CLAIMS

1. A method of controlling game states, applied to a game of cards performed on a desktop, comprising:

determining, in response to that an interface call instruction is acquired, a target interface from a plurality of preset interfaces according to the interface call instruction, wherein each of the plurality of interfaces is encapsulated with a state switching condition, and before acquiring the interface call instruction, game states are switched automatically;

acquiring a desktop image of the game in a current state of the game;

detecting state information of one or more desktop elements of the game according to the desktop image;

transferring the state information to the target interface, and

switching the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information.

2. The method according to claim 1, wherein

the desktop elements comprise:

a marker placement region for placing a preset marker before the game is started, and

an item operation region for operating a game item if the game is in an item operation state;

determining the target interface from the plurality of preset interfaces according to the interface call instruction comprises:

determining, if the interface call instruction is used for instructing to invalidate a current round of the game and start a next round of the game, that the target interface is an invalidation interface; and

switching the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information comprises:

determining, if the target interface is the invalidation interface and the current state is a paused state, whether the preset marker exists in the marker placement region and whether the item operation region is in an empty state according to the state information; and

switching, if the preset marker exists in the marker placement region and the item operation region is in the empty state, the game to a prop operation state.

3. The method according to claim 1, wherein

the desktop elements comprise:

a game item,

a marker placement region for placing a preset marker before the game is started, and

a game item operation region for operating the game item if the game is in an item operation state;

determining the target interface from the plurality of preset interfaces according to the interface call instruction comprises:

determining, if the interface call instruction is used for instructing to restore a prop operation state, an item operation state, or a result processing state of a current round of the game, that the target interface is a restoration interface; and

switching the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information comprises:

determining, if the target interface is the restoration interface and the current state is a paused state, whether the preset marker exists in the marker placement region and whether the item operation region is in an empty state according to the state information;

switching, if the preset marker exists in the marker placement region and the item operation region is in the empty state, the game to the prop operation state; and

switching, if the preset marker exists in the marker placement region and the game item exists in the item operation region, the game to the item operation state or the result processing state according to state information of the game item in the item operation region.

4. The method according to claim 3, wherein the state information of the game item comprises a quantity of the game items; and switching the game to the item operation state or the result processing state according to state information of the game item in the item operation region comprises:

switching, if the quantity of the game items in the item operation region is less than a preset quantity threshold, the game to the item operation state.

5. The method according to claim 4, wherein the state information of the game item further comprises state information of an identification surface of the game item; and switching the game to the item operation state or the result processing state according to state information of the game item in the item operation region further comprises:

determining, if the quantity of the game items is greater than or equal to the preset quantity threshold, whether the identification surface of each of the game items in the item

operation region is a visible state or a hidden state according to the state information of the identification surface of each of the game items in the item operation region;

if the identification surface of each of the game items is in the visible state,

acquiring the identifier information of each of the game items, and

switching, if a game result is generated according to the identifier information, the game to the result processing state; and

switching, if the identification surface of at least one of the game items is in the hidden state, the game to the item operation state.

6. The method according to any one of claims 3 to 5, wherein

the item operation region comprises at least two item subregions;

determining, according to the state information, whether the item operation region is in an empty state comprises:

determining, according to the state information, whether each of the item subregions of the item operation region is in the empty state; and

switching the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information further comprises:

keeping, if a relationship between quantities of items in the item subregions of the item operation region does not satisfy a preset relationship, the game in the paused state.

7. The method according to claim 1, wherein determining the target interface from the plurality of preset interfaces according to the interface call instruction comprises:

if the interface call instruction is used for instructing to end the game, determining that the target interface is an idle interface;

if the interface call instruction is used for instructing to start a round of game, determining that the target interface is a prop interface;

if the interface call instruction is used for instructing to pause the game, determining that the target interface is a pause interface; and

if the interface call instruction is used for instructing to reset the game, determining that the target interface is a resetting interface.

8. The method according to claim 7, wherein switching the game to a target state according to a state switching condition encapsulated in the target interface and corresponding to the current state by calling the target interface comprises:

if the target interface is the idle interface and the current state is a prop operation state, switching the game to an idle state;

if the target interface is the prop interface and the current state is the idle state, switching the game to the prop operation state;

if the target interface is the pause interface and the current state is the prop operation state or an item operation state, switching the game to a paused state;

if the target interface is the resetting interface and the current state is in any state, switching the game to the idle state; and

if the target interface is any interface and the current state is in a blank state, switching the game to the idle state.

9. An apparatus for controlling game states, applied to a game of cards performed on a desktop, comprising:

a target module, configured to: determine, in response to that an interface call instruction is acquired, a target interface from a plurality of preset interfaces according to

the interface call instruction, wherein each of the plurality of interfaces is encapsulated with a state switching condition, and before acquiring the interface call instruction, game states are switched automatically;

an image module, configured to acquire a desktop image of the game in a current state of the game;

a detection module, configured to detect state information of one or more desktop elements of the game according to the desktop image;

a transfer module, configured to transfer the state information to the target interface; and

a switching module, configured to: switch the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information.

