

METHOD AND APPARATUS FOR REALIZING COLOR TWEEN ANIMATION

TECHNICAL FIELD

[0001] The present application relates to the field of image processing
5 technologies, and in particular, to a method and an apparatus for realizing color tween
animation.

BACKGROUND

[0002] Animation includes frame-by-frame animation and tween animation. The
frame-by-frame animation is to draw different content frame-by-frame based on the
10 timeline and form an animation by consecutively playing these frames. The tween
animation is to automatically generate an intermediate part of an animation by using
software based on a start frame and an end frame of the known animation, which
omits a complex process for making the intermediate part.

[0003] Color tween animation is a type of tween animation. The color tween
15 animation means that a fill color in a certain area changes with time to create
animation effects. The color tween animation can be used alone or in combination
with other tween animation (for example, shape tween animation, displacement tween
animation, scaling tween animation, or rotation tween animation in a fill area) to
create more complex animation effects.

20 [0004] The existing technology provides an animation realization for gradually
changing a fill color in a certain display area from a first solid color to a second solid
color. Generally, a color value of the fill color is gradually changed from a color value
of the first solid color to a color value of the second solid color in accordance with a
specified rule. However, for complex non-solid color fill schemes, such as fill
25 schemes with gradual change, texture, pattern fill effects, etc., to create harmonious
color effects, the fill schemes need to cooperate with each other when the color values
change, since the color values of pixels in the display area are mostly different. In the
existing technology, the color value change method generates an awkward and stiff

animation with poor effects. As a result, a designer needs to provide fill schemes for an intermediate part of the animation, which means a high workload and low efficiency of animation realization.

SUMMARY

5 **[0005]** The present application provides a method for realizing color tween animation, where a display area of the color tween animation includes two overlapping layers: one layer is non-transparent and the other layer is completely transparent. The method includes the following: displaying a start fill scheme at the non-transparent layer and displaying an end fill scheme at the completely transparent
10 layer; and gradually changing transparency of the non-transparent layer to completely transparent and gradually changing transparency of the completely transparent layer to completely non-transparent.

[0006] The present application further provides an apparatus for realizing color tween animation, where a display area of the color tween animation includes two
15 overlapping layers: one layer is non-transparent and the other layer is completely transparent. The apparatus includes the following: a start and end display unit, configured to display a start fill scheme at the non-transparent layer and display an end fill scheme at the completely transparent layer; and a first transparency change unit, configured to gradually change transparency of the non-transparent layer to
20 completely transparent and gradually change transparency of the completely transparent layer to completely non-transparent.

[0007] It can be seen from the previous technical solutions that in the implementations of the present application, the layer displaying the start fill scheme is gradually changed from non-transparent to completely transparent. The layer
25 displaying the end fill scheme is gradually changed from completely transparent to non-transparent, through changing transparency of the two overlapping layers in the display area of the color tween animation, to create animation effects of changing the start fill scheme to the end fill scheme. No matter if a fill scheme is complex or not, smooth and natural animation effects can be automatically created by using software
30 without extra work of a designer, thereby improving animation realization efficiency.

BRIEF DESCRIPTION OF DRAWINGS

[0008] FIG. 1 is a flowchart illustrating a method for realizing color tween animation according to implementation 1 of the present application.

[0009] FIG. 2 is a flowchart illustrating a method for realizing color tween animation according to implementation 2 of the present application.

[0010] FIG. 3 is a schematic diagram illustrating a correspondence among a credit score of a user, a credit rating, and a fill scheme according to an application example of the present application.

[0011] FIG. 4 is a flowchart illustrating animation display of a user credit rating, according to an application example of the present application.

[0012] FIG. 5 is a hardware structural diagram illustrating a device according to an implementation of the present application.

[0013] FIG. 6 is a logical structural diagram illustrating an apparatus for realizing color tween animation according to an implementation of the present application.

DESCRIPTION OF IMPLEMENTATIONS

[0014] Implementations of the present application provide a new method for realizing color tween animation. A start fill scheme and an end fill scheme are respectively displayed on two overlapping layers in a display area. The layer displaying the start fill scheme is gradually changed from non-transparent to completely transparent. Moreover, the layer displaying the end fill scheme is gradually changed from completely transparent to non-transparent. The layers create animation effects of changing the start fill scheme to the end fill scheme. Smooth and natural animation effects can be created without extra work of a designer, thereby improving animation realization efficiency and resolving a problem in the existing technology.

[0015] The implementations of the present application can be applied to any device having computing, storage and display capabilities, including devices such as a mobile phone, a tablet computer, a personal computer (PC), a notebook, and a server.

