



US 20140096087A1

(19) **United States**  
(12) **Patent Application Publication**  
**Luan et al.**

(10) **Pub. No.: US 2014/0096087 A1**  
(43) **Pub. Date: Apr. 3, 2014**

(54) **METHOD AND DEVICE FOR SOFTWARE INTERFACE DISPLAY ON TERMINAL, AND COMPUTER STORAGE MEDIUM**

**Publication Classification**

(71) Applicant: **Tencent Technology (Shenzhen) Company Limited**, Shenzhen (CN)

(51) **Int. Cl.**  
**G06F 3/0481** (2006.01)  
(52) **U.S. Cl.**  
CPC ..... **G06F 3/04815** (2013.01)  
USPC ..... **715/848**

(72) Inventors: **Lan Luan**, Shenzhen (CN); **Zhan Chen**, Shenzhen (CN)

(57) **ABSTRACT**

(21) Appl. No.: **14/099,147**

A method and device for software interface display on a terminal is described. The method includes that: a terminal acquires an interface dragging instruction, and records a dragging distance corresponding to the interface dragging instruction; and performs coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface according to a result of the coordinate transformation, achieving the 3D effect or simulated 3D visual effect. With the present method, it is possible to implement software interface display in 3D on a terminal.

(22) Filed: **Dec. 6, 2013**

**Related U.S. Application Data**

(63) Continuation of application No. PCT/CN2012/086073, filed on Dec. 6, 2012.

