

TRANSLATOR'S DECLARATION:

I, Carl T. Sullivan hereby declare:

1. I am well versed in both the Japanese and English languages and have over 30 years of experience translating Japanese technical documents into English.
2. The following translation of the patent document JP2006323502A (Arai) to English is accurate and complete to the best of my knowledge.

I declare under penalty of perjury under the laws of the United States of America that the foregoing is true and accurate. Executed this 2nd Day of April, 2024, at Sterling, Utah.



Carl T. Sullivan

Samsung v. Cerence
IPR2024-01267
Exhibit 1005

(19) Japan Patent Office (JP)

(12) Japanese Unexamined Patent
Application Publication (A)(11) Japanese Unexamined Patent
Application Publication Number2006-323502
(P2006-323502A)

(43) Publication date November 30, 2006

(51) Int. Cl.	FI	Theme Code (Reference)
G06K 9/03 (2006.01)	G06K 9/03 B	5B009
G06F 3/041 (2006.01)	G06F 3/03 380R	5B064
G06F 17/22 (2006.01)	G06F 17/22 514A	5B068
G06K 9/62 (2006.01)	G06K 9/62 G	
G06K 9/72 (2006.01)	G06K 9/72 Z	

Request for examination: Not yet requested: Number of claims: 10 Online (Total of 28 pages)

(21) Application number	2005-144227 (P2005-144227)	(71) Applicant	000001007 CANON CORP. 3-30-2 SHIMOMARUKO, OTA-KU, TOKYO-TO
(22) Date of application	May 17, 2005	(74) Agent	100076428 PATENT ATTORNEY OTSUKA, YASUNORI
		(74) Agent	100112508 Patent Attorney TAKAYANAGI, Jiro
		(74) Agent	100115071 Patent Attorney OTSUKA, Yasuhiro
		(74) Agent	100116894 Patent Attorney KIMURA, Shuji
		(72) Inventor	ARAI, Tsuneichi c/o CANON CORP. 3-30-2 SHIMOMARUKO, OTA-KU, TOKYO-TO
		F Terms (Reference)	5B009 LA01 MB03

Continued on last page.

(54) (TITLE OF THE INVENTION)

Information Processing Device and Controlling Method and Program Thereof

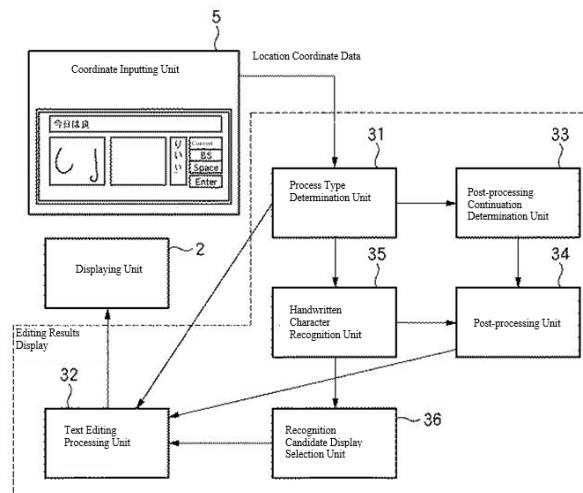
(57) [ABSTRACT]

[PROBLEM] To provide an information processing device, controlling method thereof, and program, able to execute, efficiently and with good accuracy, handwritten character recognition.

[RESOLUTION MEANS]

Characters corresponding to stroke data that are inputted manually are recognized. Candidate character strings for the character recognition results are subjected to post-processing. Whether or not to append a newly inputted character to a character string that is subject to post-processing is evaluated based on a process type according to an instruction for calling for execution of various types of processes, including input processes and processes relating to character recognition result candidates.

[Selected Drawings] FIG. 3



(2)

[CLAIM 1]

An information processing device for inputting handwritten character stroke data, recognizing a character, and executing post-processing on the character recognition result, equipped with:

an input means for inputting stroke data;

a character recognition means for recognizing a character corresponding to the stroke data inputted by the input means;

a post-processing means for performing post-processing of a candidate character string of results of character recognition by the character recognition means;

an instruction means for instructing the execution of various types of processes comprising an inputting process by the input means and a process relating to a character recognition result candidate; and

a determination means for determining, based on a process type for which execution is directed by the instruction means, whether or not to append a new character, inputted after an instruction by the instruction means, to a post-processing character string that has already become subject to post-processing by the post-processing means.

