Title: CURSOR CONTROL SYSTEM AND METHOD

Abstract: A display system, comprising a graphic user interface generated by a processor for a display device, the graphic user interface further comprising a field having a N x M matrix of spaces, wherein N and M are integers greater than three, and wherein each space of the matrix includes one of a plurality of different items. A cursor having an associated item and configured to allow a user to exchange a selected item from the field with the cursor item. Wherein forming a predetermined arrangement of three or more identical items results in the items within the arrangement being removed from the field.
CURSOR CONTROL SYSTEM AND METHOD

TECHNICAL FIELD

[0001] The present disclosure relates to control of interactive displays, and more specifically to a cursor control system and method that provides additional cursor control functionality.

BACKGROUND OF THE INVENTION


SUMMARY OF THE INVENTION

[0003] A cursor system is provided that includes a graphic user interface generated by a processor for a display device, such as where the graphic user interface further includes a field having a N x M matrix of spaces, wherein N and M are integers greater than three, and wherein each space of the matrix includes one of a plurality of different items. For example, an eight by eight matrix can be generated that has 64 items, where one item is shown in each space of the matrix, and where each of the items in the 64 spaces are selected from a set of six possible icons, such as a diamond and a water drop. A cursor having an associated icon allows a user to exchange a selected icon from the field with the cursor item, such as where the cursor icon is a diamond and the selected icon from the field is a water drop, such that the cursor icon becomes the water drop and the field item becomes the diamond.

[0004] Other systems, methods, features, and advantages of the present disclosure will be or become apparent to one with skill in the art upon examination of the following drawings and detailed description. It is intended that all such additional
systems, methods, features, and advantages be included within this description, be within the scope of the present disclosure, and be protected by the accompanying claims.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0005] Aspects of the disclosure can be better understood with reference to the following drawings. The components in the drawings are not necessarily to scale, emphasis instead being placed upon clearly illustrating the principles of the present disclosure. Moreover, in the drawings, like reference numerals designate corresponding parts throughout the several views, and in which:

[0006] FIGURE 1 is a diagram of a display in accordance with an exemplary embodiment of the present disclosure;

[0007] FIGURE 2 is a diagram of an algorithm for controlling a display in accordance with an exemplary embodiment of the present disclosure;

[0008] FIGURE 3 is a diagram of an algorithm for a display in accordance with an exemplary embodiment of the present disclosure;

[0009] FIGURE 4 is a diagram of an algorithm for a display in accordance with an exemplary embodiment of the present disclosure; and

[0010] FIGURE 5 is a diagram of a system for controlling a display in accordance with an exemplary embodiment of the present disclosure.

DETAILED DESCRIPTION OF THE INVENTION

[0011] In the description that follows, like parts are marked throughout the specification and drawings with the same reference numerals. The drawing figures might not be to scale and certain components can be shown in generalized or schematic
form and identified by commercial designations in the interest of clarity and conciseness.

[0012] FIGURE 1 is a diagram of a display 100 in accordance with an exemplary embodiment of the present disclosure. A used herein, "virtual" refers to computer simulated spaces, objects or other suitable computer simulations, such as a virtual lobby, a virtual world, a virtual room, a virtual item, or other suitable constructs. Display 100 can be generated as a graphic user interface on a screen of a general purpose computer, a computer network, a cellular telephone, a tablet computer, a notebook computer, a personal gaming device or other suitable systems, and can be implemented in hardware or a suitable combination of hardware and software, such as one or more software systems operating on a processor. In addition to a processor, user interface devices such as a touch screen interface, a keyboard, a mouse, a stylus, a voice response system utilizing a microphone and voice processing system or other suitable user interface devices can be provided. Data memory devices can be provided for storing one or more algorithms for generating display 100 and controlling the functionality of display 100 as described herein. The data memory devices can be configured to provide permanent or transient electronic, optical, magnetic or other suitable data storage. Additional data interface devices such as USB ports, fire wire ports, serial ports, PCI ports or other suitable data interface devices can also be provided.