[0016] In the implementations of the present application, the color tween animation is performed in a certain display area. The display area can be in any shape, of any size, and at any location. The display area can be fixed with a fixed shape, a

fixed size, and a fixed location, or can change dynamically with a changing shape, a changing size, and/or a changing location. That is, the technical solutions for realizing color tween animation in the implementations of the present application can be used alone, or can be used in combination with technical solutions for realizing other tween animation (for example, shape tween animation, displacement tween animation, scaling tween animation, or rotation tween animation in a fill area).

[0017] In this implementation of the present application, the two layers are generated in the display area of the color tween animation during initialization. The two layers overlap each other. The transparency of one layer is set to completely transparent, and the transparency of the other layer is set to non-transparent. For a specific method for generating a layer and a specific method for setting transparency of a layer, references can be made to the existing technology for implementation. Details are not described.

[0018] The two overlapping layers have an upper and a lower layer superimposed on each other. When the upper layer is non-transparent, the lower layer is completely blocked, and the user views content displayed at the upper layer in the display area. When the upper layer is completely transparent, the lower layer is completely revealed, and the user views content displayed at the lower layer in the display area. Alternatively, when the transparency of the upper layer is a middle value between completely transparent and non-transparent, the content that the user views in the display area is the visual effects obtained after the upper layer and the lower layer are superimposed based on the middle value of the transparency.

[0019] In implementation 1 of the present application, color tween animation realizes change from the start fill scheme to the end fill scheme with animation effects. The start fill scheme or the end fill scheme can be a solid color fill scheme, a fill scheme with fill effects of gradual change, texture, pattern, etc., or various other fill schemes with color change that are generated by the user. FIG. 1 is a flowchart illustrating a method for realizing color tween animation in this implementation.

[0020] Step 110: Display a start fill scheme at a non-transparent layer, and display an end fill scheme at a completely transparent layer.

[0021] After initialization is completed, there are two overlapping layers in a display area. One layer is the non-transparent layer, and the other layer is the completely transparent layer. The start fill scheme is displayed at the non-transparent layer, and the end fill scheme is displayed at the completely transparent layer. As

such, the user can view visual effects corresponding to the start fill scheme in the display area.

[0022] Step 120: Gradually change transparency of the non-transparent layer to completely transparent, and gradually change transparency of the completely transparent layer to completely non-transparent.

[0023] In an animation duration of the tween animation, the transparency of the non-transparent layer is gradually changed from non-transparent to completely transparent. In the same animation duration, the transparency of the completely transparent layer is gradually changed from completely transparent to non-transparent.

As such, the user can view the visual effects corresponding to the start fill scheme in the display area. The visual effects are changed with transparency in the animation duration, and are gradually changed to the visual effects corresponding to the end fill scheme. Therefore, the animation realization from the start fill scheme to the end fill scheme is completed.

[0024] The existing technology provides a plurality of methods for gradually changing transparency in the animation duration. References can be made to these methods for the implementations of the present application. It should be noted that methods for changing the transparency of the two layers can be independently performed, or can be cooperatively performed in an associated method. The transparency 0 indicates being completely transparent, and the transparency 1 indicates being non-transparent. In an example, the transparency of the upper layer can be changed from 0 to 1 at a constant speed in the animation duration, and the transparency of the lower layer can be quickly changed from 1 to 0 in the animation duration. In another example, the transparency of the upper layer can be quickly changed from 0 to 1 in animation duration, and the sum of the transparency of the upper layer and the transparency of the lower layer is 1.

[0025] After the transparency of the two layers is changed, the color tween animation ends. As such, the current completely transparent layer, that is, the layer displaying the start fill scheme in this implementation, can be deleted to release the occupied system resource in a timely manner.

[0026] In implementation 2 of the present application, there are N fill schemes (N is a natural number greater than or equal to 2). In the display area, the first fill scheme is changed to the second fill scheme with animation effects, and then the second fill scheme is changed to the third fill scheme with animation effects (when N is greater

than or equal to 3), until the Nth fill scheme. Afterwards, the animation ends. Each fill scheme can be a solid color fill scheme, a fill scheme with fill effects of gradual change, texture, pattern, etc., or various other fill schemes with color changes that are generated by the user.

5 **[0027]** FIG. 2 is a flowchart illustrating a method for realizing color tween animation in this implementation. This implementation mainly describes parts different from implementation 1. For a same part, references can be made to descriptions in implementation 1, and the description is not repeated here.

10 **[0028]** Step 210: Display the first fill scheme at a non-transparent layer, and display the second fill scheme at a completely transparent layer.