(30) **Foreign Application Priority Data**

Dec. 19, 2011 (CN) ..... 201110427317X

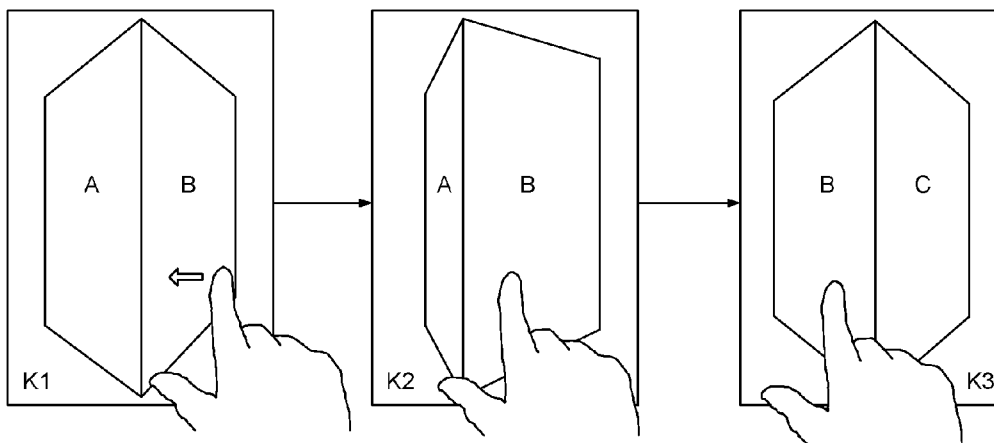


Fig. 1

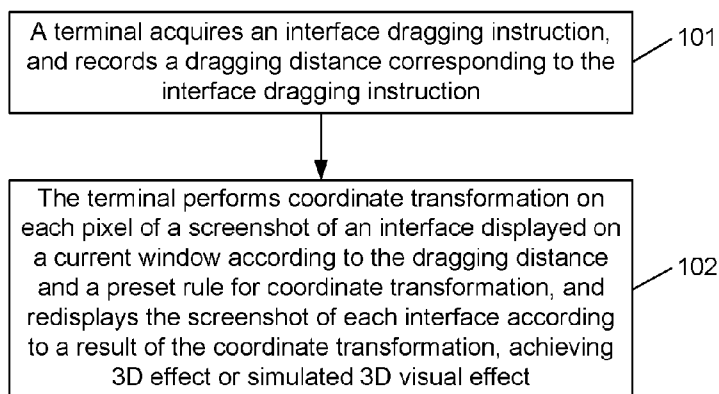


Fig. 2

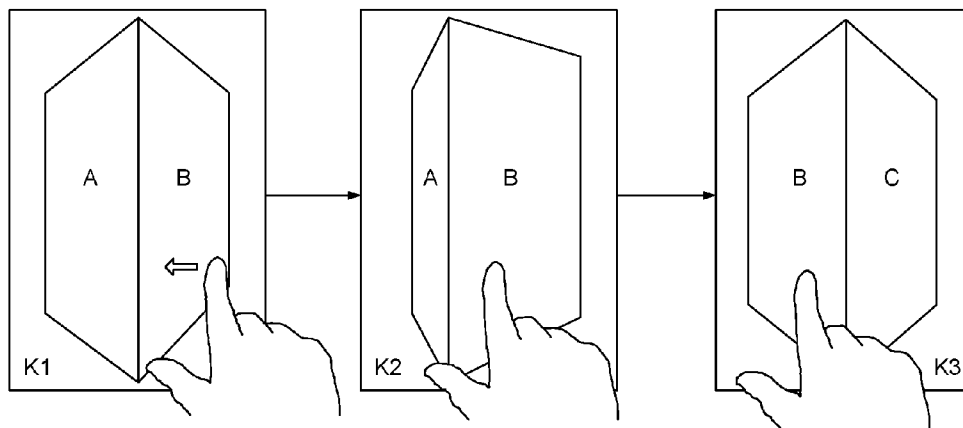


Fig. 3

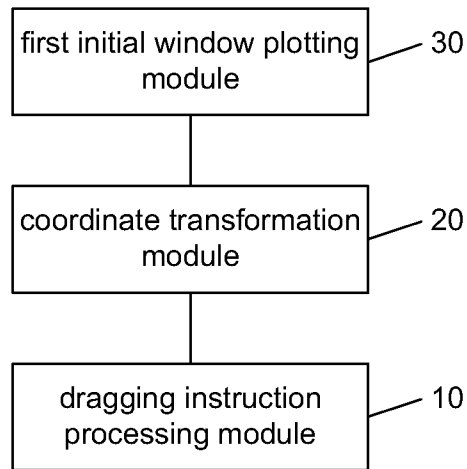


Fig. 4

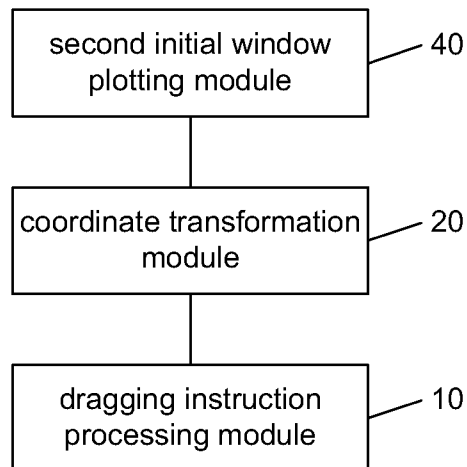
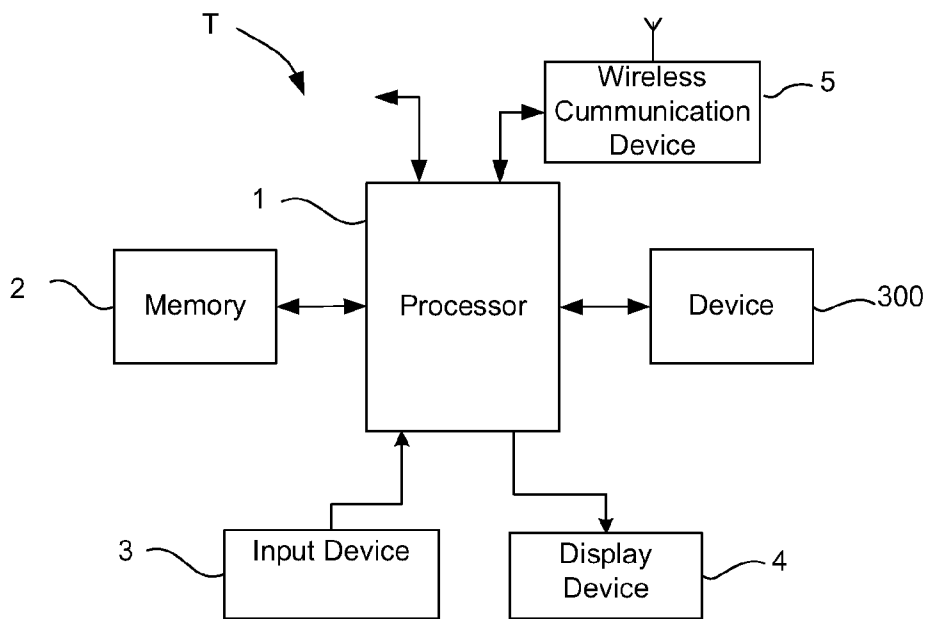