[0013] As used herein, "hardware" can include a combination of discrete components, an integrated circuit, an application-specific integrated circuit, a field programmable gate array, or other suitable hardware. As used herein, "software" can include one or more objects, agents, threads, lines of code, subroutines, separate software applications, two or more lines
of code or other suitable software structures operating in two or more software applications or on two or more processors, or other suitable software structures. In one exemplary embodiment, software can include one or more lines of code or other suitable software structures operating in a general purpose software application, such as an operating system, and one or more lines of code or other suitable software structures operating in a specific purpose software application. As used herein, the term "couple" and its cognate terms, such as "couples" and "coupled," can include a physical connection (such as a copper conductor), a virtual connection (such as through randomly assigned memory locations of a data memory device), a logical connection (such as through logical gates of a semiconductor device), other suitable connections, or a suitable combination of such connections.

[0014] Display 100 includes a field that comprises an eight by eight matrix of spaces. In one exemplary embodiment, the field can be implemented as algorithms, as one or more objects for each space having graphical, data and operational attributes or in other suitable manners. Display 100 includes a plurality of instances of items, such as cross item 102, pentagram item 104, five point star item 106, circle item 108, triangle item 110, six point star item 112 and X mark item 114. Likewise, gems, colors, pictures, figures, or other suitable items can be used to populate the field for a suitable purpose. The items can be randomly generated at the start of a session, can be assigned in a predetermined configuration, or other suitable processes can also or alternatively be used.

[0015] Display 100 also includes feature 116, feature 118 and feature 120, each of which can be implemented as algorithms, as one or more objects having graphical, data and operational attributes, or in other suitable manners.
[0016] In one exemplary embodiment, display 100 can be used to implement a suitable process. At the start of the process, the field can be populated with items in each of the 64 spaces of the field, and feature 116 can be used for a suitable purpose, such as for a countdown timer that is initiated with a predetermined period of time, such as one minute, a move counter that is initiated with a predetermined number of available moves (such as where each swap of a cursor item with a field item constitutes a move), such as 30 moves, or in other suitable manners. A cursor item 122, which can be implemented as one or more objects having graphical, data and operational attributes, can be generated having a randomly selected item, such as circle item 108 as shown. The user can move cursor item 122 using the user interface device so as to place cursor item 122 over an item in one space of the field, and can activate a control to cause the item in the selected field space to be replaced with the cursor item, and to cause the cursor item to be replaced with the corresponding item from the field space, such as by selecting a control on the user interface device, by tapping a touch screen interface, or in other suitable manners. In this exemplary embodiment, cursor item 122 can be placed over the field space containing the triangle B item, and can be activated to cause the triangle B item to be replaced with a circle item, and to cause the circle item that is currently cursor item 122 to be replaced with a triangle item. The user can then move the cursor item 122 (which is now a triangle item) over the field space containing the cross F item, and can activate the cursor control to replace the cross F item in the field space with a triangle, so as to form a line of three triangle items in the three adjacent field spaces (i.e., new triangle item F and existing triangle items G and H).
[0017] After a line of three triangle items has been formed in this manner, a function can be performed. In one exemplary embodiment, each space in the field can be implemented as an object having graphical, data and operational attributes, such as object item type attributes that identify the item type that is presently represented in the space, graphical attributes that generate a graphical image associated with that item type, and operational attributes for monitoring the object item type attributes in adjacent field spaces. In this exemplary embodiment, when a center field space object detects that the field space objects to the left and right of the center field space object have the same object item type attribute as the center field space object, the center field space object can activate an operational attribute for all three field space objects to cause the field space objects, such as to change to a new object, an empty object item type state, or other suitable functions. Likewise, each field space object in the field can have operational attributes that cause the field space objects to interact with adjacent field space objects to simulate movement of the items adjacent to the empty spaces into the empty spaces, and to change their state to empty space states. In this exemplary embodiment, each object can have operational attributes that are modeled in pseudo code as follows:

10 if field space object below has empty space item type, transmit item type attribute to field space object below

20 change item type attribute to empty space

[0018] In this manner, a suitable interactive function can be performed that is initiated using the interactive cursor process to initiate one or more functions.