[0029] Step 220: Gradually change transparency of the non-transparent layer to completely transparent, and gradually change transparency of the completely transparent layer to completely non-transparent.

15 **[0030]** Step 230: After the transparency change is completed, if the kth (k is a natural number not greater than N) fill scheme displayed at a current non-transparent layer is not the Nth fill scheme, display the (k+1)th fill scheme at a current completely transparent layer; gradually change transparency of the current completely transparent layer to non-transparent, and gradually change transparency of the current non-transparent layer to completely transparent; and repeat this step until k is equal to
20 N.

[0031] After the transparency change is completed, it is assumed that the kth fill scheme is displayed at the current non-transparent layer (that is, the layer changed from completely transparent to non-transparent in the latest transparency change process), and the kth fill scheme is the last fill scheme of the color tween animation
25 up until now in the current implementation. If k is not equal to N, the kth fill scheme further needs to be changed to the (k+1)th fill scheme through animation. If k is equal to N, the animation ends.

[0032] When k is not equal to N, the (k+1)th fill scheme is displayed at the current completely transparent layer (that is, the layer changed from non-transparent to completely transparent in the latest transparency change process). Then, in an
30 animation duration from the kth fill scheme to the (k+1)th fill scheme, the transparency of the current completely transparent layer is gradually changed to non-transparent, and the transparency of the current non-transparent layer is gradually changed to completely transparent.

[0033] In this transparency change process, the user can view the visual effects corresponding to the k th fill scheme in the display area. The visual effects are changed with transparency in the animation duration and are gradually changed to the visual effects corresponding to the $(k+1)$ th fill scheme.

5 **[0034]** The current non-transparent layer and the current completely transparent layer after the transparency change are interchanged with the current non-transparent layer and the current completely transparent layer before the transparency change. This step is repeated until k is equal to N . As such, the first fill scheme is changed to the second fill scheme with animation effects in the display area and sequentially
10 changed to the N th fill scheme one by one based on an arrangement order of the fill schemes. Afterwards, the animation ends.

[0035] When k is equal to N and the animation ends, the current completely transparent layer can be deleted.

[0036] It can be seen that, in implementation 1 and implementation 2 of the
15 present application, the two overlapping layers are arranged in the display area: the first layer is gradually changed from non-transparent to completely transparent, and the second layer is gradually changed from completely transparent to non-transparent, to create animation effects of gradually changing the fill scheme at the first layer to the fill scheme at the second layer. In addition, the change process is smooth and
20 natural, and extra manual work is avoided while good effects are achieved, thereby improving animation realization efficiency.

[0037] In an application example of the present application, an online credit reference system generates a credit score for each user by using a specified evaluation algorithm based on dimensions such as a credit history, a behavior preference, an
25 agreement fulfillment ability, an identity, and a personal connection, and determines a credit rating of the user based on the credit score of the user. Each credit rating corresponds to a specified range of credit scores. The range of the credit scores of the user in the credit reference system is 350 to 950, which is divided into five credit ratings from 1 to 5. A higher rating indicates better credit.

30 **[0038]** The user can view the user's credit rating at the client application (app) of the credit reference system. In the app, different credit ratings of the user are displayed in a circular area with a fill scheme with a gradual change effect of different colors. The k th fill scheme corresponds to the k th credit rating. FIG. 3 shows a correspondence among a credit score of a user, a credit rating, and a fill scheme in the

credit reference system. The first fill scheme is a red fill scheme where a color value is gradually changed from #EA5254 to #F4653C, and corresponds to the first credit rating. Similarly, the second to the fifth fill schemes are respectively orange, green, cyan, and blue fill schemes with a gradual change effect, and respectively correspond to the second to the fifth credit ratings.

[0039] To improve user experience, when the user opens a page displaying the user's credit rating, a credit rating enhancement process of the user is displayed in an animation method from the first credit rating to the credit rating of the user. That is, in the circular area representing the credit rating, the intermediate fill schemes are traversed in an animation method from the first fill scheme to the Nth fill scheme corresponding to a credit rating N of the user. Afterwards, the animation ends.

[0040] FIG. 4 shows an animation realization process in this application example.

[0041] Step 401: Initialize two UIViews in a circular area, where an upper layer is a view Front, and a lower layer is a view Bottom; and set transparency of the Front to 0, and set transparency of the Bottom to 1. In this application example, the transparency 0 indicates completely transparent, and the transparency 1 indicates non-transparent.

[0042] Step 402: Obtain a credit rating N of a user.

[0043] Step 403: Determine whether N is 1; and if yes, perform step 404; or if no, go to step 405.

