GAME DEVICE, GAME PROCESSING METHOD, INFORMATION RECORDING MEDIUM, AND PROGRAM

Inventor: Masashi Takehiro, Tokyo (JP)

App. No.: 13/139,939

PCT Filed: Dec. 10, 2009

PCT No.: PCT/JP2009/070666

§ 371 (c)(1), (2), (4) Date: Aug. 11, 2011

Publication Classification

Int. Cl. A63F 9/24 (2006.01)

U.S. Cl. 463/43

ABSTRACT

A state storage unit (201) stores a current game state. A history storage unit (202) stores a history of the game state. An operation reception unit (203) receives a command input from a player. An update unit (204) changes the game state according to the received command input and updates the current game state. A display unit (205) displays a state image (251) based on the current game state and a history image (252) based on the game state history. When an instruction to resume a game is received after an interruption of the game, the resumption control unit (206) resumes the game at a time preceding the interrupted in-game time by a predetermined period. The resumption control unit (206) compares the game state history to the game state after being resumed and evaluates the player in accordance with the history of the better game state.
FIG. 3
Receive operation instruction and store current game parameter

Store history of game parameter

Is interruption instruction?

Yes: Temporarily save history of game parameter during predetermined period

Yes: Receive operation instruction and store current game parameter

Store history of game parameter

Has predetermined period passed after resumption?

Yes: Receive operation instruction and store current game parameter

Store history of game parameter

No: Resume game from prior to predetermined period before interruption of the game

FIG. 6
FIG. 7

A

610

COMPARE TEMPORARILY SAVED HISTORY OF GAME PARAMETER WITH HISTORY OF GAME PARAMETER AFTER RESUMPTION

611

GAME PARAMETER AFTER RESUMPTION IS BETTER?

NO

YES

612

DISCARD TEMPORARILY SAVED HISTORY OF GAME PARAMETER

613

UPDATE HISTORY OF GAME PARAMETER WITH TEMPORARILY SAVED HISTORY OF GAME PARAMETER

END
\[
\begin{align*}
(A_{1}) & \quad (A_{2}) \\
\cdots & \quad \cdots \\
& \quad \\
\cdots &= (1+\lambda)A \\
\cdots &= -(2+\lambda)A \\
\cdots &= (N)A \\
\cdots &= (1)A \\
\lambda &= 1 \\
\lambda &= N \\
T &= T_{1} \\
T &= T_{2}
\end{align*}
\]

FIG. 8
FIG. 16
FIG. 19
GAME DEVICE, GAME PROCESSING METHOD, INFORMATION RECORDING MEDIUM, AND PROGRAM

TECHNICAL FIELD

[0001] The present invention relates to a game device, a game processing method, an information recording medium and a program that are preferable for a player to easily grasp a continuity of a game before and after interruption of the game.

BACKGROUND ART

[0002] Most of currently-prevailing games have a function of interrupting and resuming a game by an instruction from a player. For example, if there is a phone call during play, the player can push a pause button to interrupt the game, then after phone conversation, the player can push the pause button again to resume the game. Patent Literature 1 discloses a game device in which several separation points are defined in a game, and when the game is resumed after interruption, the game is automatically reproduced from the separation point before and closest to the game interruption point to the game interruption point. Patent Literature 1: Japanese Patent No. 3731781.

DISCLOSURE OF INVENTION

Problem to be Solved by the Invention

[0003] However, in action games and shooting games, for example, once the game is interrupted, the player may not be able to immediately respond to a feel of manipulating a controller and a sense of following a flow of the game, often leading to a failure of the play immediately after resumption. It is difficult for the player to regain a feel and a sense of game before interruption only by reproducing the player’s past play after resuming the game, as the above conventional art.

[0004] The present invention has been made in order to solve such problems and has an objective to provide a game device, a game processing method, an information recording medium and a program that are preferable for the player to easily grasp a continuity of a game before and after interruption of the game.

Means to Solve the Problem

[0005] In order to attain the above objective, the following invention will be disclosed according to a principle of the present invention.

[0006] A game device according to a first aspect of the present invention can interrupt and resume a game, and includes a state storage unit, a history storage unit, an operation reception unit, an update unit and a resumption control unit.

[0007] The state storage unit stores a current state of the game.

[0008] The history storage unit stores a history of the game state stored in the state storage unit while the game is not interrupted.

[0009] The operation reception unit receives an operation instruction from a player while the game is not interrupted.

[0010] The update unit updates the game state stored in the state storage unit on the basis of the received operation instruction.

[0011] If the game is resumed after interruption, the resumption control unit resumes the game from prior to a predetermined period before interruption of the game.

[0012] After resumption of the game followed by the predetermined period, if it is determined that the game state based on the operation instruction received during the predetermined period is better than the game state stored in the state storage unit on the basis of a predetermined determination criterion, the resumption control unit further updates the game state stored in the state storage unit with the game state based on the operation instruction received during the predetermined period.

[0013] In the game played in the game device according to the present invention, the player can not only input a command but also interrupt the game at any timing after start of the game and resume the game at any timing after interruption. The player can input an instruction to interrupt or resume the game. The game device may interrupt the game without an input from the player, for example, in a case where there is no input for a predetermined period or more.

[0014] A predetermined game parameter is internally defined in the game, and the game device changes the game parameter according to an operation input from the player and progresses the game. The game parameter includes, for example, a score acquired by the player, an experience value and physical strength value of a game character, a time set within a game (in-game time) and the like. The game device stores a current game parameter and a history of a game parameter. These game parameter and history are updated as needed.

[0015] When the game device resumes the game after interruption, it resumes the game from a predetermined period before in-game interruption time. The predetermined period is defined by a fixed value, such as 5 seconds or 10 seconds. Alternatively, the predetermined period may be a variable value that can be obtained as a good timing to stop the game, such as a period from the latest timing of the timings when a ball goes over a side line and the play stops in a soccer game to a current time. The present invention does not limit a content of the game.

[0016] If the game is resumed from the predetermined period before interruption, the player can input a new command during this predetermined period. A play history after resumption and a play history before interruption both are retained. When the predetermined period has passed since resumption, in other words, when in-game time after resumption reaches in-game interruption time again, a history of a game parameter before interruption and a history of a game parameter after resumption are evaluated. As the result of evaluation, one of these histories becomes a game result of the player. For example, of the previous play history and current play history, the play history with a better result becomes a final play content.

[0017] Typically, data, such as a game score acquired by the player, a remaining time of a game, an elapsed time of a game, a rank in e.g., a race game, a hit probability or the number of hits in e.g., a shooting game, are used as an evaluation criterion. These data is properly processed with the use of an arbitrary function and can be used as the evaluation criterion. Whether the result is better or not is a relative determination on the basis of the evaluation criterion.

[0018] For example, in the case where “an acquired score is high” is an evaluation criterion, if a previous (before interruption) acquired score is higher than a current (after resump-
acquired score during the predetermined period, the previous acquired score with a better result is adopted and the game continues on the basis of the previous acquired score. A history of the game parameter is rewritten with (returned to) the history of the previous game parameter. For example, if the previous acquired score is lower than the current acquired score, the current acquired score with a better result is adopted, and the game continues on the basis of the current acquired score.

For example, in the case where “an acquired score is low” is an evaluation criterion, if the previous (before interruption) acquired score is lower than the current (after resumption) acquired score during the predetermined period, the previous acquired score with a worse result is adopted and the game continues on the basis of the previous acquired score. A history of the game parameter is rewritten with (returned to) the history of the previous game parameter. For example, if the previous acquired score is higher than the current acquired score, the current acquired score with a worse result is adopted, and the game continues on the basis of the current acquired score.

According to the present invention, the player not only can review a past game process but also use the predetermined period as a practice period for regaining a sense of game in resuming the game. Even after interruption, the player can immediately grasp a continuity of the game and easily join the game.