[0019] Likewise, when a field space object at the top of the field changes state to an empty object item type, a new item
type can be randomly generated for that field space object, so as to maintain an item in each space of the field, or other suitable functions can be performed.

[0020] Likewise, in addition to removing a horizontal line of three adjacent identical items, a vertical line of three adjacent identical items, a vertical or horizontal line of four or more adjacent identical items, or other suitable configurations of items can be removed in a similar manner, or other suitable functions can be performed.

[0021] In another exemplary embodiment, a class of items can be excluded from the functions, such as X mark item 114. In this exemplary embodiment, when a field space object has an X mark item attribute, the operational attribute for monitoring for identical adjacent field space object item attributes can be suspended. In addition, when field space objects at the bottom of the field change to an X mark item attribute, those field space objects can have an additional operational attribute that causes the field space objects to change to an empty space attribute, to change to an empty space attribute if there is one or more adjacent field space objects with X mark item attributes, or in other suitable manners.

[0022] In addition, for field space objects with an X mark item attribute, the cursor exchange operational attribute can also be excluded, such as to prevent the cursor item 122 from being exchanged with an field space object having an X mark item attribute.

[0023] Feature 118 can be used to provide an additional cursor item 122 for placement in the field. In one exemplary embodiment, feature 118 can be implemented as an object having graphical, data and operational attributes, such as an operational attribute that monitors field space objects, feature 124 or other suitable data or objects to determine if matching
items have been removed. As icons are processed, a progress indicator graphical attribute of feature 118 can be modified to represent a progress towards a metric. In one exemplary embodiment, the metric can be an icon 126 that can be placed in a field space object and which causes a predetermined number of item attributes for adjacent field space objects to be changed, such as to an empty space attribute or other suitable attributes. Icons 126 can be implemented as one or more objects that have graphical, data and operational attributes. In this exemplary embodiment, a user can obtain different types of icons 126, such as icons that remove left adjacent field space object items, right adjacent field space object item attributes, left and right adjacent field space object item attributes, top adjacent field space object item attributes, bottom adjacent field space object item attributes, top and bottom adjacent field space object item attributes, or other suitable combinations of field space object item attributes. In another exemplary embodiment, icon 126 can have no effect on field space objects having an X mark item attribute.

[0024] Feature 120 can be implemented as one or more objects having graphical, data and operational attributes that are used to generate a data display as a function of a mode of operation. In one exemplary embodiment, feature 120 can generate a status indicator for an operational mode that shows a user's progress through a number of mode objectives. In this exemplary embodiment, the operational mode can have an associated architecture for allowing a user to progress through mode objectives, such as a predetermined number of objectives, where each objective has a predetermined number of sub-objectives. In another exemplary embodiment, feature 120 can generate a status indicator for a mode, such as a status indicator that shows other user data related to the mode.
Feature 124 can be implemented as one or more objects having graphical, data and operational attributes that are used to generate display data. In one exemplary embodiment, users can track the status of functions performed by icons, the number of icon replacements or other suitable display functional attributes.

FIGURE 2 is a diagram of an algorithm 200 for controlling a display in accordance with an exemplary embodiment of the present disclosure. Algorithm 200 can be implemented in hardware or a suitable combination of hardware and software, such as one or more software systems operating on a general purpose processing platform.

Algorithm 200 begins at 202, where a feature is selected. In one exemplary embodiment, a graphic user interface can be generated that allows a user to select one or more features that provide one or more predetermined functions. In this exemplary embodiment, each feature can be selected from an available set of features, or can be selected in other suitable manners. Features can provide functions such as:

- Providing a graphic enhancement to highlight available matches using the current cursor item
- Providing an enhancement to the number of icons that the current icon can interact with
- Providing additional time to a countdown timer
- Providing a multiplier for functions performed
- Causing items adjacent to matched items to be removed from the display

After one or more functions are selected by the user, the algorithm can apply the function ability, such as by modifying the operational attributes of one or more objects in
display 100 to implement the selected function. The algorithm then proceeds to 204.