[0044] Step 404: Display the first fill scheme at the Bottom, delete the Front, and end the procedure. When the credit rating of the user is 1, there is no need to perform animation display, and the first fill scheme is directly displayed.

[0045] Step 405: Set b to 1, and set f to 2. Here, b indicates the sequence number of the fill scheme displayed at the Bottom, and f indicates the sequence number of the fill scheme displayed at the Front.

[0046] Step 406: Display the bth fill scheme at the Bottom, and display the fth fill scheme at the Front.

[0047] Step 407: Determine whether transparency of the current Front is 0; and if yes, perform step 408; or if no, go to step 411. This step is used to determine whether the current completely transparent layer is Front or Bottom.

[0048] Step 408: In a predetermined duration, change the transparency of the Front from 0 to 1 at a constant speed, and change the transparency of the Bottom from 1 to 0 at a constant speed. The predetermined duration is a predetermined duration for

displaying animation effects between adjacent credit ratings.

[0049] Step 409: Determine whether f is equal to N ; and if yes, end the animation, and go to step 414; or if no, perform step 410. In this step, alternatively, whether b is equal to $(N-1)$ can be determined.

5 **[0050]** Step 410: Add 2 to b , refresh the display at the Bottom (that is, display the $(b+2)$ th fill scheme), and go to step 407.

[0051] Step 411: In a predetermined duration, change the transparency of the Front from 1 to 0 at a constant speed, and change the transparency of the Bottom from 0 to 1 at a constant speed.

10 **[0052]** Step 412: Determine whether b is equal to N ; and if yes, end the animation, and go to step 414; or if no, perform step 413. In this step, alternatively, whether f is equal to $(N-1)$ can be determined.

[0053] Step 413: Add 2 to f , refresh the display at the Front (that is, display the $(f+2)$ th fill scheme), and go to step 407.

15 **[0054]** Step 414: Delete a view whose current transparency is 0.

[0055] Corresponding to the previous procedure implementation, the implementations of the present application further provide an apparatus for realizing color tween animation. The apparatus can be implemented by software, hardware, or a combination of hardware and software. Software implementation is used as an example. As a logical apparatus, the software is formed by reading a corresponding computer program instruction in a non-volatile memory and running the instruction in a memory by a processor in a device that the software is located. In terms of hardware implementation, in addition to the CPU, the memory, and the non-volatile memory shown in FIG. 5, the device in which the apparatus is located generally further includes other hardware such as a chip for sending and receiving radio signals, and/or other hardware such as a card configured to implement a network communication function.

20

25

[0056] FIG. 6 shows an apparatus for realizing color tween animation according to an implementation of the present application. A display area of the color tween animation includes two overlapping layers: one layer is non-transparent and the other layer is completely transparent. The apparatus includes a start and end display unit and a first transparency change unit. The start and end display unit is configured to display a start fill scheme at the non-transparent layer and display an end fill scheme at the completely transparent layer. The first transparency change unit is configured to

30

gradually change transparency of the non-transparent layer to completely transparent and gradually change transparency of the completely transparent layer to completely non-transparent.

5 **[0057]** In an example, the color tween animation includes changing from the first fill scheme to the second fill scheme and sequentially changing to the Nth fill scheme one by one, where N is a natural number greater than 1. The start fill scheme is the first fill scheme, and the end fill scheme is the second fill scheme. The apparatus further includes a second transparency change unit, configured to the following: after the transparency change is completed, if the kth fill scheme displayed at a current non-transparent layer is not the Nth fill scheme, display the (k+1)th fill scheme at a current completely transparent layer; gradually change transparency of the current completely transparent layer to non-transparent, and gradually change transparency of the current completely non-transparent layer to completely transparent; and repeat this step until k is equal to N, where k is a natural number not greater than N.

10 **[0058]** In the previous example, N is a credit rating of a user, and the kth credit rating corresponds to the kth fill scheme. The apparatus can further include an animation cancellation unit, configured to display the first fill scheme at the non-transparent layer and delete the completely transparent layer for a user whose credit rating is 1.

15 **[0059]** Optionally, the apparatus further includes an animation end unit, configured to delete the current completely transparent layer after the animation ends.

20 **[0060]** Optionally, the start fill scheme and the end fill scheme include a fill scheme with effects of gradual change, texture, or pattern.

25 **[0061]** Optionally, the display area of the color tween animation includes a fixed display area or a dynamically changing display area.

30 **[0062]** The previous descriptions are merely examples of the implementations of the present application, but are not intended to limit the present application. Any modification, equivalent replacement, or improvement made without departing from the spirit and principle of the present application shall fall within the protection scope of the present application.