The game device may further include a display unit to display a state image based on a game state stored in the storage unit.

The state image is a game image generated using a current game parameter, i.e., a real-time game image. A monitor connected to the device (or a monitor the device includes) displays a game image that changes according to an operation instruction inputted from the player. The player can grasp a content of an instruction inputted by himself/herself, information about how the game state changes by the instruction, or information about the his/her game result.

If the game is resumed after interruption, the display unit may obtain a history of a game state within the predetermined period of histories of a game state stored in the storage unit, and further display history images based on the obtained history of a game state in sequence.

The history image is a game image generated using a history of a game parameter and reproduces a past play. The monitor displays a current game image, as well as a game image before interruption. That is, the player can compare his/her play before interruption with his/her play after resumption. For example, in a music game in which a music instrument is played according to displayed musical notes, a degree of proficiency can be grasped by comparing a previous performance with a current performance.

The display unit may display the state image and the history image such that the state image is more relatively highlighted than the history image.

The monitor displays a current game image after resumption bigger than a past game image before interruption, for example. The current game image after resumption may be displayed in full-screen and the past game image before interruption may be displayed in a small window, for example. Accordingly, the player can check his/her past play without interrupting his/her play after resumption.

The game state may be indicated using a numerical parameter that changes according to a received operation instruction. The predetermined evaluation criterion may be defined using the numerical parameter when the predetermined period has passed since resumption of the game, and the numerical parameter when the game was interrupted.

The numerical parameter is a score acquired by the player, for example. When the game is resumed after interruption, a score at an in-game interruption is compared with a score when a resumed game reaches the same in-game time. Then, a history of a game parameter corresponding to a higher score of those scores is used for determining a game result of the player.

According to the present invention, the player not only can review a past game process but also use the predetermined period as a practice period for regaining a sense of game in resuming the game. Even after interruption, the player can immediately grasp a continuity of the game and easily join the game. A history of a game parameter corresponding to a lower score of those scores may be used to determine a result of the player.

The game state may be indicated using a numerical parameter that changes according to the received operation instruction. The predetermined evaluation criterion may be defined using a history of the numerical parameter for the predetermined period after resumption of the game, and a history of the numerical parameter for the predetermined period before interruption.

When the game is resumed after interruption, a history of a game parameter during a past predetermined period before in-game interruption time is compared with a history of a game parameter from resumption of the game to the same in-game time of that interruption. For example, a history obtained by combining portions of histories with a better result is used to determine the result of the player. That is, only portions with a better performance of two plays are picked up and used to determine the result of the player. Even after interruption, the player can warm up for some time to regain a sense of game, grasp a continuity of the game and easily join the game. Alternatively, only portions with a worse performance of two plays may be picked up and used to determine the result of the player.

A game processing method according to another aspect of the present invention is performed by a game device. The game device having a state storage unit, a history storage unit, an operation reception unit, an update unit and a resumption control unit and being able to interrupt and resume a game, the game processing method including an operation reception step, an update step and a resumption control step.

The state storage unit stores a current state of the game.

The history storage unit stores a history of the game state stored in the state storage unit while the game is not interrupted.

In the operation reception step, the operation reception unit receives an operation instruction from a player while the game is not interrupted.

In the update step, the update unit updates the game state stored in the state storage unit on the basis of the received operation instruction.

In the resumption control step, the resumption control unit, if the game is resumed after the interruption, resumes the game from prior to a predetermined period before interruption of the game.
Further, in the resumption control step, after resumption of the game followed by the predetermined period, if it is determined that the game state based on the operation instruction received during the predetermined period is better than the game state stored in the state storage unit on the basis of a predetermined evaluation criterion, the resumption control unit updates the game state stored in the state storage unit with the game state based on the operation instruction received during the predetermined period.

According to the present invention, the player not only can review a past game process but also use the predetermined period as a practice period for regaining a sense of game in resuming the game. Even after interruption, the player can immediately grasp a continuity of the game and easily join the game.

An information recording medium according to another aspect of the present invention stores a program that makes, a computer being able to interrupt and resume a game, function as a state storage unit, a history storage unit, an operation reception unit, an update unit and a resumption control unit.

The state storage unit stores a current state of the game.

The history storage unit stores a history of the game state stored in the state storage unit while the game is not interrupted.

The operation reception unit receives an operation instruction from a player while the game is not interrupted.

The update unit updates the game state stored in the state storage unit on the basis of the received operation instruction.

The resumption control unit, if the game is resumed after the interruption, resumes the game from prior to a predetermined period before interruption of the game.

Further, after resumption of the game followed by the predetermined period, if it is determined that the game state based on the operation instruction received during the predetermined period is better than the game state stored in the state storage unit on the basis of a predetermined evaluation criterion, the resumption control unit updates the game state stored in the state storage unit with the game state based on the operation instruction received during the predetermined period.

The present invention can make a computer function as a game device operating as above.

A program of the present invention can be recorded on a computer-readable information storage medium, such as a compact disc, a flexible disc, a hard disk, a magneto-optical disk, a digital video disc, a magnetic tape, or a semiconductor memory.

The above program can be distributed and sold, independently of the computer on which the program is executed, via a computer communication network. In addition, the above information storage medium can be distributed and sold independently of the computer.

Effect of the Invention

The present invention can provide a game device, a game processing method, an information recording medium and a program that are preferable for a player to easily grasp a continuity of a game before and after interruption of the game.

BRIEF DESCRIPTION OF DRAWINGS

FIG. 1 is a diagram illustrating a general configuration of a typical information processing device in which a game device according to the present invention can be realized.

FIG. 2 is a diagram for explaining a functional configuration of a game device.

FIG. 3 is a diagram for explaining a configuration of a controller.

FIG. 4 is a diagram illustrating a configuration example of a game screen.

FIG. 5A is a diagram representing a change of a game parameter by an operation input before interruption and a change of a game parameter by an operation input after resumption.

FIG. 5B is a diagram representing a change of a game parameter used for evaluation of a result of a player.

FIG. 5C is a diagram representing a change of a game parameter used for evaluation of a result of a player.

FIG. 6 is a flow chart for explaining a game processing.

FIG. 7 is a flow chart for explaining a game processing (continued from FIG. 6).

FIG. 8 is a diagram illustrating a history of a game parameter stored in a history storage unit at the time of interruption of a game.

FIG. 9 is a diagram illustrating a history of a game parameter stored in a history storage unit after interruption and before resumption of a game.

FIG. 10 is a diagram illustrating a history of a game parameter stored in a history storage unit after a predetermined period after resumption of a game.

FIG. 11A is a diagram illustrating a history of a game parameter stored in a history storage unit after having
compared a game parameter before interruption of a game with a game parameter after resumption of the game.

[0072] FIG. 11B is a diagram illustrating a history of a game parameter stored in a history storage unit after having compared a game parameter before interruption of a game with a game parameter after resumption of the game.

[0073] FIG. 12 is a diagram illustrating a screen example while a game is not interrupted.

[0074] FIG. 13 is a diagram illustrating a screen example while a game is interrupted.

[0075] FIG. 14 is a diagram illustrating a screen example when a game is resumed after interruption.

[0076] FIG. 15 is a diagram illustrating a screen example within a predetermined period after interruption of a game.

[0077] FIG. 16 is a diagram illustrating another screen example within a predetermined period after interruption of a game.

[0078] FIG. 17A is a diagram representing a change of a game parameter by an operation input before interruption and a change of a game parameter by an operation input after resumption.

[0079] FIG. 17B is a diagram representing a change of a game parameter used for evaluation of a result of a player.

[0080] FIG. 17C is a diagram representing a change of a game parameter used for evaluation of a result of a player.

[0081] FIG. 18A is a diagram representing a change of a game parameter by an operation input before interruption and a change of a game parameter by an operation input after resumption.