[0029] At 204, the field is populated with icons. In one exemplary embodiment, each space of the field can be implemented as one or more objects having graphical, data and operational attributes, and can include an icon type attribute that is selected from a predetermined set of icon types, such as those shown in display 100 or other suitable items types. Likewise, the graphical attribute for each field space object can be determined as a function of the item type attribute, or in other suitable manners. In another exemplary embodiment, a predetermined item distribution can be used, such as for different modes of operation. After the field is populated with items, the algorithm proceeds to 206.

[0030] At 206, a timer is initiated. In one exemplary embodiment, the timer can be implemented as one or more objects having graphical, data and functional attributes, such as a data attribute for a predetermined initial amount of time, an operational attribute to allow the timer to be paused, or other suitable attributes. After the timer is initiated at 206, the algorithm proceeds to 208.

[0031] At 208, a user swaps a cursor item with a selected item. In one exemplary embodiment, the cursor can be implemented as one or more objects having graphical, data and operational attributes, and can include an item type attribute that can be randomly generated, that can be predetermined, or that can be assigned in other suitable manners. The cursor object can have operational attributes that allow the cursor to be moved within display 100, such as by using a touch screen interface, a mouse, a stylus, or other suitable user interface devices. When a user places the cursor over a field space object having an associated item, the user can activate a user
control to cause an operational attribute of the cursor object, the field space object or other suitable objects or systems to exchange the item type attribute of the cursor with the item type attribute of the field space object, so as to swap the cursor item with the field space item. For example, if the cursor item is a cross item 102 and the field space object item is a circle item 108 as shown in display 100, activation of the cursor control can cause the cursor item to be changed to a circle item 108 and the field space object item to be changed to a cross item 102, such as by changing a item type data attribute of the cursor object and the field space object, or in other suitable manners. The algorithm then proceeds to 210.

[0032] At 210, it is determined whether a first condition has been satisfied, such as three or more matching items are located in a line or other suitable configurations within the field. In one exemplary embodiment, each field space object can include operational attributes that compare the item type attribute of the field space object with the item type attribute of adjacent field space objects or other suitable configurations of field space objects. If it is determined that the first condition is satisfied, the algorithm proceeds to 212, otherwise, the algorithm proceeds to 220.

[0033] At 212, a first function is performed that is associated with the first condition. In one exemplary embodiment, each field space object containing a matching item can execute an operational attribute that generates an animation or other suitable graphic displays that cause the item graphic within the field space to disappear, such as to momentarily create the appearance of an empty space. In addition, a tracking object or other suitable algorithms can be incremented. In addition, a function algorithm that causes items adjacent to the matching items to be eliminated, that causes a metric to be
accelerated, or other suitable functions can also or alternatively be implemented. The algorithm then proceeds to 214.

[0034] At 214, a second function related to the first condition is performed. In one exemplary embodiment, each field space object can include an operational attribute that monitors an item type attribute of one or more adjacent field space objects, such as field space objects located directly underneath the field space object. If the operational attribute of a first field space object determines that a second adjacent field space object represents an empty space, then operational attributes of the first field space object, the second field space object or other suitable objects or algorithms can cause the item type attribute of the first field space object and the second field space object to be exchanged, such as to swap the empty space item type attribute of the second field space object with the item type attribute of the first field space object. In this exemplary embodiment, each item of the field that is directly above an empty field space will appear to shift downwards one space. Likewise, lateral shifts, diagonal shifts, random shifts or other suitable functions can also or alternatively be utilized. The algorithm then proceeds to 216.

[0035] At 216, a third function related to the first condition is performed. In one exemplary embodiment, after field items have been shifted to fill in the empty spaces created after matching items have been removed, empty spaces will exist at a border of the field, such as the top border if a downward shift is used, the left border if a right shift is used, and so forth. New icons can be added to these empty spaces, such as by randomly selecting an item type attribute for each empty field space object, by assigning a predetermined item type attribute to each empty field space object, by applying a
functional operational attribute to assign a special item type attribute to the empty field space object, or in other suitable manners. After the third function has been performed, the algorithm proceeds to 218.