[CLAIM 2]

The information processing device according to claim 1, further comprising an insertion means for inserting, as the newly inputted character, a character for which a character code corresponding thereto is confirmed.

[CLAIM 3]

The information processing device according to claim 1, wherein when the process type is an inputting process by the input means, the determination means causes a candidate character that is newly inputted, as a character recognition result by the character recognition means in relation to stroke data that are inputted by the input means, to be appended to a post-processing character string that is already subject to post-processing by the post-processing means.

[CLAIM 4]

The information processing device according to claim 1, further equipped with a display means for displaying an operating screen that includes the input means, the instruction means, and a candidate display area for displaying a candidate character of a character recognition result by the character recognition means; an application screen that is generated by an application that operates on the information processing device; and a switching operation screen for switching a plurality of types of applications that run on the information processing device.

[CLAIM 5]

The information processing device according to claim 4, wherein the input means is equipped with an input area for inputting the stroke data; and

the instruction means is equipped with:

a changing unit for changing a candidate character in the candidate display area;

a conversion unit for carrying out kana-kanji conversion of a candidate character in the candidate display area;

a deletion unit for deleting a candidate character before the position of a cursor that is displayed on the candidate display region;

a space character insertion unit for inserting a space character in the candidate display region; and

a candidate character string confirmation unit for confirming a candidate character string in the candidate display area and inserting the candidate character string into the application screen.

[CLAIM 6]

(3)

The information processing device according to claim 5, wherein when the process type is one from among an inputting process for the input area, a timer event process that occurs when a time without input into the input area exceeds a prescribed time interval, a deleting process by the deletion unit, or a switching process by the switching operation screen, a candidate character that is newly inputted as a character recognition result by the character recognition means for stroke data that have been inputted into the input area is caused to be appended to a post-processing character string that is already subject to post-processing by the post-processing means.

[CLAIM 7]

The information processing device according to claim 1, further equipped with an insertion means for inserting characters that are stored in a storing area that is secured in internal memory by the operating system that controls the information processing device, wherein when the process means is an insertion process by the insertion means, the determination means causes a character that is newly inputted by the insertion process to be appended to a post-processing character string that is already subject to post-processing by the post-processing means.

[CLAIM 8]

The information processing device according to claim 1, wherein the post-processing means calculates a chain probability between candidate character strings of the character recognition results.

[CLAIM 9]

A controlling method for an information processing device for inputting handwritten character stroke data, recognizing a character, and executing post-processing on the character recognition result, comprising:

- an input step for inputting the stroke data;
- a character recognition step for recognizing a character corresponding to the stroke data inputted by the input step;
- a post-processing step for performing post-processing of a candidate character string of results of character recognition by the character recognition step;
- an instruction step for instructing the execution of various types of processes, comprising an inputting process by the input step and a process relating to a character recognition result candidate; and
- a determination step for evaluating, based on a process type for which execution is directed by the instruction step, whether or not to append a new character, inputted after an instruction by the instruction step, to a post-processing character string that has already become subject to post-processing by the post-processing step.

[CLAIM 10]

A program for achieving a controlling method for an information processing device for inputting handwritten character stroke data, recognizing a character, and executing post-processing on the character recognition result, equipped with:

- program code for an input step for inputting stroke data;
- program code for a character recognition step for recognizing a character corresponding to the stroke data inputted by the input step;
- program code for a post-processing step for performing post-processing of a candidate character string of results of character recognition by the character recognition step;

(4)

program code for an instruction step for instructing execution of various types of processes comprising an inputting process by the input step and a process relating to a character recognition result candidate; and

program code for a determination step for evaluating, based on a process type for which execution is directed by the instruction step, whether or not to append a new character, inputted after an instruction by the instruction step, to a post-processing character string that has already become subject to post-processing by the post-processing step.

[DETAILED DESCRIPTION OF THE INVENTION]

[TECHNICAL FIELD]

[0001]

The present invention relates to an information processing device and controlling method and program thereof for inputting stroke data of a handwritten character and performing character recognition, and executing post-processing in relation to the character recognition result.