[0082] FIG. 18B is a diagram representing a change of a game parameter used for evaluation of a result of a player.

[0083] FIG. 19 is a diagram illustrating a screen example while a game is interrupted.

BEST MODE FOR CARRYING OUT THE INVENTION

Embodiment 1

[0084] Embodiments of the present invention will be described. In order to ease understanding of the following, embodiments in which the present invention is applied to an information processing device for a game will be described. However, the following embodiments are only for explanation and do not limit the scope of the invention of the present application. Accordingly, it is possible for those skilled in the art to adopt embodiments wherein some or all of the elements herein have been replaced with equivalent respective elements. These embodiments are also included within the scope of the present invention.

[0085] FIG. 1 is a schematic diagram illustrating a general configuration of a typical information processing device 100 that serves as a game device of the present invention by executing a program. Hereinafter, an embodiment will be described with reference to FIG. 1.

[0086] The information processing device 100 is provided with a CPU (Central Processing Unit) 101, a ROM (Read Only Memory) 102, a RAM (Random Access Memory) 103, an interface 104, a controller 105, an external memory 106, a DVD-ROM (Digital Versatile Disk—Read Only Memory) drive 107, an image processing unit 108, a sound processing unit 109, and an NIC (Network Interface Card) 110.

[0087] By inserting a DVD-ROM storing a program and data for a game into the DVD-ROM drive 107 and powering on the information processing device 100, the program is executed and the game device according to the present embodiment is realized.

[0088] The CPU 101 controls the entire operation of the information processing device 100, and is connected to individual components for exchanging control signals and data. The CPU 101 can use an ALU (Arithmetic Logic Unit) (not shown) to perform an arithmetic operation such as four operations, a logical operation such as logical addition, logical multiplication and logical negation, and a bit operation such as bit addition, bit multiplication, bit inversion, bit shift and bit rotation, for a high-speed accessible storage area called a register (not shown). Moreover, the CPU 101 may be configured to perform saturate calculation such as four arithmetical operations for dealing with multimedia processing, and a vector operation such as trigonometric function at high speed, or may be realized with a coprocessor.

[0089] Recorded in the ROM 102 is an IPL (Initial Program Loader), which is executed immediately after power-on. By executing the IPL, the program stored in the DVD-ROM is read into the RAM 103, and the execution of the program by the CPU 101 is commenced. The ROM 102 also stores programs and various data for an operating system necessary for controlling the overall operation of the information processing device 100.

[0090] The RAM 103 is for temporarily storing data and programs, and retains programs and data read out from the DVD-ROM, as well as data necessary for other operations, such as game progress or chat communication. The CPU 101 provides the RAM 103 with a variable area and performs processing as follows: the CPU 101 makes the ALU directly carry out an operation on a value stored in the variable area, or the CPU 101 stores a value stored in the RAM 103 to the register and then carries out an operation on the register; then, the CPU 101 writes a result of the operation on the memory.

[0091] The controller 105, connected via the interface 104, receives an operation input by a player in playing a game such as a dance game or a soccer game. To the interface 104, a plurality of controllers 105 may be connected.

[0092] The external memory 106, being detachably connected via the interface 104, stores data such as data indicating a game play status (e.g., past result) or game progress, or chat logs (records) of a game using a network, such data being stored in a rewritable manner. By performing an operation input via the controller 105, the player can appropriately record such data on the external memory 106.

[0093] Recorded on the DVD-ROM loaded into The DVD-ROM drive 107 is the program for realizing the game, as well as image data and sound data associated with the game. Under the control of the CPU 101, the DVD-ROM drive 107 performs processing to read out the necessary program and data from the DVD-ROM loaded therein. The read-out information is then temporarily stored in the RAM 103 or similar memory.

[0094] The image processing unit 108 processes data read from the DVD-ROM by using the CPU 101 and/or an image operation processor (not shown) provided in the image processing unit 108, and then records the resulting data in a frame memory (not shown) provided in the image processing unit 108. The image information recorded in the frame memory is converted into a video signal at a predetermined synchronous timing, and subsequently outputted to a monitor (not shown) connected to the image processing unit 108. This enables the display of various images.
The image operation processor is able to execute a transparency operation such as overlaying two-dimensional images or a blending, as well as various saturate calculations at high speed.

In addition, the image operation processor is also able to perform a high-speed operation whereby polygon information, being disposed in a virtual three-dimensional space and having various types of texture added thereto, is rendered by means of z-buffering, thereby obtaining a rendered image showing a view of polygons disposed in the virtual three-dimensional space in a predetermined point of view.

Furthermore, the CPU 101 and the image operation processor operating in cooperation, it is possible to draw a string of characters such as a two-dimensional image in the frame memory or on respective polygon surfaces, according to font information that defines the shapes of the characters.

By previously storing information such as image files in the DVD-ROM and developing these images in the frame memory, a game state and so on can be displayed on the screen.

The sound processing unit 109 converts audio data read out from the DVD-ROM into an analog audio signal, and then causes the audio signal to be outputted from a speaker (not shown) connected thereto. Under control of the CPU 101, the sound processing unit 109 also generates sound effects and music data that should be emitted as the game progresses, and then causes the speaker to output sounds corresponding thereto.

In the sound processing unit 109, if the audio data recorded in the DVD-ROM is MIDI data, the MIDI data is converted into PCM data by referring to audio source data the MIDI data has. If the audio data recorded in the DVD-ROM is compressed audio data such as ADPCM (Adaptive Differential Pulse Code Modulation) system or Ogg Vorbis system, this is extracted to be converted into PCM data. The PCM data is subjected to D/A (Digital/Analog) conversion at a timing corresponding to its sampling frequency, and then outputted to a speaker, thereby permitting for audio output.

The NIC 110 is for connecting the information processing device 100 to a computer communication network (not shown) such as the Internet. The NIC 110 may conform to the 10BASE-T/100BASE-T standard used when forming LAN (Local Area Network), or alternatively, the NIC 110 may be composed of a modem, such as an analog modem for connecting to the Internet using a telephone circuit, an ISDN (Integrated Services Digital Network) modem, an ADSL (Asymmetric Digital Subscriber Line) modem, or a cable modem for connecting to the Internet using a cable television circuit, as well as an interface (not shown) that interfaces between the CPU 101 and any one of the above modems.

In addition, the information processing device 100 may also be configured to use a large-capacity external storage device such as a hard disk to perform the same function as a component such as the ROM 102, the RAM 103, the external memory 106, or the DVD-ROM loaded into the DVD-ROM drive 107.

Next will be described a functional configuration of a game device 200 according to the present embodiment that is realized by the information processing device 100 having the above configuration.

FIG. 2 is a diagram explaining a functional configuration of the game device 200. The game device 200 is provided with a state storage unit 201, a history storage unit 202, an operation reception unit 203, an update unit 204, a display unit 205 and a resumption control unit 206.

FIG. 3 is a schematic diagram for explaining a configuration of the controller 105 used with being connected to the game device 200 according to the present embodiment. The controller 105 is a mat-type controller disposed on a floor and so on. The controller 105 has a left button 301, a bottom button 302, a top button 303 and a right button 304 disposed thereon. An instruction is generally inputted to the controller 105 by a player stepping on. Alternatively, the player may input an instruction by pushing the respective buttons with hand.

FIG. 4 is a configuration example of a screen 400 of a game performed in the game device 200 according to the present embodiment. The screen 400 displays stationary marks 401 to 404 immovable drawn at predetermined positions in the screen 400, step position indication marks 410 (four types in FIG. 4: 410A, 410B, 410C, 410D) whose drawn positions move as time passes, a score 420 acquired by the player in the game, a gauge 430 indicating a game result of the player, and a background image.

First, will be described a game played on the game device 200 according to the present embodiment. In this game, music is played back by the game device 200. The player can dance to the played music by inputting an instruction according to indication marks called foot musical notes (hereinafter also referred to as “task”).