[0036] At 218, it is determined whether a third condition has occurred as a result of the first, second and third functions. For example, when items are shifted within the field after matched items are eliminated, the new configuration of items within the field may result in a match of three or more items in a row, column or other suitable configurations. Likewise, when new items are added to empty field spaces along the border of the field (such as at the top of the field), the new items may result in a match of three or more items in a row, column or other suitable configurations. Other suitable functions related to the display of icons can also or alternatively be performed. If such a new match exists, the algorithm returns to 212, otherwise, the algorithm returns to 208.

[0037] If it is determined at 210 that the first condition has not been satisfied, the algorithm proceeds to 220, where it is determined whether a second condition has been satisfied. If it is determined that the second condition has not been satisfied, the algorithm returns to 208, otherwise, the algorithm proceeds to 222.

[0038] At 222, the first function is performed, such as discussed above or for other suitable functions related to the second condition. The algorithm then proceeds to 224, where the second function is performed, and the algorithm then proceeds to 226, where the third function is performed, as previously discussed or as otherwise suitable. The algorithm then proceeds to 218.

[0039] During the execution of algorithm 200, if the timer object times out, the algorithm can terminate and a status can
be generated, or other suitable functions can be performed. Likewise, if a function has been activated that provides for additional time, the additional time can be added, an extended time graphic can be generated, or other suitable processes can be implemented. Likewise, a function can be provided that allows the user to make one or more additional cursor item swaps after the expiration of the timer prior to determination of the status.

[0040] Although algorithm 200 has been described as a series of steps, algorithm 200 can also or alternatively be implemented as a state diagram or in other suitable manners. Likewise, although certain embodiments have been described in regards to object oriented programming implementations, other suitable programming types, languages, paradigms or regimes can also or alternatively be used.

[0041] **FIGURE 3** is a diagram of an algorithm 300 for a display in accordance with an exemplary embodiment of the present disclosure. Algorithm 300 can be implemented in hardware or a suitable combination of hardware and software, such as one or more software systems operating on a general purpose processing platform.

[0042] Algorithm 300 begins at 302, where one or more objectives are identified. In one exemplary embodiment, a graphic user interface can be generated that identifies predetermined objectives, such as the elimination of a predetermined number of items, the elimination of a predetermined number of a predetermined item type, or other suitable objectives. In one exemplary embodiment, elimination can be used to remove repeated data from a set of data. The algorithm then proceeds to 304, where the user performs the objective, such as using algorithm 200 or in other suitable
manner, and the results are received. The algorithm then proceeds to 306.

[0043] At 306, it is determined whether the objective has been achieved. In one exemplary embodiment, multiple objectives can be implemented, and it can be determined which of the objectives has been achieved. For example, elimination of a first predetermined number of items can be defined as a first objective, and elimination of a second predetermined number of items can be defined as a second objective, or other suitable multiple objectives can be defined. If it is determined that the objective has been met, the algorithm proceeds to 308, where a next level is selected, a new set of image data is selected, or where other suitable functions are performed. The algorithm then proceeds to 310.

[0044] At 310, a new objective or objectives are set, where suitable. In one exemplary embodiment, if a user has completed all requirements to proceed to a next level, the new objective or objectives can be associated with the next level. Likewise, where a user has not completed all of the requirements to proceed to the next level, the same objective or objectives for the current level can remain in effect until the user has completed all requirements. The algorithm then returns to 302.

[0045] If it is determined that an objective has not been met at 306, the algorithm proceeds to 312, where it is determined whether the user has not selected a different objective. If it is determined that the user has not selected a different objective, the algorithm proceeds to 308, otherwise, the algorithm returns to 302.

[0046] Although algorithm 300 has been described as a series of steps, algorithm 300 can also or alternatively be implemented as a state diagram or in other suitable manners. Likewise, although certain embodiments have been described in regards to
object oriented programming implementations, other suitable programming types, languages, paradigms or regimes can also or alternatively be used.

[0047] FIGURE 4 is a diagram of an algorithm 400 for a display in accordance with an exemplary embodiment of the present disclosure. Algorithm 400 can be implemented in hardware or a suitable combination of hardware and software, such as one or more software systems operating on a general purpose processing platform.

