; and

displaying a challenge capture effect for the pet virtual character as a struggle escape effect when a second capture condition is satisfied, the struggle capture effect comprising that the pet virtual character escapes from capture after the pet virtual character struggles in a second form.

3. The method according to claim 2, wherein the displaying a challenge capture effect for the pet virtual character as a struggle capture effect when a first capture condition is satisfied comprises:

displaying the challenge capture effect for the pet virtual character as the struggle capture effect in response to that a count of struggle of the pet virtual character in the capture prop is less than or equal to a count threshold.

4. The method according to claim 2, wherein the displaying a challenge capture effect for the pet virtual character as a struggle escape effect when a second capture condition is satisfied comprises:

displaying the challenge capture effect for the pet virtual character as the struggle escape effect in response to that a count of struggle of the pet virtual character in the capture prop is greater than a count threshold.

5. The method according to any one of claims 1 to 4, further comprising:

displaying the challenge capture effect for the pet virtual character when a capture probability of the pet virtual character falls within a target probability interval.

6. The method according to any one of claims 3 to 5, further comprising:

generating n random numbers according to a preset rule, n being a positive integer; and

determining, based on a magnitude relationship between the n random numbers and the capture probability, the count of struggle of the pet virtual character in the capture prop.

7. The method according to claim 6, wherein the determining, based on a magnitude relationship between the n random numbers and the capture probability, the count of struggle of the pet virtual character in the capture prop comprises:

determining a count in the n random numbers that is greater than the capture probability as the count of struggle of the pet virtual character in the capture prop.

8. The method according to any one of claims 5 to 7, further comprising at least one of the following:

determining, when the capture probability of the pet virtual character falls within a first probability interval, that a capture type for the pet virtual character is a successful capture type and displaying a capture effect for the pet virtual character as a successful capture effect, the successful capture effect being configured for representing that the pet virtual character does not struggle and is successfully captured;

determining, when the capture probability of the pet virtual character falls within a second probability interval, that a capture type for the pet virtual character is a successful capture type and displaying a capture effect for the pet virtual character as a struggle capture effect; and

determining, when the capture probability of the pet virtual character falls within a third probability interval, that a capture type for the pet virtual character is a failed capture type and displaying a capture effect for the pet virtual character as a failed capture effect, the failed capture effect being configured for representing that the pet virtual character does not struggle and fails to be captured; and a minimum value of the first probability interval being greater than a maximum value of the second probability interval, a minimum value of the second probability interval being greater than a maximum value of the target probability interval, and a minimum value of the target probability interval being greater than a maximum value of the third probability interval.

9. The method according to claim 7 or 8, further comprising: obtaining a basic capture value

for the pet virtual character, the basic capture value being configured for indicating a basic capture probability of capturing the pet virtual character; and

obtaining a correction type, and correcting the basic capture probability based on content indicated by the correction type to obtain the capture probability.

10. The method according to claim 9, wherein the correction type comprises at least one of the following: state correction, configured for indicating correction of the basic capture value by a state of the pet virtual character; gain correction, configured for indicating correction of the basic capture value by a positive gain effect and/or a negative gain effect; level suppression correction, configured for indicating correction of the basic capture value by a level difference between the main control virtual character and the pet virtual character; continuous capture compensation correction, configured for indicating correction of the basic capture value by a continuous capture operation performed by the main control virtual character; and capture force correction, configured for indicating correction of the basic capture value by a capture capability of the main control virtual character.

11. The method according to any one of claims 1 to 10, further comprising: displaying an aiming circle for the pet virtual character in response to a pre-aiming operation on the pet virtual character, and displaying prompt information based on the aiming circle, the prompt information being configured for guiding the main control virtual character in selecting the capture prop.

12. The method according to claim 11, wherein the prompt information comprises at least one of the following: color information of the aiming circle; style information of the aiming circle; icon information in a first peripheral side position of the aiming circle; and recommendation information in a second peripheral side position of the aiming circle.

13. The method according to claim 12, wherein the color information comprises at least one of the following:

first color information, the first color information being configured for guiding the main control virtual character in selecting a capture prop of a first level, and the first level being a level of a current capture prop of the main control virtual character;

second color information, the second color information being configured for guiding the main control virtual character in switching a capture prop of a second level to a capture prop of a third level, and a grade of the third level being higher than a grade of the second level;

third color information, the third color information being configured for guiding the main control virtual character in switching a capture prop of a fourth level to a capture prop of a fifth

level, and a grade of the fifth level being lower than a grade of the fourth level; and

fourth color information, the fourth color information being configured for guiding the main control virtual character in selecting a capture prop of a first preset level, and the first preset level being a level of a specified capture prop.

14. A pet virtual character capture apparatus, comprising:

a display module, configured to display a main control virtual character located in a virtual world, and a pet virtual character in a wild state; and

a control module, configured to control, in response to a capture operation on the pet virtual character, the main control virtual character to release a capture prop or a capture skill for the pet virtual character,

the display module being configured to display a challenge capture effect for the pet virtual character, the challenge capture effect comprising a struggle capture effect or a struggle escape effect; and the struggle capture effect being configured for representing that the pet virtual character is successfully captured after the pet virtual character struggles, and the struggle escape effect being configured for representing that the pet virtual character escapes from capture after the pet virtual character struggles.

15. A computer device, the computer device comprising: a processor and a memory, the memory having at least one computer program stored therein, and the at least one computer program being loaded and executed by the processor to implement the pet virtual character capture method according to any one of claims 1 to 13.

16. A computer storage medium, the computer-readable storage medium having at least one computer program stored therein, and the at least one computer program being loaded and executed by a processor, to implement the pet virtual character capture method according to any one of claims 1 to 13.

17. A computer program product, comprising a computer program, the computer program being stored in a computer-readable storage medium, and being read from the computer-readable storage medium and executed by a processor of a computer device, causing the computer device to perform the pet virtual character capture method according to any one of claims 1 to 13.