Fig. 5



**METHOD AND DEVICE FOR SOFTWARE INTERFACE DISPLAY ON TERMINAL, AND COMPUTER STORAGE MEDIUM**

**CROSS-REFERENCE TO RELATED APPLICATIONS**

[0001] This is a continuation application of International Patent Application No. PCT/CN2012/086073, filed on Dec. 6, 2012, which claims priority to Chinese Patent Application No. 201110427317.X filed on Dec. 19, 2011, the disclosure of which is incorporated by reference herein in its entirety.

**TECHNICAL FIELD**

[0002] The present disclosure generally relates to the field of software interface display on a terminal, and in particular to a method and device for software interface display on a terminal and a non-transitory computer-readable storage medium.

**BACKGROUND**

[0003] Most software interfaces at present are displayed in a 1D (one-dimensional) or 2D (two-dimensional) mode. In the case of 1D display, when a certain function is opened, the new interface is displayed by overlapping another layer; with such a displaying mode, just one interface (function) can be opened at one time, which increases the cost for function switching. In the case of 2D display, when a user triggers a certain hotspot, content of the background is displayed by a flip.

[0004] Both the 1D display mode and the 2D display mode are limited to display a software interface with a simple information structure, which is disadvantageous for software function expansion.

**SUMMARY**

[0005] The embodiments of the present disclosure provide a method and device for software interface display on a terminal and a non-transitory computer-readable storage medium, so as to implement software interface display in 3D on a terminal.

[0006] According to an aspect of the present disclosure, a method for software interface display on a terminal is provided, including:

[0007] a terminal acquires an interface dragging instruction, and records a dragging distance corresponding to the interface dragging instruction; and

[0008] the terminal performs coordinate transformation on each pixel of a screenshot of an interface displayed on a current window of the terminal according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface according to a result of the coordinate transformation, so as to achieving 3D effect or simulated 3D visual effect of the display.

[0009] According to another aspect of the present disclosure, a device for software interface display on a terminal is provided, and includes a dragging instruction processing module and a coordinate transformation module,

[0010] wherein the dragging instruction processing module is configured to acquire an interface dragging instruction, and record a dragging distance corresponding to the interface dragging instruction; and

[0011] wherein the coordinate transformation module is configured to perform coordinate transformation on each

pixel of a screenshot of an interface displayed on a current window of the terminal according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface according to a result of the coordinate transformation, so as to achieve 3D effect or simulated 3D visual effect of the display.

[0012] The present disclosure further provides a computer readable storage medium in which computer-executable instructions are stored, the computer-executable instructions being used for executing the method for software interface display on a terminal according to the disclosure.

[0013] With the method and device for software interface display on a terminal and the computer readable storage medium provided by the embodiments of the present disclosure, an interface dragging instruction is acquired first, and a dragging distance corresponding to the interface dragging instruction is recorded; then, coordinate transformation is performed on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and the screenshot of each interface is redisplayed according to a result of the coordinate transformation, so that 3D effect or simulated 3D visual effect may be archived. With the embodiments of the present disclosure, expansion is made on a software structure, thus realizing software interface display in 3D on a terminal, which not only meets a demand for richer terminal functionality, but also enhances users' operational experience.

**BRIEF DESCRIPTION OF THE DRAWINGS**

[0014] FIG. 1 is a flowchart of a method for software interface display on a terminal according to an embodiment of the present disclosure.

[0015] FIG. 2 is a schematic diagram of 3D effect in software interface display on a terminal in an embodiment of the present disclosure.

[0016] FIG. 3 is a schematic diagram of a structure of a device for software interface display on a terminal according to an embodiment of the present disclosure.

[0017] FIG. 4 is a schematic diagram of a structure of another device for software interface display on a terminal according to an embodiment of the present disclosure.

