CLAIMS

What is claimed is:

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

displaying a user interface, the user interface comprising an image corresponding to a virtual environment and a first virtual object located in the virtual environment;

obtaining an execution range of a first operation of the first virtual object in the virtual environment;

offsetting, based on a density of a plurality of second virtual objects in the execution range satisfying a first condition, a logical position of at least one second virtual object in the plurality of second virtual objects, to obtain an offset logical position of the at least one second virtual object, the logical position being an initial position used in processing the second virtual object in a logical layer of a client of the terminal; wherein an interface position of the second virtual object does not change according to a change of the logical position, the interface position of the second virtual object being a position used when a representation layer of the client processes the second virtual object; wherein a distance between the offset logical positions of every two second virtual objects in the plurality of second virtual objects is greater than or equal to a first distance; and

selecting a target virtual object, on which corresponding to the first operation is performed, from the plurality of second virtual objects based on the offset logical position of the at least one second virtual object;

wherein there are n second virtual objects, n being an integer greater than 1, and

the offsetting a logical position of at least one second virtual object in the plurality of second virtual objects, to obtain an offset logical position of the one second virtual object comprises:

determining an offset direction of an i^{th} second virtual object in the n second virtual objects and determining an offset distance of the i^{th} second virtual object, the offset direction being a direction further away from other second virtual objects, and i being a positive integer less than or equal to n; and

moving a logical position of the i^{th} second virtual object along the offset direction by the offset distance, to obtain an offset logical position of the i^{th} second virtual object.

~~2. The method according to claim 1, wherein an interface position of the second virtual object does not change according to a change of the logical position.~~

~~3. The method according to claim 1, wherein there are n second virtual objects, n being an integer greater than 1, and~~

~~the offsetting a logical position of at least one second virtual object in the plurality of second virtual objects, to obtain an offset logical position of the one second virtual object comprises:~~

~~determining an offset direction of an i^{th} second virtual object in the n second virtual objects and determining an offset distance of the i^{th} second virtual object, the offset direction being a direction further away from other second virtual objects, and i being a positive integer less than or equal to n; and~~

~~moving a logical position of the i^{th} second virtual object along the offset direction by the offset distance, to obtain an offset logical position of the i^{th} second virtual object.~~

~~4~~2. The method according to claim ~~3~~1, wherein the determining the offset direction of the i^{th} second virtual object comprises:

determining repulsive virtual objects corresponding to the i^{th} second virtual object, the repulsive virtual objects comprising other second virtual objects of which distances from the i^{th} second virtual object are less than ~~a~~ the first distance in the plurality of second virtual objects;

obtaining repulsive forces of the repulsive virtual objects to the i^{th} second virtual object, a magnitude of the repulsive force being associated with attribute information of the repulsive virtual object, and a direction of the repulsive force being a direction from the repulsive virtual object to the i^{th} second virtual object, the attribute information including at least one of the following: a distance between the repulsive virtual object and the i^{th} second virtual object, a category to which the repulsive virtual object belongs, a life value of the repulsive virtual object, an energy value of the repulsive virtual object; and

determining a direction of a resultant force of repulsive forces of the repulsive virtual objects to the i^{th} second virtual object as the offset direction of the i^{th} second virtual object.

~~5~~3. The method according to claim ~~3~~1, wherein the determining the offset direction of the i^{th} second virtual object comprises:

obtaining a center position of the execution range; and

determining a direction from the center position to the i^{th} second virtual object as the offset direction of the i^{th} second virtual object.

~~6~~4. The method according to claim ~~3~~1, wherein the determining the offset direction of the i^{th} second virtual object comprises:

obtaining a central axis of the execution range; and

determining a direction opposite to a direction of the i^{th} second virtual object relative to the central axis as the offset direction of the i^{th} second virtual object.