The step position indication marks 410 are scroll-displayed according to the played music. The step position indication marks 410 are drawn as arrows corresponding to the buttons 301 to 304, any of the arrows being an upward arrow, downward arrow, leftward arrow, or rightward arrow.

The stationary marks 401 to 404 indicate timings (hereinafter also referred to as “task time”) for the player to push the buttons 301 to 304, respectively. The stationary marks 401 to 404 are drawn as arrows, any of the arrows being an upward arrow, downward arrow, leftward arrow, or rightward arrow.

The step position indication marks 410 move toward the positions where the stationary marks 401 to 404 are drawn according to the play speed of the music. When the step position indication marks 410 move to the same positions of those of the stationary marks 401 to 404, the player pushes any of the buttons 301 to 304 corresponding to the directions of arrows of the stationary marks 401 to 404. Then, predetermined points will be added to the score 420 of the player, and/or a value (dance meter) indicated by the gauge 430 will increase.

When the player achieves a predetermined task at a task time previously associated with the task, the CPU 101 increases a value indicated by the gauge 430 and/or increases a value indicated by the score 420, and evaluates that the game result is good. The CPU 101 also evaluates that the closer to the task time, the time when the task is achieved is, the better the game result is.

For example, when the step position indication marks 410 move to a position overlapping any of the stationary marks 401 to 404, the player steps with one foot on the button (any of the buttons 301 to 304) corresponding to an arrow of the step position indication mark 410 that has moved, whereby the player can perform a model dance step to the music being played back.

The CPU 101 can interrupt (pause) the game and resume the game after interruption by an instruction from the
player. For example, the player can interrupt or resume the game at any timing by pushing a predetermined pause button. Hereinafter, an instruction to interrupt the game will be referred to as an “interruption instruction”, and an instruction to resume the interrupted game will be referred to as a “resumption instruction”.

[0114] The CPU 101 may interrupt or resume the game depending on whether or not a predetermined condition is fulfilled, regardless of with or without an interruption instruction or a resumption instruction from the player. For example, the CPU 101 may interrupt the game if the controller 105 does not receive an input from the player for a predetermined time period or more. In addition, for example, if the CPU 101 receives a resumption instruction from the player, or if a predetermined pause time period has passed after interruption of the game, the CPU 101 may resume the game.

[0115] Next, a functional configuration of the game device 200 will be described in details.

[0116] The state storage unit 201 stores a current game state that is being played in the game device 200. The CPU 101 and RAM 103 work together to serve as the state storage unit 201.

[0117] The game state is a game parameter indicating, for example, a value of the score 420 acquired by the player (score), a value of the gauge 430 (dance meter), an elapsed time in the game (in-game time), or a real elapsed time. The CPU 101 updates respective values of various game parameters previously set in the game, as the game progresses, and stores the updated values in a predetermined memory area in the RAM 103.

[0118] The history storage unit 202 stores a history of the game state stored in the state storage unit 201 while the game is not interrupted. The CPU 101 and RAM 103 work together to serve as the history storage unit 202.

[0119] Specifically, the CPU 101 makes the RAM 103 store a past game state as the history. The CPU 101 stores, for example, a history of a game state during a real-time past period (such as “from 5 seconds ago to the present time”). Alternatively, the CPU 101 may store a history of a game state during an in-game past period (such as “from the start of the first period to the present time”) in an ice hockey game.

[0120] For example, the CPU 101 defines array variables to store a history of a game parameter for a predetermined in-game time period, stores the array variables in the RAM 103 and periodically updates respective values of the array variables so as to store a history of a game parameter from a predetermined in-game time period before the present time until the present time. An update timing is typically a timing when a vertical synchronization (VSYNC) interruption occurs.

[0121] The operation reception unit 203 receives an operation instruction from the player while the game is not interrupted. The operation instruction is an instruction other than an interruption instruction and a resumption instruction, such as an input instruction of a command of the game played in the game device 200. The CPU 101 and controller 105 work together to serve as the operation reception unit 203.

[0122] The command is previously defined by pattern data composed of data strings indicating one or more operations to be input by the player. For example, the command is indicated by pattern data such as “left button once” in the controller 105 having the buttons 301 to 304. At this time, when the player presses the button 301 once, a game event previously associated with this pattern data (for example, a music associated with the button 301 is played back) occurs and the game progresses.

[0123] According to the present embodiment, the CPU 101 receives an operation instruction and an interruption instruction while the game is progressing (is not interrupted). Meanwhile, the CPU 101 receives a resumption instruction while the game is not progressing (is interrupted).

[0124] The update unit 204 updates the current game state stored in the state storage unit 201 on the basis of an operation instruction received from the player. The CPU 101 and RAM 103 work together to serve as the update unit 204.

[0125] That is, when the player inputs a command, the CPU 101 changes respective values of game parameters according to the inputted command, stores the changed game parameters in the RAM 103, and progresses the game.

[0126] The display unit 205 generates a state image 251 based on the game state stored in the state storage unit 201, and displays the state image 251 on the monitor. When the game is resumed after interruption, the display unit 205 can obtain a history of a game state contained within the latest predetermined period, of histories of a game state stored in the history storage unit 202, generate history images 252 based on the game state contained in the obtained history of a game state, and display the history images 252 on the monitor in sequence. The CPU 101, RAM 103 and image processing unit 108 work together to serve as the display unit 205.

[0127] Images displayed on the monitor are classified into two types: the state image 251 and the history image 252.

[0128] The state image 251 is a game image generated by the CPU 101 and the like on the basis of a current game parameter, that is, a game parameter stored in the state storage unit 201. In other words, the state image 251 is a real-time image changing according to an operation instruction inputted from the player.

[0129] The history image 252 is a game image generated by the CPU 101 and the like on the basis of a past game parameter, that is, a history of a game parameter stored in the history storage unit 202. In other words, the history image 252 is a replay image reproduced by using a history of an operation instruction inputted from the player in the past.

[0130] The CPU 101 controls the image processing unit 108 to generate the state image 251 and/or the history image 252 every VSYNC. Then, the CPU 101 makes the state image 251 and/or the history image 252 to be displayed on the monitor. The CPU 101 may make both of the state image 251 and history image 252 to be generated and displayed, or may make either one of the state image 251 and history image 252 to be generated and displayed.

[0131] When the resumption control unit 206 resumes the game after interruption, it resumes the game from the predetermined period before the interruption. That is, when the player inputs an interruption instruction to interrupt the game and then inputs a resumption instruction, the game is resumed from the predetermined period before the in-game interruption time. After the game is interrupted, the game is resumed from a little before the interrupted scene. When the game is resumed after interruption, the player can play twice during this predetermined period.

[0132] When the predetermined period has passed since resumption of the game, the resumption control unit 206 evaluates whether or not a game state based on an operation instruction received during the predetermined period is better
than a game state stored in the state storage unit 201 on the basis of a predetermined evaluation criterion.

[0133] Herein, “a game state is better” means better in terms of the predetermined evaluation criterion. For example, if “a player has a higher score value” is the evaluation criterion, a higher score value indicates a better result. For example, if “a player has a lower score value” is the evaluation criterion, a lower score value indicates a better result.

[0134] If the resumption control unit 206 determines that the game state based on an operation instruction received during the predetermined period is better than the game state stored in the state storage unit 201, the resumption control unit 206 updates the game state stored in the state storage unit 201 with the game state based on an operation instruction received during the predetermined period.

[0135] That is, the game is resumed from the predetermined period before interruption, and when the predetermined period has passed, the game result based on the past operation instruction inputted by the player before interruption is compared with the game result based on the operation instruction newly inputted by the player after resumption. The game result is an evaluation value derived by a predetermined computation expression with the use of a game parameter. Typically, this evaluation value is a value indicated by the score 420 or a value indicated by the gauge 430 that is able to increase or decrease as the game progresses. Then, a final game result of the player is decided according to the result of comparison, as the following cases.