[0018] FIG. 5 is a schematic diagram of a structure of a terminal according to an embodiment of the present disclosure.

**DETAILED DESCRIPTION**

[0019] A technical solution of the present disclosure is further elaborated below with reference to the drawings and specific embodiments.

[0020] FIG. 1 shows a method for software interface display on a terminal provided by an embodiment of the present disclosure, mainly includes the follows steps.

[0021] In step 101, a terminal acquires an interface dragging instruction, and records a dragging distance corresponding to the interface dragging instruction.

[0022] Once a user executes a dragging operation on the screen of a touch-screen terminal, or clicks on a current display interface through a mouse and executes a mouse dragging operation on a non-touch-screen terminal, the interface dragging instruction will be triggered, with which the

terminal acquires the interface dragging instruction, and records a dragging distance corresponding to the interface dragging instruction.

**[0023]** In step 102, the terminal performs coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface according to a result of the coordinate transformation, achieving 3D effect or simulated 3D visual effect.

**[0024]** the rule for coordinate transformation is for example preset based on dragging distances within the terminal. After recording the dragging distance, the terminal performs coordinate transformation on each pixel of the screenshot of the interface displayed on the current window according to the rule for coordinate transformation, and redisplay the screenshot of each interface according to the result of the coordinate transformation.

**[0025]** As examples, there are two modes for implementing 3D display of software interface on a terminal.

**[0026]** The first mode of implementation may be called pseudo-3D. when a window is initially displayed (before the terminal acquires the interface dragging instruction), the terminal acquires a screenshot of each interface which needs to be displayed on a current window; and by performing 2D abscissa and ordinate calculation on each element of the screenshot of each interface under 2D coordinates of the current window, the terminal stretches or enlarges the screenshot of each interface according to a certain angle, and plots the stretched or enlarged screenshot on the current window, so as to achieve the simulated 3D visual effect. Accordingly, when obtaining the interface dragging instruction, the terminal records the corresponding dragging distance, calculates new 2D coordinates of each pixel of the screenshot of each interface according to a preset scaling factor (which represents a scaling relation between the dragging distance and the 2D abscissa and ordinate under the current window) and the dragging distance, and re-plots the screenshot of each interface according to the calculated new 2D coordinates.

**[0027]** The second mode of implementation may be called real-3D. When a window is initially displayed (before the terminal acquires the interface dragging instruction), the terminal acquires a screenshot of each interface which needs to be displayed on a current window; constructs (or renders) a cube under 3D coordinates (spatial coordinates on axes X, Y, Z) of the current window, and attaches the screenshot of each interface to a corresponding surface of the constructed cube as a texture image, so as to display 3D effect. Accordingly, when receiving the interface dragging instruction, the terminal records the corresponding dragging distance, performs spatial coordinate transformation on the cube according to the dragging distance, and redisplay a corresponding screenshot of an interface on each surface of the cube according to a result of the spatial coordinate transformation on the cube. The real-3D display may be implemented through a graphics library, such as an Open Graphics Library for Embedded Systems (OpenGL ES), which is a cross-platform, mobile-platform-based 3D graphic application tool with sophisticated functions. In some examples, the real-3D display within the terminal may be implemented by calling an Application Program Interface (API) of the OpenGL ES.

**[0028]** The terminal renders a cube under 3D coordinates of the current window by calling an API of the OpenGL ES, acquires a screenshot of each interface to be displayed on the

current window, and attaches the screenshot of each interface to a corresponding surface of the rendered cube as a texture image to achieve 3D display effect. The subsequent spatial coordinate transformation on the cube is implemented by the terminal by calling an API of the OpenGL ES as well. When initially displayed, the screenshot of each interface is attached to a corresponding surface of the cube as a texture image. Accordingly, when the spatial coordinates of the cube are transformed, the change corresponding to the change in spatial coordinates of the surface of the cube will occur in spatial coordinates of the screenshot of an interface attached to that surface, and the screenshot of an interface will show a visual 3D effect.

**[0029]** The method for software interface display on a terminal according to an embodiment of the present disclosure is further elaborated below with reference to 3D effect in software interface display on a terminal as shown in FIG. 2.