[BACKGROUND ART]

[0002]

Conventionally there have been character recognition devices for identifying paths (strokes) of inputted handwritten characters. In such a character recognition device (for example, in Patent Document 1) there has been the need for indicating clearly positions of breaks by designating to the user that strokes are to be inputted in a specific area that is surrounded by a character input box, in order to designate breaks between words.

[0003]

In online handwritten character identifying devices that execute post-processing by character string units, there are those that are configured so as to define, as subject to post-processing, handwritten character strings that are inputted up until a button is pressed to start recognition.

[PATENT DOCUMENT 1] Japanese Unexamined Patent Application Publication H6-208655

[DISCLOSURE OF THE INVENTION]

[PROBLEM TO BE SOLVED BY THE INVENTION]

[0004]

However, in the conventional character recognition device, the device side cannot recognize a break between words unless the user instructs inputting of word breaks each time. Because of this, during inputting the user is forced to perform other input operations in addition to the actual intended character input, adversely affecting a natural use experience.

[0005]

Moreover, when carrying out post-processing by the character string unit, the handwritten character string that is inputted up until the button is pressed to initiate character recognition becomes subject to the post-processing. Because of this, when the button to start recognition is pressed part way through inputting of a character string, the characters that are inputted thereafter will not be linked to the recognized characters, which may reduce the recognition rate and preclude the desired recognition result.

[0006]

The present invention was created to solve the problem set forth above; the object thereof is to provide an information processing device, and controlling method and program thereof, that enable efficient and accurate execution of handwritten character recognition.

[MEANS FOR SOLVING THE PROBLEM]

[0007]

(5)

An information processing device according to the present invention, in order to achieve the object set forth above, comprises the following structures, specifically:

- an information processing device for inputting handwritten character stroke data, recognizing a character, and executing post-processing on the character recognition result, equipped with:

- an input means for inputting stroke data;

- a character recognition means for recognizing a character corresponding to the stroke data inputted by the input means;

- a post-processing means for performing post-processing of a candidate character string of results of character recognition by the character recognition means;

- an instruction means for instructing the execution of various types of processes comprising an inputting process by the input means and a process relating to a character recognition result candidate; and

- a determination means for determining, based on a process type for which execution is directed by the instruction means, whether or not to append a new character, inputted after an instruction by the instruction means, to a post-processing character string that has already become subject to post-processing by the post-processing means.

[0008]

Preferably it further comprises insertion means for inserting, as the newly inputted character, a character for which a character code corresponding thereto is confirmed.

[0009]

Moreover, preferably, when the process type is an inputting process by the input means, the determination means causes a candidate character that is newly inputted, as a character recognition result by the character recognition means in relation to stroke data that is inputted by the input means, to be appended to a post-processing character string that is already subject to post-processing by the post-processing means.

[0010]

Moreover, preferably it further comprises a display means for displaying an operating screen that comprises the input means, the instruction means, and a candidate display area for displaying a candidate character of a character recognition result by the character recognition means; an application screen that is generated by an application that operates on the information processing device; and a switching operation screen for switching a plurality of types of applications that run on the information processing device.

[0011]

- Preferably the input means comprises an input area for inputting the stroke data; and

- the instruction means comprises:

- a changing unit for changing a candidate character in the candidate display area;

- a conversion unit for carrying out kana-kanji conversion of a candidate character in the candidate display area;

- a deletion unit for deleting a candidate character before the position of a cursor that is displayed on the candidate display area;

- a space character insertion unit for inserting a space character in the candidate display area; and

- a candidate character string confirmation unit for confirming a candidate character string in the candidate display area and inserting the candidate character string into the application screen.

[0012]

Moreover, preferably, when the process type is that of an input process for the input area, a timer event process that occurs when a time without input into the input area exceeds a

(6)

prescribed time interval, a deleting process by the deletion unit, or a switching process by the switching operation screen, a candidate character that is newly inputted as a character recognition result by the character recognition means for stroke data that have been inputted into the input area is caused to be appended to a post-processing character string that is already subject to post-processing by the post-processing means.

[0013]

Moreover, preferably it is further equipped with an insertion means for inserting characters that are stored in a storage area that is secured in internal memory by the operating system that controls the information processing device, wherein when the process hand type is an insertion process by the insertion means, the determination means causes a character that is newly inputted by the insertion process to be appended to a post-processing character string that is already subject to post-processing by the post-processing means.