[0136] (Case 1) Case where the game result based on an operation instruction newly inputted after resumption is better than the game result based on an operation instruction previously inputted before interruption when the predetermined period has passed since the resumption:

[0137] The CPU 101 adopts the game result based on the operation instruction newly inputted after resumption, and discards the game result based on the operation instruction previously inputted before interruption. The CPU 101 adopts the game result based on the operation instruction newly inputted after resumption as the game result of the player during the predetermined period.

[0138] (Case 2) Case where the game result based on the operation instruction newly inputted after resumption is worse than or equal to the game result based on the operation instruction previously inputted before interruption when the predetermined period has passed since the resumption:

[0139] The CPU 101 adopts the game result based on the operation instruction previously inputted before interruption and discards the game result based on the operation instruction newly inputted after resumption. The CPU 101 adopts the game result based on the operation instruction previously inputted before interruption as the game result of the player during the predetermined period.

[0140] That is, of the previously inputted operation instruction and the newly inputted operation instruction, the operation instruction with a better result when the predetermined period has passed since resumption is adopted as the game result of the player.

[0141] The case 1 will be described.

[0142] FIG. 5A is a diagram representing a change with time of a game parameter (for example, a value of the gauge 430 (dance meter)) by an operation input before interruption and a change with time of the game parameter by an operation input after resumption. At in-game time T1, an interruption instruction is inputted, and then when a resumption instruction is inputted, the CPU 101 resumes the game from in-game time T2 that is a predetermined period TP before the in-game time T1.

[0143] Suppose, for example, a curve 510 indicates a change of the dance meter by an operation input after resumption, and a curve 520 indicates a change of the dance meter by an operation input before interruption. When a type of a button pushed by the player and a timing of the push are close to a predetermined task content and task time, a value indicated by the dance meter increases. On the contrary, a type of a button pushed by the player and a timing of the push are away from a predetermined task content and task time, a value indicated by the dance meter decreases. The CPU 101 evaluates that the higher value the dance meter indicates, the better the game result is.

[0144] At in-game time T1 when the game is interrupted, the CPU 101 compares a value VB of the dance meter by an operation input before interruption with a value VA of the dance meter by an operation input after resumption. In this example, since “VA (after resumption) > VB (before interruption)”, the CPU 101 adopts the curve 510 as a change of the dance meter to be used for a game result from in-game time T2 to in-game time T1. As a result, the game result of the player is evaluated based on a curve 530 indicated by FIG. 5B, that is, based on the change of the dance meter by the (second) operation input after resumption that has a better result. Then, the CPU 101 will calculate a value of the dance meter after in-game time T1, using a point 515 as a start point.

[0145] Next, the Case 2 will be described.

[0146] Suppose, for example, the curve 510 indicates a change of the dance meter by an operation input before interruption and the curve 520 indicates a change of the dance meter by an operation input after resumption.

[0147] At in-game time T1 when the game is interrupted, the CPU 101 compares the value VA of the dance meter by an operation input before interruption with the value VB of the dance meter by an operation input after resumption. In this example, since “VA (before interruption) > VB (after resumption)”, the CPU 101 adopts the curve 510 as a change of the dance meter to be used for the game result from in-game time T2 to in-game time T1. As a result, the game result of the player is evaluated based on the curve 530 indicated by FIG. 5B, that is, based on the change of the dance meter by the (first) operation input before interruption that has a better result. Then, the CPU 101 will calculate a value of the dance meter after in-game time T1, using the point 515 as a start point.

[0148] Alternatively, the CPU 101 may evaluate such that a smaller value indicated by the dance meter indicates a better result. That is, the CPU 101 may decide a final game result of the player as following Case 1’ and Case 2’, instead of the above Case 1 and Case 2.

[0149] (Case 1’) Case where the game result based on an operation instruction newly inputted after resumption is better than the game result based on an operation instruction previously inputted before interruption when the predetermined period has passed since the resumption:

[0150] The CPU 101 adopts the game result based on an operation instruction previously inputted before interruption, and discards the game result based on an operation instruction newly inputted after resumption. The CPU 101 adopts the game result based on an operation instruction previously inputted before interruption as the game result of the player during the predetermined period.
[0151] Suppose, for example, the curve 510 indicates a change of the dance meter by an operation input after resumption, and the curve 520 indicates a change of the dance meter by an operation input before interruption. At in-game time T1 when the game is interrupted, the CPU 101 compares the value VB of the dance meter by an operation input before interruption with the value VA of the dance meter by an operation input after resumption. Since “VA (after resumption) > VB (before resumption)”, the game result of the player is evaluated based on a curve 540 indicated by FIG. 5C, that is, based on the change of the dance meter by the (first) operation input before interruption that has a worse result. Then, the CPU 101 will calculate a value of the dance meter after in-game time T1, using a point 525 as a start point.

[0152] (Case 2) Case where the game result based on an operation instruction newly inputted after resumption is worse than or equal to the game result based on an operation instruction previously inputted before interruption when the predetermined period has passed since the resumption:

[0153] The CPU 101 adopts the game result based on an operation instruction newly inputted after resumption and discards the game result based on an operation instruction previously inputted before interruption. The CPU 101 adopts the game result based on an operation instruction newly inputted after resumption as the game result of the player during the predetermined period.

[0154] Suppose, for example, the curve 510 indicates a change of the dance meter by an operation input before interruption and the curve 520 indicates a change of the dance meter by an operation input after resumption. At in-game time T1 when the game is interrupted, the CPU 101 compares the value VA of the dance meter by an operation input before interruption with the value VB of the dance meter by an operation input after resumption. In this example, since “VA (before interruption) > VB (after resumption)”, the game result of the player is evaluated based on the curve 540 indicated by FIG. 5C, that is, based on the change of the dance meter by the (second) operation input after resumption that has a worse result. Then, the CPU 101 will calculate a value of the dance meter after in-game time T1, using the point 525 as a start point.

[0155] In the aforementioned Case 2 and Case 2’, if after the predetermined period has passed, the game result based on an operation instruction newly inputted after resumption is equal to the game result based on an operation instruction previously inputted before interruption, the CPU 101 may adopt either of the results.

[0156] Next, game processing performed by the aforementioned units in the game device 200 will be described, using a flow chart and so on. In the following description, “a value indicated by a game parameter is higher” is an evaluation criterion.

[0157] FIGS. 6 and 7 are flow charts illustrating game processing according to the present embodiment. FIGS. 8, 9, 10, 11A and 11B are diagrams for explaining a history of a game parameter stored in the history storage unit 202.

[0158] First, the CPU 101 starts the game, receives an operation instruction from the player to progress the game, and stores a current game parameter in the RAM 103 (step S601). The current game parameter is stored in the state storage unit 201.

[0159] The CPU 101 stores a history of a game parameter in the RAM 103 (step S602). The history of a game parameter is stored in the history storage unit 202.

[0160] That is, the CPU 101 obtains a game parameter every predetermined time period, and stores at least the latest N number of times of game parameters (N is an integer greater or equal to one) of the obtained game parameters, as a history, in the RAM 103.

[0161] For example, FIG. 8 illustrates a state where game parameters obtained from the start of the game are stored in the order of obtaining (in the order of time) in the history storage unit 202. A value of the game parameter at in-game time T1 is VA

[0162] The CPU 101 controls the image processing unit 108 to generate the state image 251 and to display the state image 251 on the monitor.

[0163] Next, the CPU 101 determines whether or not an interruption instruction of the game was inputted (step S603).

[0164] If it is determined that an interruption instruction was not inputted (step S603; NO), the CPU 101 repeats processing from steps S601 to S603. That is, the game is not interrupted and continues to progress. By repeating processing, respective generated state images 251 are displayed in series, which look like a dynamic image.