**[0030]** The pseudo-3D display is introduced first. When a window is initially displayed, the terminal acquires a screenshot of each interface to be displayed on a current window (such as screenshot A and screenshot B shown in FIG. 2). Under 2D coordinates of the current window, by performing 2D abscissa and ordinate calculation on each element of the screenshot (screenshot A and screenshot B) of each interface, the terminal stretches or enlarges the screenshot of each interface according to a certain angle, and plots the stretched or enlarged screenshot on the current window, so as to achieve the simulated 3D visual effect, with the displayed effect as shown in window K1 of FIG. 2. When receiving the interface dragging instruction (dragging toward the left as shown in FIG. 2), the terminal records a corresponding dragging distance, calculates new 2D coordinates of each pixel of the screenshot of each interface according to a preset scaling factor and the dragging distance, and re-plots the screenshot of each interface according to the new calculated 2D coordinates, with the displayed effect as shown in window K2 of FIG. 2.

**[0031]** Turning to the real-3D display. When a window is initially displayed, the terminal renders a cube under 3D coordinates of a current window by calling an API of the OpenGL ES, the terminal acquires a screenshot of each interface to be displayed on the current window (such as screenshot A, screenshot B, and screenshot C), and attaches the screenshot of each interface to a corresponding surface of the constructed cube as a texture image, so as to display 3D effect. When receiving the interface dragging instruction (dragging toward the left as shown in FIG. 2), the terminal records a corresponding dragging distance, performs spatial coordinate transformation on the cube according to the dragging distance, and redisplay a corresponding screenshot of an interface on each surface of the cube according to a result of the spatial coordinate transformation on the cube. When initially displayed, the screenshot of each interface is attached to a corresponding surface of the cube as a texture image. Accordingly, when the spatial coordinates of the cube are transformed, the change corresponding to the change in spatial coordinates of the surface of the cube will occur in spatial coordinates of the screenshot of an interface attached to that surface, and the screenshot of an interface will show a visual 3D effect, with the displayed effect as shown in window K2, K3 of FIG. 2.

**[0032]** An embodiment of the present disclosure provides a device for software interface display on a terminal.

**[0033]** A device for implementing pseudo-3D display, as shown in FIG. 3, includes a dragging instruction processing module 10, a coordinate transformation module 20, and a first initial window plotting module 30.

**[0034]** The first initial window plotting module 30 is configured to, when the window is initially displayed, acquire a screenshot of each interface to be displayed on the current window; and under 2D coordinates of the current window, stretch or enlarge the screenshot of each interface according to a certain angle through abscissa and ordinate calculation, and plot the stretched or enlarged screenshot on the current window to simulate 3D visual effect.

**[0035]** The dragging instruction processing module 10 is configured to acquire an interface dragging instruction, and record a dragging distance corresponding to the interface dragging instruction.

**[0036]** The coordinate transformation module 20 is configured to perform coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface according to a result of the coordinate transformation. Specifically, the coordinate transformation module 20 is configured to calculate new 2D coordinates of each pixel of the screenshot of each interface according to a preset scaling factor and the dragging distance, and re-plot the screenshot of each interface according to the calculated new 2D coordinates. The coordinate transformation module 20 is connected to the first initial window plotting module 30. When performing 2D coordinate transformation, the coordinate transformation module 20 acquires, from the first initial window plotting module 30, 2D coordinates of the screenshot of each interface at the time the window is initially displayed, and performs 2D coordinate transformation based on the 2D coordinates of the initially displayed screenshot.

**[0037]** A device for implementing real-3D display, as shown in FIG. 4, includes a dragging instruction processing module 10, a coordinate transformation module 20, and a second initial window plotting module 40.

**[0038]** The second initial window plotting module 40 is configured to, when the window is initially displayed, acquire a screenshot of each interface to be displayed on the current window; construct a cube under 3D coordinates of the current window, and attach the screenshot of each interface to a corresponding surface of the constructed cube as a texture image so as to achieve 3D display effect.