[0048] Algorithm 400 begins at 402, where one or more features are selected. In one exemplary embodiment, features can be selected based on a level or other suitable processes can be used to select features. In addition, features can be re-selected on a periodic basis, such as once a week or in other suitable manners. The algorithm then proceeds to 404.

[0049] At 404, the features are displayed. The algorithm then proceeds to 406.

[0050] At 406, the features are applied and the results are received. After the results are received at 406, the algorithm proceeds to 408.

[0051] At 408, it is determined whether a ranking change has occurred. If it is determined that a ranking change has not occurred, the algorithm proceeds to 412. If it is determined that a ranking change has occurred, the algorithm proceeds to 410, where the ranking is modified. The algorithm then proceeds to 412.

[0052] At 412, it is determined whether the process has ended. If it is determined that the process has not ended, the algorithm returns to 404. If it is determined that the process has ended at 412, the algorithm proceeds to 414.

[0053] At 414, the final rankings are displayed. The algorithm then proceeds to 416.
At 416, an output is generated for tournament mode competitors.

Although algorithm 400 has been described as a series of steps, algorithm 400 can also or alternatively be implemented as a state diagram or in other suitable manners. Likewise, although certain embodiments have been described in regards to object oriented programming implementations, other suitable programming types, languages, paradigms or regimes can also or alternatively be used.

FIGURE 5 is a diagram of a system 500 for controlling a display in accordance with an exemplary embodiment of the present disclosure. System 500 includes display control system 502, match system 504, item control system 506, feature control system 508, feature mode system 510, function mode system 512, and user systems 514A through 514N (where N is an integer greater than 1), each of which can be implemented as hardware or a suitable combination of hardware and software, and which can be one or more software systems operating on a processor.

Display control system 502 coordinates the activities of a display, interfacing the components of system 500 and other suitable activities. In one exemplary embodiment, display control system 502 can generate code that generates a graphic user interface on one or more of user systems 514A through 514N, where users can select features and can use the display to perform functions. Display control system 502 can also store account data for each user, such as account data that uniquely identifies each user, security data for allowing a user to access their account, data that is used to rank the user and to provide other functionality as described herein. Display control system 502 can also or alternatively be implemented in conjunction with a network, website or system, so as to allow users to access display control system 502 through their network.
account, to allow users to provide alerts or notifications to other network users regarding objectives achieved using display control system 502, or to perform other suitable functions.

[0058] Match system 504 can interface with item control system 506, feature control system 508, feature mode system 510, function mode system 512 or other suitable systems, as described further herein.

[0059] Item control system 506 coordinates the generation and selection of items within match system 504. In one exemplary embodiment, item control system 506 can generate items for placement in a field, can place items are randomly within the field, or can perform other suitable functions. In addition, where each item is not independently controlled, such as by being implemented as one or more objects, item control system 506 can control the function of each item during display, such as by checking the arrangement of items to determine whether matching items are in a predetermined configuration (such as a horizontal or vertical line), to eliminate matching items, to shift items within the field to fill in empty spaces created by eliminated items, to generate new items for empty border spaces within the field that are created by the shifting of items, or in other suitable manners.

[0060] Feature control system 508 allows users to obtain and use features for use during display. In one exemplary embodiment, feature control system 508 can coordinate with display control system 502 to determine whether a user has one or more feature slots, whether a user is eligible to increase the number of feature slots, or to otherwise control user access to features of the display. Feature control system 508 can further allow users to use a feature, to trade features to other users or for other items, or to perform other suitable functions. In one exemplary embodiment, a user can select one
or more features for use based on features that the user currently has in an inventory.

[0061] Feature mode system 510 allows a user to enter a feature mode, such as to allow the user to obtain additional icons or other suitable features. In one exemplary embodiment, feature mode system 510 can be accessed through a virtual environment. Feature mode system 510 can interface with match system 504 to allow a user to match icons in order to achieve predetermined objectives, such as the elimination of a predetermined number of icons, the elimination of a predetermined number of a predetermined type of icon, or other suitable objectives. Feature mode system 510 can coordinate with display control system 502 or other systems to generate status data for users that meet the predetermined objectives.