[0165] If it is determined that an interruption instruction was inputted (step S603; YES), the CPU 101 temporarily stops the game, and temporarily saves a history of a game parameter for the predetermined period in a buffer memory 900 (step S604). The buffer memory 900 is a memory area provided within a predetermined area of the RAM 103.

[0166] For example, if there is an interruption instruction at in-game time T1 illustrated in FIG. 8, the CPU 101 copies the latest M (M is an integer greater or equal to one) pieces of histories V (N-M+1) to V(N) for the predetermined period into the buffer memory 900, as illustrated in FIG. 9.

[0167] While the game is interrupted, the CPU 101 controls the image processing unit 108 to display the last-generated state image 251 and a message of interruption (during a pause) and the like on the monitor.

[0168] Next, the CPU 101 determines whether or not a resumption instruction was inputted (step S605).

[0169] If it is determined that a resumption instruction has not been inputted (step S605; NO), the CPU 101 waits until a resumption instruction is inputted.

[0170] If it is determined that a resumption instruction was inputted (step S605; YES), the CPU 101 resumes the game from in-game time T2 that is the predetermined period before in-game time T1 of interruption (step S606).

[0171] After resumption of the game, the CPU 101 receives an operation instruction from the player to progress the game, and stores a current game parameter in the RAM 103 (step S607).

[0172] The CPU 101 stores a history of a game parameter in the RAM 103 (step S608). At this time, the CPU 101 overwrites a history of a game parameter for in-game time T2 before interruption V(j) (j is an integer greater or equal to N-M+1 and less or equal to N) with a game parameter for in-game time T2 after resumption.

[0173] The CPU 101 controls the image processing unit 108 to generate the state image 251 and display the state image 251 on the monitor.

[0174] Until the aforementioned predetermined period has passed since resumption of the game, the CPU 101 may generate the history image 252 on the basis of a game parameter stored in the buffer memory 900 (that is, a past game parameter before interruption of the game) and display the history image 252 on the monitor.
[0175] The CPU 101 determines whether or not the predetermined period has passed since resumption of the game (step S609).

[0176] It is determined that the predetermined period has not passed yet (step S609; NO), the CPU 101 progresses the game until the predetermined period has passed. The game progress for this time period is the second-time game progress.

[0177] For example, FIG. 10 is a diagram illustrating a history of a game parameter that is stored in the history storage unit 202 when in-game time reaches T1 again after resumption of the game. A value of a game parameter at in-game time T1 is VB. A value of a game parameter by the first (before interruption) play is VA, and a value of a game parameter by the second (after resumption) play is VB.

[0178] If it is determined that the predetermined period has passed, that is, in-game time reaches T1 again (step S609; YES), the CPU 101 compares a history of a game parameter temporarily saved in the buffer memory 900 with a history of a game parameter from in-game time T2 to in-game time T1 after resumption (step S610).

[0179] As the result of the comparison, if it is determined that the game parameter after resumption (the second game parameter) is better than the game parameter before interruption (the first game parameter), that is, “VA<VB” (step S611; YES), the CPU 101 discards the history of a game parameter temporarily stored in the buffer memory 900 (step S612).

[0180] That is, as illustrated in FIG. 11A, since the game parameter after resumption is still stored in the history storage unit 202, this game parameter is used as the game result of the player.

[0181] Meanwhile, as the result of the comparison, if it is determined that the game parameter before interruption (the first game parameter) is better than the game parameter after resumption (the second game parameter), that is, “VA>VB” (step S611; NO), the CPU 101 updates the history of a game parameter stored in the history storage unit 202 with the history of a game parameter temporarily saved in the buffer memory 900 (step S613).

[0182] That is, as illustrated in FIG. 11B, since the game parameter stored in the history storage unit 202 returns to the game parameter before interruption, the game parameter before interruption is used as the result of the game.

[0183] In the present embodiment, if “VA>VB”; it is determined NO in step S611, but it may be determined YES.

[0184] After in-game time T1, the CPU 101 continues to progress the game. When a predetermined termination condition of the game is fulfilled, the CPU 101 terminates the game and uses the history of a game parameter stored in the history storage unit 202 to calculate the game result of the player. The predetermined termination condition is, for example, “a predetermined time limit has passed” or “a value of the gauge 430 becomes below a predetermined value”.

[0185] As described above, a final game result of the player is decided using better one of the result for the predetermined period before interruption and the result for the predetermined period after resumption. For example, even if the player has an urgent phone call while playing the game, pause the game, and then resumes the game, the game is resumed a little before the pause, not the pause and therefore the player easily grasp the continuity of the game before and after interruption. The present embodiment makes it easier for the player to grasp the continuity of the game before and after interruption. In addition, since a command can be given again from the predetermined period before interruption, the player can be prevented from losing a sense of game or forgetting content of the game due to interruption of the game.

[0186] If a game parameter with a better result, of a game parameter before interruption and a game parameter after resumption, is adopted as with the above Cases 1 and 2, the player uses the predetermined period after resumption as a practice period for regaining a sense of game and easily grasps the continuity of the game, thereby easily progressing the game even after interruption.

[0187] If a game parameter with a worse result, of a game parameter before interruption and a game parameter after resumption, is adopted as with the above Cases 1 and 2, the player must play the game with a constant tension to “achieve the better result”. Accordingly, the game device 200 can adjust the game so that the tension or challenge level is increased or a penalty is imposed, which also leads to improvement of the player’s technique.

[0188] Next, a configuration of the screen 400 will be described with reference to FIGS. 12 to 16.

[0189] FIG. 12 illustrates an example of the screen 400 displayed in the aforementioned steps S601 to S603. The screen 400 displays the stationary marks 401 to 404, score 420, gauge 430, as well as step position indication marks 1201 to 1203 and the like. In FIG. 12, only the three step position indication marks are displayed, which is merely an example, and the number and positions of the step position indication marks vary depending on how to set a task. The CPU 101 makes the step position indication marks 1201, 1202 and 1203 move toward positions of the stationary marks 403, 404 and 402, respectively, at a predetermined speed. In FIG. 12, the step position indication marks 1201, 1202 and 1203 correspond to the state image 251.

[0190] FIG. 13 illustrates an example of the screen 400 displayed in the above step S604. When receiving an interruption instruction, the CPU 101 stops moving of the step position indication marks 1201, 1202 and 1203, and makes a message (for example, “PAUSE”) or an image indicating that the game is interrupted to be displayed.

[0191] FIG. 14 illustrates an example of the screen 400 when the game is resumed in the above step S606. The step position indication marks 1401 and 1402 are those displayed at in-game time when an interruption instruction was received, respectively. These are illustrated only for explaining the present invention, but actually, these are preferably not displayed within the screen 400.

[0192] When receiving a resumption instruction, the CPU 101 makes the positions of the step position indication marks 1201, 1202 and 1203 to be returned by a distance ΔL to the positions where they were a predetermined period before in-game time of receiving the interruption instruction, and displayed there, and then resumes moving of the step position indication marks 1201, 1202 and 1203. The player resumes the game from the position returned by the distance ΔL from the positions where the game was interrupted.

[0193] If there is another task that is prior to (hereinafter referred to as “prior task”) a task corresponding to the step position indication marks 1201, 1202 and 1203 within the distance ΔL, the CPU 101 makes a step position indication mark 1403 corresponding to the prior task within the distance ΔL to be displayed. There may be a plurality of prior tasks or may be no priority tasks within the distance ΔL. The player will replay the prior task existing within the distance ΔL.
FIG. 14, the step position indication marks 1201, 1202 and 1403 correspond to the state image 251.