35 **[0063]** In a typical configuration, the computing device includes one or more processors (CPU), an input/output interface, a network interface, and a memory.

40 **[0064]** The memory can include a non-persistent memory, a random access memory (RAM), a non-volatile memory, and/or another form that are in a

computer-readable medium, for example, a read-only memory (ROM) or a flash memory (flash memory). The memory is an example of the computer-readable medium.

[0065] The computer-readable medium includes persistent, non-persistent,
5 movable, and unmovable media that can store information by using any method or technology. The information can be a computer-readable instruction, a data structure, a program module, or other data. An example of a computer storage medium includes, but is not limited to a phase change memory (PRAM), a static random access memory (SRAM), a dynamic random access memory (DRAM), another type of random access
10 memory (RAM), a read-only memory (ROM), an electrically erasable programmable read-only memory (EEPROM), a flash memory or another memory technology, a compact disc read-only memory (CD-ROM), a digital versatile disc (DVD) or other optical storage, a cassette magnetic tape, tape and disk storage or another magnetic storage device or any other non-transmission media that can be configured to store
15 information that a computing device can access. As described in this specification, the computer-readable medium does not include transitory computer-readable media (transitory media), such as a modulated data signal and a carrier.

[0066] It should be further noted that, the term "include", "contain", or any other variant thereof is intended to cover a non-exclusive inclusion, so that a process, a
20 method, a commodity, or a device that includes a series of elements not only includes those elements but also includes other elements that are not expressly listed, or further includes elements inherent to such process, method, commodity, or device. An element preceded by "includes a ..." does not, without more constraints, preclude the existence of additional identical elements in the process, method, commodity, or
25 device that includes the element.

[0067] A person skilled in the art should understand that the implementations of the present application can be provided as a method, a system, or a computer program product. Therefore, the present application can use a form of hardware only implementations, software only implementations, or implementations with a
30 combination of software and hardware. In addition, the present application can use a form of a computer program product that is implemented on one or more computer-usable storage media (including but not limited to a disk memory, a CD-ROM, an optical memory, etc.) that include computer-usable program code.

CLAIMS

1. A computer-implemented method, comprising:

displaying, on a display of a computing device, a start fill scheme in a first layer that is initially non-transparent;

displaying, on the display, an end fill scheme in a second layer that overlaps the first layer and is initially completely transparent; and

gradually changing a first transparency value of the first layer to a value corresponding to transparency, and gradually changing a second transparency of the second layer to a value corresponding to non-transparency, wherein both the first transparency value and the second transparency value are gradually changed during a particular time period, and wherein gradually changing the first transparency value and the second transparency value includes changing the values by a plurality of increments over the particular time period and displaying a colour tween animation comprising:

changing, on the display, from a first fill scheme to a second fill scheme, including sequentially changing to an Nth fill scheme, wherein N is an integer greater than 1;

setting the start fill scheme to the first fill scheme, and setting the end fill scheme to the second fill scheme; and

performing a transparency change to the first layer and the second layer transparent, and

wherein the transparency change comprises:

when a kth fill scheme displayed at a current non-transparent layer is not an Nth fill scheme, displaying a (k+1)th fill scheme at a current completely transparent layer;

gradually changing transparency of the current completely transparent layer to non-transparent, and gradually changing transparency of the current non-transparent layer to completely transparent; and

repeating the displaying and gradually changing until k is equal to N, wherein k is a natural number.

2. The computer-implemented method of claim 1, wherein N is a credit rating of a user, wherein a kth credit rating corresponds to the kth fill scheme, the method further comprising:

for a user whose credit rating is 1, displaying the first fill scheme at the current non-transparent layer; and

deleting the completely transparent layer.

3. The computer-implemented method of claim 1, further comprising deleting the current completely transparent layer after the colour tween animation ends.

4. The computer-implemented method of claim 1, wherein the start fill scheme and the end fill scheme comprise a fill scheme with effects of gradual change, a texture, or a pattern.

5. The computer-implemented method of claim 1, wherein a display area of the colour tween animation comprises a fixed display area.

6. The computer-implemented method of claim 1, wherein a display area of the colour tween animation comprises a dynamically changing display area.

7. A non-transitory, computer-readable medium storing one or more instructions executable by a computer system to perform the method of claims 1 to 6.

8. A computer-implemented system, comprising:

one or more computers; and

one or more computer memory devices interoperably coupled with the one or more computers and having tangible, non-transitory, machine readable media storing one or more instructions that, when executed by the one or more computers, perform the method of claims 1 to 6.