[0014]

Moreover, preferably the post-processing means calculates a chain probability between candidate character strings of the character recognition results.

[0015]

Moreover, in order to achieve the object described above, a method for controlling an information processing device according to the present invention is equipped with the following composition. Specifically, it is a control method for an information processing device for inputting handwritten character stroke data, recognizing a character, and executing post-processing on the character recognition result, equipped with:

- an input step for inputting stroke data;
- a character recognition step for recognizing a character corresponding to the stroke data inputted by the input step;
- a post-processing step for performing post-processing of a candidate character string of results of character recognition by the character recognition step;
- an instruction step for instructing execution of various types of processes including an input process by the input step and a process relating to a character recognition result candidate; and
- a determination step for determining, based on a process type for which execution is directed by the instruction step, whether or not to append a new character, inputted after an instruction by the instruction step, to a post-processing character string that has already become subject to post-processing by the post-processing step.

[0016]

In order to achieve the above object, the program according to the present invention is equipped with the following features. Specifically, it is a program for achieving a control method for an information processing device for inputting handwritten character stroke data, recognizing a character, and executing post-processing on the character recognition result, equipped with:

- program code for an input step for inputting stroke data;
- program code for a character recognition step for recognizing a character corresponding to the stroke data inputted by the input step;
- program code for a post-processing step for performing post-processing of a candidate character string of results of character recognition by the character recognition step;
- program code for an instruction step for instructing execution of various types of processes comprising an inputting process by the input step and a process relating to a character recognition result candidate; and

(7)

program code for a determination step for determining, based on a process type for which execution is directed by the instruction step, whether or not to append a new character, inputted after an instruction by the instruction step, to a post-processing character string that has already become subject to post-processing by the post-processing step.

[EFFECTS OF THE INVENTION]

[0017]

The present invention enables the provision of an information processing device, and a control method and program thereof, so as to be able to execute handwritten character recognition efficiently and accurately.

[PREFERRED MODE FOR CARRYING OUT THE PRESENT INVENTION]

[0018]

Embodiments according to the present invention will be explained in detail below using the drawings.

[0019]

The present invention provides a post-processing continuation determination unit, depending on the type of input instructed by a user in an information processing device that executes post-processing, by the character string unit, on character strings through handwritten input. Given this, the explanation will be for a configuration that achieves post-processing naturally, and by appropriate processing units, through the post-processing continuation determination unit, without the user consciously making any particular inputs instructing the end or continuation, or the like, of a text element. Through this, the present invention is able to achieve an improvement in the probability that the characters that are the result of recognition of a handwritten input by the user will be the recognition result that is intended by the user.

[0020]

Moreover, an insertion unit is provided for inserting, as a portion of a character string that is to be subjected to post-processing, an ordinary text character string (character code string) that is structured from a character code. Additionally, a configuration will be described that achieves proper post-processing even when a text character string is inserted into a character recognition result editing area that is to be subject to post-processing, through this insertion unit. This enables an improvement in the efficiency of inputting sentences as intended by the user, even when handwritten character recognition results and ordinary text character strings are mixed.

[0021]

< EMBODIMENT 1 >

FIG. 1 is a diagram depicting an example of the form of use of a character recognition device of a first embodiment according to the present invention. FIG. 2 is a diagram depicting a hardware structure for a mobile information device that functions as the character recognition device of the first embodiment according to the present invention.

[0022]

FIG. 1 is a mobile information device, such as a PDA (Personal Digital Assistant), mobile telephone, digital camera, notebook PC (personal computer), or the like.

[0023]

This mobile information device 1 is equipped with: a display unit 2; a stylus 3; a camera unit 4; a coordinate input unit 5; a CPU 6; a RAM 7; a ROM 8; a media interface (I/F) 9 that is equipped with a slot for insertion of a storing medium (for example, a Smart Media, a CompactFlash (registered trademark), or the like); a keyboard 10 (which may be an ordinary keyboard or a soft keyboard that is displayed on the display unit 2); an external interface (USB,

Bluetooth, or the like) 11 for connecting to another information device; and a network interface 12.

[0024]

The mobile information device 1 displays, on a liquid crystal display unit 2, an image captured by the camera unit 4, where additional information, such as caption information, or the like, to be added to the image, can be produced through stroke data that is inputted through the coordinate input unit 5 (handwritten stroke data through a handwritten inputting operation by the user).