[0062] Function mode system 512 coordinates the activities of two or more users. In one exemplary embodiment, function mode system 512 can select a group of users for a function. In this exemplary embodiment, the group of users can be selected based on their status. Likewise, other suitable data can be used to select users, such as users having similar demographic data.

[0063] User systems 514A through 514N access and interface with display control system 502 and its associated systems, such as to interact with the virtual environment created by display control system 502. User systems 514A through 514N can be implemented using a web browser interface or application program of a general purpose processor or tablet computer, a web browser interface or application program for a cellular telephone or hand-held device, or on or in conjunction with other suitable devices.

[0064] It should be emphasized that the above-described embodiments are merely examples of possible implementations. Many variations and modifications may be made to the above-
described embodiments without departing from the principles of
the present disclosure. All such modifications and variations
are intended to be included herein within the scope of this
disclosure and protected by the following claims. Although
certain embodiments may have been described in regards to object
oriented programming implementations, other suitable programming
types, languages, paradigms or regimes can also or alternatively
be used.
WHAT IS CLAIMED IS:

1. A display system, comprising:
   a graphic user interface generated by a processor for a display device, the graphic user interface further comprising:
   a field having a N x M matrix of spaces, wherein N and M are integers greater than three, and wherein each space of the matrix includes one of a plurality of different icons;
   a cursor having an associated icon and configured to allow a user to exchange a selected icon from the field with the cursor item; and
   wherein forming a predetermined arrangement of three or more identical items results in the items within the arrangement being removed from the field.

2. The display system of claim 1 wherein the graphic user interface is generated by a display control system, and the display control system further comprises a feature control system configured to allow a user to select one of a plurality of features.

3. The display system of claim 1 wherein the graphic user interface further comprises an icon that is configured to be placed within one of the spaces of the field and to cause one or more icons of the field to be eliminated.

4. The display system of claim 1 wherein the graphic user interface is generated by a display control system, and the display control system further comprises a function mode system configured to allow a user to select a function.
5. The display system of claim 1 wherein the graphic user interface is generated by a display control system, and the display control system further comprises a function mode system configured to allow a user to enter select a function mode of operation.

6. The display system of claim 1 wherein the graphic user interface further comprises:

an icon that is configured to be placed within one of the spaces of the field and to cause one or more items of the field to be eliminated;

a feature that is configured to receive a plurality of data types and increment a meter status indicator upon receipt of the data type; and

wherein the icon is enabled to be placed by the user when the meter status indicator has reached a maximum setting.

7. A method for controlling a display comprising:

electronically generating a virtual field of spaces using a processor, wherein the field is a matrix having dimensions $N$ by $M$, where $N$ and $M$ are integers greater than three;

electronically selecting an item for each space of the field from a set of predetermined items stored in a memory device;

electronically starting a timer;

receiving a user selection from a user input device of one of the field items using a cursor placed over the selected field item, wherein a location of the cursor is controlled using the user input device;

electronically replacing the field item with an item associated with the cursor;
electronically replacing the item associated with the cursor with the field item;

determining with the processor whether a predetermined arrangement of matching items is present in the field after replacing the field item with the item associated with the cursor; and

electronically removing the matching items in the predetermined arrangement if the predetermined arrangement of the matching items is present in the field after replacing the field item with the item associated with the cursor so as to form empty spaces in the field.

8. The method of claim 7 further comprising:
shifting one or more items in spaces adjacent to the empty spaces into the empty spaces; and
randomly generating a new item for each of the empty spaces.

9. The method claim 7 wherein electronically selecting an item for each space of the field from a set of predetermined items stored in a memory device comprises electronically setting an item type attribute for an object associated with each space.

10. The method of claim 7 further comprising electronically incrementing a meter by a predetermined amount in response to the number of matching items that have been removed.

11. The method of claim 10 further comprising:
electronically determining whether the meter has reached a maximum setting; and
enabling a icon if the meter has reached the maximum setting.
12. The method of claim 11 further comprising:
electronically associating the icon with the cursor; and
electronically replacing a second field icon with icon.