[0194] FIG. 15 illustrates an example of the screen 400 displayed in the above steps S606 to S609. FIG. 15 illustrates the screen 400 when the left button 301, which corresponds to the step position indication mark 1403 indicating the prior task, is pushed by the player at a task time corresponding to the prior task redisplayed (at a timing just when the stationary mark 401 and step position indication mark 1403 overlap to each other). The CPU 101 finds the player’s result regarding the step position indication mark 1403 and displays the result (for example, “GREAT”).

[0195] The CPU 101 also makes the history image 252 based on a history of a game parameter before interruption to be displayed. In FIG. 15, the history image 252 corresponds to an image 1501.

[0196] That is, during the predetermined period after resumption, the player can know a result of this time (after resumption), as well as a result of the last time (before interruption). The player is able to determine his/her degree of proficiency such as “I have improved from the last time”, or is able to work out a strategy such as “since this time is better than the last time, this strategy will be employed from next time”.

[0197] In FIG. 15, the CPU 101 may not make the history image 252 to be displayed.

[0198] FIG. 16 illustrates another example of the screen 400 displayed in the above steps S606 to S609. As illustrated in FIG. 16, the CPU 101 may make a game screen after resumptions and a game screen before interruption to be divided and displayed.

[0199] That is, the state image 251 is displayed as an entire game window and includes all of the stationary marks 401 to 404, the score 420 after resumption, the gauge 430 after resumption, the step position indication marks 1201, 1202 and 1403 and an evaluation result after resumption. Meanwhile, the history image 252 is displayed as an entire game window and includes all of the stationary marks 401 to 404, the score 420 before interruption, the gauge 430 before interruption, the step position indication marks 1201, 1202 and 1403 and an evaluation result before interruption. The player can compare the game screen of the last time with the game screen of this time.

[0200] For example, the CPU 101 may change an image display before and after interruption, as follows:

[0201] (X) before interruption and during interruption: a full-screen display of the state image 251.

[0202] (Y) within a predetermined period after resumption: the state image 251 and the history image 252 are split-screen displayed, and

[0203] (Z) a predetermined period after resumption, or later: a full-screen display of the state image 251.

[0204] In this (Y), the CPU 101 preferably makes the state image 251 and history image 252 to be displayed such that the state image 251 and history image 252 are easily differentiated by the player. Alternatively, the CPU 101 preferably makes the state image 251, which is an image of the game currently being played, to be displayed such that the state image is easier to be seen by the player.

[0205] For example, the CPU 101 makes the state image 251 and history image 252 to be displayed such that the state image 251 is larger than the history image 252 in size.

[0206] For example, the CPU 101 makes the state image 251 and history image 252 to be displayed such that the state image 251 is more highlighted than the history image 252. A method for highlighting includes an approach to use a difference in image sizes, or a difference in image color tones.

[0207] For example, the CPU 101 makes the state image 251 to be full-screen displayed and the history image 252 to be displayed in a small window (what is called “wipe”).

[0208] For example, the CPU 101 makes; in (X), the state image 251 to be full-screen displayed; in screen transition from (X) to (Y), the state image 251 to be displayed and the history image 252 gradually to be wiped in; in screen transition from (Y) to (Z), the state image 251 to be displayed and the history image 252 gradually to be wiped out; and in (Z), the state image 251 to be full-screen displayed.

[0209] By displaying the state image 251 and history image 252 together, the player can more intuitively grasp the difference between an evaluation result of this time and an evaluation result of the last time during the predetermined period after resumption.

Embodiment 2

[0210] Next, another embodiment of the present invention will be described. In the above embodiment, the game result of the player is evaluated on the basis of the difference between the past game parameter at interruption and the game parameter after a predetermined period after resumption. Meanwhile, the present embodiment is different from the above embodiment in that the game result of the player is evaluated on the basis of a history of a game parameter for a predetermined period after interruption and a history of a game parameter for the predetermined period after resumption.

[0211] In FIG. 10, the CPU 101 compares respective game parameters stored in the history storage unit 202 with respective game parameters stored in the buffer memory 900.

[0212] That is, the CPU 101 compares an array element V(N-M+1) of the game parameter stored in the history storage unit 202 with an array element V(N-M+1) of the game parameter stored in the buffer memory 900. The CPU 101 also compares, an array element V(N-M+2) of the history storage unit 202 with an array element V(N-M+2) of the buffer memory 900. Similarly, the CPU 101 compares all of array elements to each other.

[0213] As a result of the comparison, if it is determined that there is an array element that a game parameter stored in the buffer memory 900 is better than a game parameter stored in the history storage unit 202, the CPU 101 updates the game parameter stored in the history storage unit 202 with the better game parameter stored in the buffer memory 900.

[0214] FIG. 17A is a diagram illustrating a change with time of a game parameter (for example, a value of the dance meter) by an operation input before interruption and a change in the game parameter by an operation input after resumption. An interruption instruction is inputted at in-game time T3, and when a resumption instruction is inputted, the CPU 101 resumes the game from in-game time T4 that is the predetermined period TP before in-game time T3.

[0215] Suppose, for example, a curve 1710 indicates a change of the dance meter by an operation input before interruption, and a curve 1720 indicates a change of the dance meter by an operation input after resumption. The CPU 101 compares respective values of the dance meter by an operation input before interruption with respective values of the dance meter by an operation input after resumption from in-game time T4 to T3.
From in-game time T4 to T5, since a value of the dance meter indicated by the curve 1720 is greater than a value of the dance meter indicated by the curve 1710, the CPU 101 adopts the curve 1720. From in-game time T5 to T6, since a value of the dance meter indicated by the curve 1720 is smaller than a value of the dance meter indicated by the curve 1710, the CPU 101 adopts the curve 1710. From in-game time T6 to T7, since a value of the dance meter indicated by the curve 1720 is greater than a value of the dance meter indicated by the curve 1710, the CPU 101 adopts the curve 1720. From in-game time T7 to T3, since a value of the dance meter indicated by the curve 1720 is smaller than a value of the dance meter indicated by the curve 1710, the CPU 101 adopts the curve 1710.

As a result, the game result of the player is evaluated based on a curve 1730 in FIG. 17B. Then, the CPU 101 calculates a value of the dance meter after in-game time T3, using a point 1715 as a start point.

If a game parameter with a better result is adopted of a game parameter before interruption and a game parameter after resumption in this way, the player can use the predetermined period after resumption to practice for regaining a sense of game and easily grasp the continuity of the game thereby to easily progress the game after interruption.

Alternatively, as a result of evaluation, if it is determined that there is an array element that a game parameter stored in the buffer memory 900 is worse than a game parameter stored in the history storage unit 202, the CPU 101 may update the game parameter stored in the history storage unit 202 with the worse game parameter stored in the buffer memory 900.

From in-game time T4 to T5, since a value of the dance meter indicated by the curve 1720 is greater than a value of the dance meter indicated by the curve 1710, the CPU 101 may adopt the curve 1710.

From in-game time T5 to T6, since a value of the dance meter indicated by the curve 1720 is smaller than a value of the dance meter indicated by the curve 1710, the CPU 101 may adopt the curve 1720.

From in-game time T6 to T7, since a value of the dance meter indicated by the curve 1720 is greater than a value of the dance meter indicated by the curve 1710, the CPU 101 may adopt the curve 1720.

From in-game time T7 to T3, since a value of the dance meter indicated by the curve 1720 is smaller than a value of the dance meter indicated by the curve 1710, the CPU 101 adopts the curve 1720.

As a result, the game result of the player is evaluated based on a curve 1740 in FIG. 17C. Then, the CPU 101 calculates a value of the dance meter after in-game time T3 using a point 1725 as a start point.

If a game parameter with a worse result is adopted of a game parameter before interruption and a game parameter after resumption in this way, the player must play the game with a constant tension, and therefore the game device 200 can adjust such that the game device 200 can increase a game tension and its level, or impose a penalty.