[0025]

The CPU 6 is connected through a system bus 13 to the various constituent elements of the mobile information device 1, such as the RAM 7, the ROM 8, the coordinate input unit 5, the display unit 2, and the like. Moreover, the CPU 6 executes various types of control through programs that are stored in the ROM 8.

[0026]

The RAM 7 not only stores various types of data, such as image data, input stroke data, location information thereof, and the like, but also functions as a data work area and temporary storage area.

[0027]

The ROM 8 stores control programs for the mobile information device, including various types of programs for implementing the present invention. More specifically, the ROM 8 stores, in addition to various types of programs such as a control processing program for post-processing at the time of handwritten character input, a camera unit control program, a display unit control program, a program for recognizing handwritten stroke data, and the like, but also data such as recognition dictionary data for recognizing handwritten characters, post-processing dictionary data (dictionary data for chain probability calculation), various types of table and setting data, and the like.

[0028]

Input stroke data can be recognized into a corresponding character code, by a handwritten character recognition function that has existed conventionally, to be inputted into the mobile information device 1. Moreover, input stroke data may be added, as additional information, as an image (image data), without having been recognized, to a captured image, or text data obtained through the character recognition function may be added as additional information to the captured image.

[0029]

The image data for the captured image to which additional information has been added may be sent through an external interface 11 to an external terminal 50, such as a personal computer, or the like, in response to an instruction operation by a user. Conversely, after the image data has been stored on a storing medium, equipping in an external terminal 50 or printer that is equipped with a media slot enables storing or printing of the image data.

[0030]

Moreover, transmission to another external terminal on a network can be achieved through a network, such as a local network, public access network, or the like, that is connected through the network interface 12, through attaching the image data to an electronic mail that is generated by an electronic mail function realized by the mobile information device 1.

[0031]

2 is a display unit, structured from, for example, a liquid crystal or LEDs. If structured from a liquid crystal display unit, the display unit 2 will have liquid crystal display elements, a liquid

(9)

crystal displaying circuit, and display memory. This display unit 2 is connected to the CPU 6 through the system bus 13. Moreover, image data or stroke data is displayed, and various types of operating screens are displayed, through instructions from the CPU 6.

[0032]

A common transparent resistance film digitizer (known as a "touch panel") for inputting coordinates, can be achieved through disposing the coordinate input unit 5 overlaying the top face of the display unit 2. The user is able to input stroke data through pressing, with a stylus 3, on the coordinate input surface of the coordinate input unit 5 (the display screen of the display unit 2).

[0033]

More specifically, the coordinate input unit 5 is connected to the CPU 6 through an internal control circuit (not shown) via the system bus 13. As a result, through the use of the stylus 3 to press on the coordinate input surface of the coordinate input unit 5, the CPU 6 is notified of position coordinate data for a position that is pressed.

[0034]

Based on this position coordinate data, the mobile information device 1 displays, in accordance with a procedure within the ROM 8, an operating screen, such as a file list display screen, through a GUI (Graphical User Interface) that is implemented on the display unit 2. Moreover, any of a variety of processes can be executed, such as displaying the content of a data file that corresponds to a data file icon (a display that indicates a data file) on the operating screen at the position that is indicated by the position coordinate data that has been inputted, editing an image that corresponds to the input position, or the like.

[0035]

Moreover, instructions for executing various functions and screen operations realized by the mobile information device 1 can be achieved using the stylus 3, where, for example, the electronic mail function, referenced above, can also be used through the use of the stylus 3.

[0036]

3 is a stylus, used for inputting handwriting stroke data, soft buttons and soft keyboards on the display screen, and the like.

[0037]

4 is a camera unit, and is equipped with structural elements such as a lens, a CCD 4a, a CCD control circuit (not shown), and the like, which are the structural elements of a typical digital camera. Moreover, this camera unit 4 is controlled by the CPU 6 with the mobile information device 1, where image data that is captured is stored temporarily in the RAM 7 under the control of the CPU 6. The ability to send an image that is captured by the camera unit 4 through attaching it to an electronic mail is as was described above.

[0038]

More specifically, the CCD control circuit of the camera unit 4 is connected to the CPU 6 through the system bus 11. Because of this, the operation of the CCD 6 is controlled, through the CCD controlling circuit, by the CPU 6, and the image captured by the CCD 6 is stored in the RAM 7.