**[0039]** The dragging instruction processing module 10 is configured to acquire an interface dragging instruction, and record a dragging distance corresponding to the interface dragging instruction

**[0040]** The coordinate transformation module 20 is configured to perform coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface according to a result of the coordinate transformation. Specifically, the coordinate transformation module 20 is configured to perform spatial coordinate transformation on the cube according to the dragging distance, and redisplay a corresponding screenshot of an interface on each surface of the cube according to a result of the spatial coordinate transformation on the cube. The coordinate transformation module 20 is connected to the second initial window plotting module 40. When performing spatial coordinate transformation, the

coordinate transformation module 20 acquires from the second initial window plotting module 40, spatial coordinates of the cube at the time the window is initially displayed, and performs spatial coordinate transformation based on the spatial coordinates of the initially displayed cube.

**[0041]** FIG. 5 illustrates a structure of a terminal according to an embodiment of the present disclosure. The terminal is configured to implement methods for software interface display on the terminal according to various embodiments of the present disclosure.

**[0042]** As shown in FIG. 5, the terminal T may include, but not limited to the following components: a processor 1 including one or more processing cores, a memory 2 including one or more non-transitory computer readable storage mediums, an input device 3, a display device 4, a wireless communication device 5, and the a device 300 for software interface display on the terminal according to the above embodiments. For example, the device 300 may be a device as shown in FIG. 3 or a device as shown in FIG. 4.

**[0043]** Those skilled in the art should understand that the above structure of the terminal shown in FIG. 5 does not constitute a limit to the terminal. The terminal can include more or less than those shown in FIG. 5, or be a combination of certain components, or be a different layout of the components.

**[0044]** When implemented in form of a software functional module and sold or used as an independent product, an integrated module of an embodiment of the present disclosure may also be stored in a computer-readable storage medium. Based on such an understanding, the essential part or a part of the technical solution of an embodiment of the present disclosure may appear in form of a software product, which software product is stored in storage media, and includes a number of instructions for allowing a computer equipment (such as a personal computer, a server, a network equipment, or the like) to execute all or part of the methods in various embodiments of the present disclosure. The storage media include various media that can store program codes such as a U disk, a mobile hard disk, a Read-Only Memory (ROM), a Random Access Memory (RAM), a magnetic disk, a CD, and the like. Thus, an embodiment of the present disclosure is not limited to any specific combination of hardware and software.

**[0045]** Accordingly, an embodiment of the present disclosure further provides a computer storage medium in which a computer program is stored, which computer program is for executing the method for software interface display on a terminal according to an embodiment of the present disclosure.

**[0046]** What described are merely various embodiments of the disclosure, and are not intended to limit the scope of the disclosure.

1. A method for software interface display on a terminal, comprising:

- acquiring, by a terminal, an interface dragging instruction, and recording, by the terminal, a dragging distance corresponding to the interface dragging instruction; and
- performing, by the terminal, coordinate transformation on each pixel of a screenshot of an interface displayed on a current window of the terminal according to the dragging distance and a preset rule for coordinate transformation, and redisplaying, by the terminal, the screenshot of each interface in 3D effect or in simulated 3D visual effect according to a result of the coordinate transformation.

2. The method according to claim 1, further comprising: before the acquiring, by a terminal, an interface dragging instruction,

acquiring, by the terminal, a screenshot of each interface to be displayed on the current window; and under 2D coordinates of the current window, stretching or enlarging, by the terminal, the screenshot of each interface according to a certain angle through abscissa and ordinate calculation, and plotting, by the terminal, the stretched or enlarged screenshot on the current window to achieve the simulated 3D visual effect.

3. The method according to claim 2, wherein the performing, by the terminal, coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplaying, by the terminal, the screenshot of each interface according to a result of the coordinate transformation, comprises:

calculating, by the terminal, new 2D coordinates of each pixel of the screenshot of each interface according to a preset scaling factor and the dragging distance, and re-plotting, by the terminal, the screenshot of each interface according to the calculated new 2D coordinates.

4. The method according to claim 1, further comprising: before the acquiring, by a terminal, an interface dragging instruction,

acquiring, by the terminal, a screenshot of each interface to be displayed on the current window; constructing, by the terminal, a cube under 3D coordinates of the current window, and attaching, by the terminal, the screenshot of each interface to a corresponding surface of the constructed cube as a texture image, and displaying the screenshot of each interface in 3D effect.