10. The apparatus according to claim 9, wherein the desktop elements comprise a marker placement region for placing a preset marker before the game is started, and an item operation region for operating a game item if the game is in an item operation state;

when the target module is configured to determine the target interface from the plurality of preset interfaces according to the interface call instruction, the target module is specifically configured to:

if the interface call instruction is used for instructing to invalidate a current round of the game and start a next round of the game, determine that the target interface is an invalidation interface; and

when the switching module is configured to switch the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the

current state together with the state information, the switching module is specifically configured to:

determine, if the target interface is the invalidation interface and the current state is a paused state, whether the preset marker exists in the marker placement region and whether the item operation region is in an empty state according to the state information; and

switch, if the preset marker exists in the marker placement region and the item operation region is in the empty state, the game to a prop operation state.

11. The apparatus according to claim 9, wherein

the desktop elements comprise:

a game item,

a marker placement region for placing a preset marker before the game is started, and

a game item operation region for operating the game item if the game is in an item operation state;

when the target module is configured to determine the target interface from the plurality of preset interfaces, the target module is specifically configured to:

determine, if the interface call instruction is used for instructing to restore a prop operation state, an item operation state, or a result processing state of a current round of the game, that the target interface is a restoration interface; and

when the switching module is configured to switch the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information, the switching module is specifically configured to:

determine, if the target interface is the restoration interface and the current state is a paused state, whether the preset marker exists in the marker placement region and whether the item operation region is in an empty state according to the state information;

switch, if the preset marker exists in the marker placement region and the item operation region is in the empty state, the game to the prop operation state; and

switch, if the preset marker exists in the marker placement region and the game item exists in the item operation region, the game to the item operation state or the result processing state according to state information of the game item in the item operation region.

12. The apparatus according to claim 11, wherein the state information of the game item comprises a quantity of the game items; and

when the switching module is configured to switch the game to the item operation state or the result processing state according to state information of the game item in the item operation region, the switching module is specifically configured to:

switch, if the quantity of the game items in the item operation region is less than a preset quantity threshold, the game to the item operation state.

13. The apparatus according to claim 12, wherein the state information of the game item further comprises state information of an identification surface of the game item; and

when the switching module is configured to switch the game to the item operation state or the result processing state according to state information of the game item in the item operation region, the switching module is specifically configured to:

determine, if the quantity of the game items is greater than or equal to the preset quantity threshold, whether the identification surface of each of plurality of game items in the item

operation region is a visible state or a hidden state according to the state information of the identification surface of each of the game items in the item operation region;

if the identification surface of each of the game items is in the visible state,

acquire the identifier information of each of the game items, and

switch, if a game result is generated according to the identifier information, the game to the result processing state; and

switch, if the identification surface of at least one of the game items is in the hidden state, the game to the item operation state.

14. The apparatus according to any one of claims 11 to 13, wherein the item operation region comprises at least two item subregions;

when the switching module is configured to determine, according to the state information, whether the item operation region is in an empty state, the switching module is specifically configured to:

determine, according to the state information, whether each of the item subregions of the item operation region is in the empty state; and

when the switching module is configured to switch the game to a target state according to the state switching condition encapsulated in the target interface and corresponding to the current state together with the state information, the switching module is further specifically configured to:

keeping, if a relationship between quantities of items in the item subregions of the item operation region does not satisfy a preset relationship, the game in the paused state.

15. The apparatus according to claim 9, wherein when the target module is configured to determine the target interface from the plurality of preset interfaces according to the interface call instruction, the target module is specifically configured to:

if the interface call instruction is used for instructing to end the game, determine that the target interface is an idle interface;

if the interface call instruction is used for instructing to start a round of game, determine that the target interface is a prop interface;

if the interface call instruction is used for instructing to pause the game, determine that the target interface is a pause interface; and

if the interface call instruction is used for instructing to reset the game, determine that the target interface is a resetting interface.

16. The apparatus according to claim 15, wherein when the switching module is configured to switch the game to a target state according to a state switching condition encapsulated in the target interface and corresponding to the current state by calling the target interface, the switching module is specifically configured to:

if the target interface is the idle interface and the current state is a prop operation state, switch the game to an idle state;

if the target interface is the prop interface and the current state is the idle state, switch the game to the prop operation state;

if the target interface is the pause interface and the current state is the prop operation state or an item operation state, switch the game to a paused state;

if the target interface is the resetting interface and the current state is in any state, switch the game to the idle state; and

if the target interface is any interface and the current state is in a blank state, switch the game to the idle state.

17. An electronic device, comprising a memory and a processor, wherein the memory is configured to store computer instructions capable of being run on the processor, and the processor implements the method according to any one of claims 1 to 8 when executing the computer instructions.

18. A computer-readable storage medium, storing a computer program, wherein the method according to any one of claims 1 to 8 is performed when the program is executed by a processor.