Embodiment 3

Next, another embodiment of the present invention will be described. In the above embodiments, the game is resumed from the predetermined period before interruption, and a history with a better result, of a history before interruption and a history after resumption, is adopted as the result of the player. However, there is a possibility that a predetermined termination condition of the game may be fulfilled within the predetermined period after resumption. The present embodiment takes this possibility into consideration.

For example, in the dance game in the above embodiments, without any interruption or resumption, when a value of the gauge 430 becomes zero, the game is forcibly terminated. Meanwhile, in a dance game in the present embodiment, even if a value of the gauge 430 becomes zero within the predetermined period after resumption, a history of a game parameter before interruption is used to progress the game.

The present embodiment will be described in detail with reference to FIGS. 18A and 18B. FIG. 18A is a diagram indicating a change with time of a game parameter (for example, a value of the dance meter) by an operation input before interruption and a change with time of the game parameter by an operation input after resumption. At in-game time T8, an interruption instruction is inputted, and then when a resumption instruction is inputted, the CPU 101 resumes the game from in-game time T9 that is the predetermined period TP before in-game time T8. A curve 1810 indicates a change of the dance meter by an operation input before interruption, and a curve 1820 indicates a change of the dance meter by an operation input after resumption.

If, at in-game time T10 during the predetermined period TP after resumption, the predetermined termination condition of the game is fulfilled, the CPU 101 adopts the curve 1810 as the change of the dance meter to be used for the game result from in-game time T9 to in-game time T8.

That is, the CPU 101 updates a history of a game parameter stored in the history storage unit 202 with a history of a game parameter temporarily saved in the buffer memory 900.

As a result, the game result of the player is calculated based on a curve 1830 in FIG. 18B, that is, the change of the dance meter by an operation input before interruption (first operation input). The CPU 101 calculates a value of the dance meter after in-game time T8, using a point 1850 as a start point.

The CPU 101 terminates display of the state image 251 at in-game time T10.

Then, the CPU 101 displays, instead of the state image 251, the history image 252 using a history of a game parameter from the point 1840 on the curve 1810 at in-game time T10 to the point 1850 on the curve 1810 at in-game time T8.

From in-game time T10 to T8 after resumption of the game, the CPU 101 will not receive any of an operation instruction, interruption instruction and resumption instruction. During this time, the player “runs through” his/her last play, whereby the player easily grasps the continuity of the game and easily progresses the game even after the predetermined period has passed.

Embodiment 4

Next, another embodiment of the present invention will be described. In the above embodiments, the player inputs an interruption instruction whereas in the present embodiment, the game device 200 can interrupt the game without an interruption instruction from the player.

For example, if there is no operation input for a predetermined threshold time period after the last operation
instruction from the player, the CPU 101 deems that there is an interruption instruction and may interrupt the game.

For example, in the case where the controller 105 has a built-in acceleration sensor, if an absolute value of a measured acceleration is less than a predetermined threshold value for more than a predetermined time period, the CPU 101 deems that there is an interruption instruction and may interrupt the game.

For example, in the case where the game device 200 is provided with a sensor for detecting a player, if the player cannot be detected for more than a predetermined time period, the CPU 101 deems that there is an interruption instruction and may interrupt the game.

When the CPU 101 deems that there is an interruption instruction and interrupts the game, the CPU 101 displays a dialog box 1910 for receiving an instruction on whether or not the game is terminated, as illustrated in FIG. 19.

If it is instructed to terminate the game, the CPU 101 terminates the display of the state image 251 and terminates the game.

If it is instructed not to terminate the game, the CPU 101 deems a resumption instruction to be inputted and resumes the game from the predetermined period before, as described above.

Displaying the dialog box 1910 for confirmation can prevent the game from being terminated without the player’s intent.

The present invention is not limited to the above embodiments, and various variations and applications are possible. In addition, respective components of the above embodiments can be combined freely.

The present invention can be used to realize a function by which whenever the player (especially, beginner) wants to practice a certain part of the game, the player can enter into a practice mode. In the above dance game, for example, the player poor at dancing to a certain phrase can input a practice instruction (corresponding to the above interruption instruction) when the phrase is about to start. When the practice instruction is inputted, the game is resumed from a predetermined period before, whereby the player can intensively practice around the phrase. Therefore, the present invention can provide an opportunity to assist the player’s proficiency.

If it is determined that a technical skill of the player is not sufficient, the present invention can be used to realize a function for the player to enter a practice mode. For example, in a shooting game in which stages should be cleared in series, if the CPU 101 determines the player has trouble in clearing a certain stage (for example, takes a longer time than a reference clear time), the CPU 101 deems that there is a practice instruction (corresponding to the above interruption instruction), and resumes the game a predetermined period before to make the player practice the game. The player can intensively practice a part at which he/she is poor at. Accordingly, the present invention can provide an opportunity to assist the player’s proficiency.

A program to make a computer operate as all or part of the game device 200 may be stored and distributed in a computer-readable recording medium such as a memory card, a CD-ROM, a DVD or an MO (Magneto Optical disk). The program may be installed to another computer to make the computer operate as the above means, or make the computer execute the above process.

Moreover, a program may be stored in, for example, a disc device in a server device on the Internet, superimposed on, for example, a carrier wave, and downloaded to a computer.

The present application claims priority right based on Japanese Patent Application No. 2008-319597, the entire content of which is incorporated herein by reference.

INDUSTRIAL APPLICABILITY

As described above, the present invention can provide a game device, a game processing method, an information recording medium and a program that are preferable for the player to grasp the continuity of the game before and after interruption of the game.

EXPLANATION OF REFERENCE NUMERALS

100 information processing device
CPU
102 ROM
103 RAM
104 interface
105 controller
106 external memory
107 DVD-ROM drive
108 image processing unit
109 sound processing unit
110 NIC
game device
201 state storage unit
202 history storage unit
203 operation reception unit
204 update unit
display unit
206 resumption control unit
207 left button
208 bottom button
top button
right button
screen (game screen)
stationary mark
step position indication mark
score
gauge (dance meter)
curve
point
buffer memory
1201 to 1203, 1401 to 1403 step position indication mark
image (history image)
1710, 1720, 1730, 1740 curve
1715, 1725 point
1810, 1820, 1830 curve
1840, 1850 point
1910 dialog box

What is claimed is:
1. A game device (200) being able to interrupt and resume a game, the game device (200) comprising:
   a state storage unit (201) to store a current state of the game;
   a history storage unit (202) to store a history of the game state stored in the state storage unit (201) while the game is not interrupted;
an operation reception unit (203) to receive an operation instruction from a player while the game is not interrupted; an update unit (204) to update the game state stored in the state storage unit (201) on the basis of the received operation instruction; and a resumption control unit (206) to resume the game from prior to a predetermined period before interruption of the game if the game is resumed after the interruption, wherein after resumption of the game followed by the predetermined period, if it is determined that the game state based on the operation instruction received during the predetermined period is better than the game state stored in the state storage unit (201) on the basis of a predetermined evaluation criterion, the game state stored in the state storage unit (201) is updated with the game state based on the operation instruction received during the predetermined period.

8. A computer-readable information recording medium configured to store a program that allows a computer that is able to interrupt and resume a game to function as:
a state storage unit (201) to store a current state of the game;
a history storage unit (202) to store a history of the game state stored in the state storage unit (201) while the game is not interrupted;
an operation reception unit (203) to receive an operation instruction from a player while the game is not interrupted;
an update unit (204) to update a game state stored in the state storage unit (201) on the basis of the received operation instruction; and a resumption control unit (206) to resume the game from prior to a predetermined period before interruption of the game if the game is resumed after the interruption, wherein after resumption of the game followed by the predetermined period, if it is determined that the game state based on the operation instruction received during the predetermined period is better than the game state stored in the state storage unit (201) on the basis of a predetermined evaluation criterion, the game state stored in the state storage unit (201) is updated with the game state based on the operation instruction received during the predetermined period.