5. The method according to claim 4, wherein the performing, by the terminal, coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplaying, by the terminal, the screenshot of each interface according to a result of the coordinate transformation, comprises:

performing, by the terminal, spatial coordinate transformation on the cube according to the dragging distance, and redisplaying, by the terminal, a corresponding screenshot of an interface on each surface of the cube according to a result of the spatial coordinate transformation on the cube.

6. A device for software interface display on a terminal, comprising:

a dragging instruction processing module configured to acquire an interface dragging instruction, and record a dragging distance corresponding to the interface dragging instruction; and

a coordinate transformation module configured to perform coordinate transformation on each pixel of a screenshot of an interface displayed on a current window of the terminal according to the dragging distance and a preset rule for coordinate transformation, and redisplay the screenshot of each interface in 3D effect or in simulated 3D visual effect according to a result of the coordinate transformation.

7. The device according to claim 6, further comprising a first initial window plotting module configured to, before the dragging instruction processing module acquires the interface dragging instruction, acquire a screenshot of each inter-

face to be displayed on the current window; and under 2D coordinates of the current window, stretch or enlarge the screenshot of each interface according to a certain angle through abscissa and ordinate calculation, and plot the stretched or enlarged screenshot on the current window to achieve the simulated 3D visual effect.

8. The device according to claim 7, wherein the coordinate transformation module is configured to calculate new 2D coordinates of each pixel of the screenshot of each interface according to a preset scaling factor and the dragging distance, and re-plot the screenshot of each interface according to the calculated new 2D coordinates.

9. The device according to claim 6, further comprising a second initial window plotting module configured to, before the dragging instruction processing module acquires the interface dragging instruction, acquire a screenshot of each interface to be displayed on the current window; construct a cube under 3D coordinates of the current window, and attach the screenshot of each interface to a corresponding surface of the constructed cube as a texture image; and display the screenshot of each interface in 3D effect.

10. The device according to claim 9, wherein the coordinate transformation module is configured to perform spatial coordinate transformation on the cube according to the dragging distance, and redisplay a corresponding screenshot of an interface on each surface of the cube according to a result of the spatial coordinate transformation on the cube.

11. A non-transitory computer-readable storage medium storing instructions thereon for execution by at least one processing circuit for software interface display on a terminal, the instructions comprising:

acquiring an interface dragging instruction, and recording a dragging distance corresponding to the interface dragging instruction; and

performing coordinate transformation on each pixel of a screenshot of an interface displayed on a current window of the terminal according to the dragging distance and a preset rule for coordinate transformation, and redisplaying the screenshot of each interface in 3D effect or in simulated 3D visual effect according to a result of the coordinate transformation.

12. The non-transitory computer-readable storage medium according to claim 11, wherein the instructions further comprises: before the acquiring an interface dragging instruction, acquiring a screenshot of each interface to be displayed on the current window;

and under 2D coordinates of the current window, stretching or enlarging the screenshot of each interface according to a certain angle through abscissa and ordinate calculation, and plotting the stretched or enlarged screenshot on the current window to achieve the simulated 3D visual effect.

13. The non-transitory computer-readable storage medium according to claim 12, wherein the performing coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplaying the screenshot of each interface according to a result of the coordinate transformation, comprises:

calculating new 2D coordinates of each pixel of the screenshot of each interface according to a preset scaling factor and the dragging distance, and re-plotting the screenshot of each interface according to the calculated new 2D coordinates.



**14.** The non-transitory computer-readable storage medium according to claim **11**, wherein the instructions further comprises: before the acquiring an interface dragging instruction, acquiring a screenshot of each interface to be displayed on the current window;

constructing a cube under 3D coordinates of the current window, and attaching the screenshot of each interface to a corresponding surface of the constructed cube as a texture image, and displaying the screenshot of each interface in 3D effect.

**15.** The non-transitory computer-readable storage medium according to claim **14**, wherein the performing coordinate transformation on each pixel of a screenshot of an interface displayed on a current window according to the dragging distance and a preset rule for coordinate transformation, and redisplaying the screenshot of each interface according to a result of the coordinate transformation, comprises:

performing spatial coordinate transformation on the cube according to the dragging distance, and redisplaying a corresponding screenshot of an interface on each surface of the cube according to a result of the spatial coordinate transformation on the cube.

\* \* \* \* \*