~~7~~5. The method according to claim ~~3~~1, wherein the determining the offset distance of the i^{th} second virtual object comprises:

obtaining a category to which the i^{th} second virtual object belongs; and

determining the offset distance of the i^{th} second virtual object according to the category to which the i^{th} second virtual object belongs.

~~8~~6. The method according to claim ~~3~~1, further comprising, after the moving the logical position of the i^{th} second virtual object along the offset direction by the offset distance:

obtaining a distance between the offset logical position of the i^{th} second virtual object and the first virtual object;

obtaining, based on the distance being greater than a second distance, an offset angle, which is an angle formed by logical positions of the i^{th} second virtual object, before and after an offset, relative to the first virtual object;

performing, based on the offset angle being less than an angle threshold, operations again from the operations of determining an offset direction of an i^{th} second virtual object and determining an offset distance of the i^{th} second virtual object, an offset direction and an offset distance of the offset logical position of the i^{th} second virtual object being determined when the operations are performed again; and

obtaining the offset logical position of the i^{th} second virtual object based on the offset angle being greater than or equal to the angle threshold.

~~9~~7. The method according to claim 1, wherein there are n second virtual objects, n being an integer greater than 1; and

after the obtaining the execution range of the first operation of the first virtual object in the virtual environment, the method further comprises:

determining distances between every two second virtual objects in the n second virtual objects; and

based on that a distance that is less than ~~a~~ the first distance being included in the distances, determining that the density satisfies the first condition.

~~10~~8. The method according to claim ~~9~~7, further comprising

obtaining an operation effect corresponding to the first operation, the operation effect being an effect produced by the first operation; and

determining the first distance according to the operation effect.

~~11~~9. The method according to any one of claims 1 to ~~10~~8, wherein the selecting a target virtual object corresponding to the first operation from the second virtual objects based on the offset logical position of the at least one second virtual object comprises:

obtaining a center position of the execution range of the first operation;

determining distances between offset logical positions of the second virtual objects and the center position; and

determining a second virtual object of which the distance satisfies a second condition as the target virtual object.

~~12~~10. A virtual object selection apparatus, comprising:

a user interface display module, configured to display a user interface, the user interface comprising an image corresponding to a virtual environment and a first virtual object located in the virtual environment;

an execution range obtaining module, configured to obtain an execution range of a first operation of the first virtual object in the virtual environment;

a logical position offset module, configured to offset, based on a density of a plurality of second virtual objects in the execution range satisfying a first condition, a logical position of at least one second virtual object in the plurality of second virtual objects, to obtain an offset logical position of the at least one second virtual object, the logical position being an initial position used in processing the second virtual object in a logical layer of a client of the terminal; wherein an interface position of the second virtual object does not change according to a change of the logical position, the interface position of the second virtual object being a position used when a representation layer of the client processes the second virtual object; wherein a distance between the offset logical positions of every two second virtual objects in the plurality of second virtual objects is greater than or equal to a first distance; and

a target object selection module, configured to select a target virtual object, on which corresponding to the first operation is performed, from the plurality of second virtual objects based on the offset logical position of the at least one second virtual object;

wherein there are n second virtual objects, n being an integer greater than 1, and

the logical position offset module, configured to:

determine an offset direction of an i^{th} second virtual object in the n second virtual objects and determine an offset distance of the i^{th} second virtual object, the offset direction being a direction further away from other second virtual objects, and i being a positive integer less than or equal to n; and move a logical position of the i^{th} second virtual object along the offset direction by the offset distance, to obtain an offset logical position of the i^{th} second virtual object.

~~13~~11. A terminal, comprising a processor and a memory, the memory storing at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction set being loaded and executed by the processor to implement the virtual object selection method according to any one of claims 1 to ~~11~~9.

~~14~~12. A computer-readable storage medium, storing at least one instruction, at least one program, a code set, or an instruction set, the at least one instruction, the at least one program, the code set, or the instruction set being loaded and executed by a processor to implement the virtual object selection method according to any one of claims 1 to ~~11~~9.