13. The method of claim 12 further comprising
electronically removing one or more icons from the field.

14. The method of claim 13 wherein electronically removing
one or more icons from the field comprises:
electronically determining a function type for the icon;
electronically determining an operational attribute for the
icon; and
applying the operational attribute to remove the one or
more items from the field.

15. A system for controlling a display, comprising:
a plurality of space objects arranged in rows and columns
to form a field, wherein each of the plurality of space objects
includes an item type attribute and an operational attribute for
monitoring each space object that is adjacent to the space
object;
a cursor object configured to be controlled by a user
interface device to be placed over any one of the plurality of
space objects, the cursor object having an item type attribute
and an operational attribute for causing the item type attribute
of the space object that the cursor object is placed over to be
exchanged for the item type attribute of the cursor object, and
for causing the former item type attribute of the space object
that the cursor object is placed over to be exchanged for the
item type attribute of the cursor object.
16. The system of claim 15 wherein the operational attribute of a first space object for monitoring each space object that is adjacent to the first space object further comprises an operational attribute for changing the item type attribute of the first space object and the item type object of each space object that is adjacent to the first space object to an empty space item type attribute if a configuration of the first space object and each space object that is adjacent to the first space object is a predetermined configuration.

17. The system of claim 15 wherein the operational attribute of a first space object for monitoring each space object that is adjacent to the first space object further comprises an operational attribute for changing the item type attribute of the first space object from a first item type attribute to an empty space item type attribute if one of the adjacent space objects has an item type attribute that is the empty space item type attribute, and for changing the item type attribute of the adjacent space object to the first item type attribute.
### Category of Subject Matter

**INV.** G06F3/0481 A63F13/00

According to International Patent Classification (IPC) or to both national classification and IPC

### Fields Searched

Minimum documentation searched (classification system followed by classification symbols)

G06F A63F

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

### Documents Considered to be Relevant

<table>
<thead>
<tr>
<th>Category</th>
<th>Citation of document, with indication, where appropriate, of the relevant passages</th>
<th>Relevant to claim No.</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>US 6 093 104 A (KASAHARA SHINICHI [JP] ET AL) 25 July 2000 (2000-07-25) column 1, line 5 - column 4, line 7; figures 4-15 column 5, line 15 - column 6, line 3</td>
<td>1-17</td>
</tr>
</tbody>
</table>

### Further Documents

Further documents are listed in the continuation of Box C.

See patent family annex.

**Note:**
- **A** document defining the general state of the art which is not considered to be of particular relevance
- **E** earlier application or patent but published on or after the international filing date
- **L** document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
- **O** document referring to an oral disclosure, use, exhibition or other means
- **P** document published prior to the international filing date but later than the priority date claimed
- **T** later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
- **X** document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
- **Y** document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
- **Z** document member of the same patent family

### Date of the Actual Completion of the International Search

11 September 2013

### Date of Mailing of the International Search Report

18/09/2013

**Name and Mailing Address of the ISA/Authorized Officer**

European Patent Office, P.B. 5018 Patentlaan 2 NL - 2280 HV Rijswijk
Tel. (+31-70) 340-2040,
Fax: (+31-70) 340-3016

Anticoli, Claud
## INTERNATIONAL SEARCH REPORT

**Information on patent family members**

### PCT/US2013/045362

<table>
<thead>
<tr>
<th>Patent document cited in search report</th>
<th>Publication date</th>
<th>Patent family member(s)</th>
<th>Publication date</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>EP 0817124 A2</td>
<td>07-01-1998</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP H1066784 A</td>
<td>10-03-1998</td>
</tr>
<tr>
<td></td>
<td></td>
<td>TW 396046 B</td>
<td>01-07-2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 6093104 A</td>
<td>25-07-2000</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 2008125854 A</td>
<td>05-06-2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2008120460 A1</td>
<td>22-05-2008</td>
</tr>
<tr>
<td></td>
<td></td>
<td>JP 2009178374 A</td>
<td>13-08-2009</td>
</tr>
<tr>
<td></td>
<td></td>
<td>US 2009197677 A1</td>
<td>06-08-2009</td>
</tr>
</tbody>
</table>