[0039]

The media interface 9 is an interface unit for a typical storage medium in a digital camera, or the like, and is, for example, a CompactFlash (registered trademark) card. In particular, the media interface 9 enables reading and writing of data of the connected media, and enables storage of data such as image data.

[0040]

The keyboard 10 is a typical keyboard, where key codes for alphanumerics, and the like, are inputted through pressing keys. The input key codes are read in by the CPU 6 through the system bus 13.

[0041]

The external interface 11 may be, for example, a USB interface unit. This may be controlled by the CPU 6 to connect to an external terminal that is a personal computer, or the like, to exchange various types of data such as stroke data, image data, and the like.

[0042]

The network interface 12 may be structured from, for example, a communication card. Connection controllers for controlling connections to public networks, such as PHS circuits, or the like, and an antenna are installed therein. It is possible to connect to a public circuit, to send and receive electronic mail, through sending a prescribed command to the controller that is in the communication card.

[0043]

Moreover, this communication card may be replaced with a network card, such as a local area network card, or the like, to be compatible with a local area network connection. Here the actual protocol for sending and receiving electronic mail is the same as that in a conventional method.

[0044]

The system bus 13 is a communication path that connects the CPU 6 to the various constituent elements of the mobile information device 1, including the RAM 7 and the ROM 8, to enable exchange of data therebetween.

[0045]

50 is an external terminal such as a personal computer, or the like. This external terminal 50 has the standard constituent elements that are equipped in a general use computer (such as, for example, a CPU, a RAM, a ROM, a hard disk, an external storage device, a network interface, an external interface display, a keyboard, a mouse, and the like).

[0046]

Moreover, in the first embodiment, the external terminal 50 may connect to the mobile information device 1 through an external interface, such as a typical USB, or the like, or through a network, to send and receive image data 6.

[0047]

The functional structure of the mobile information device of the first embodiment will be explained next using FIG. 3.

[0048]

FIG. 3 is a diagram depicting the functional structure of a mobile information device according to a first embodiment according to the present invention.

[0049]

Note that in FIG. 3 the coordinate input unit 5 and the display unit 2, which are hardware, are depicted, for use in the explanation, in addition to the various functional structures (surrounded by the dotted lines) of the mobile information device. This functional structure may be achieved, under control of the CPU 6, through a control program that is stored in the ROM 8, or some or all thereof may be achieved through specialized hardware circuits.

[0050]

In FIG. 3, 31 is a process type determination unit, for determining a process type in response to a range of position coordinate data received from the coordinate input unit 2. Additionally,

depending on the identified process type, processing is carried out with the input coordinate data allocated to various processing portions.

[0051]

32 is a text editing processing unit, which executes editing processes on a text character string obtained from the handwritten character recognition unit 35, such as deletion, kana-kanji conversion, insertion into a character input area of an application, and the like.

[0052]

33 is a post-processing continuation determination unit, which, depending on each individual process type received from the process type determination unit 31, evaluates whether to append to or terminate (stop) a post-processing character string (a character string that is subject to post-processing), and sends the determination result to a post-processing unit 34.

[0053]

34 is the post-processing unit. Here post-processing is carried out, in particular, by the character string unit, and an algorithm that is similar to the algorithm taught in Japanese Unexamined Patent Application Publication H10-198759, for example, may be used. Specifically, an inputted handwritten character string recognition candidate is determined through considering the probability that the characters in the candidate character string, recognized by the handwritten character recognition unit 35, are linked together, and considering the degree of match with reference character dictionary data within the recognition dictionary.

[0054]

Note that the post-processing is not limited thereto, but rather may be, for example, morphological analysis, statistical processes, or the like, insofar as they are post-processes on the character strings obtained from the recognition process.

[0055]

35 is a handwritten character recognition unit, which receives, from the process type determination unit 31, a position coordinate data string for the handwritten character, to output candidate character codes through a process of matching with reference character dictionary data within the recognition dictionary.

[0056]

36 is a recognition candidate selection display unit, which carries out a process for selecting and displaying handwritten character recognition candidates. This process is actually a typical process conventionally, where, when a display of a first candidate for a given handwritten character is tapped using the stylus 3, a candidate group for the first candidate through the fifth candidate for the character will be displayed in a candidate display area. Given this, when an arbitrary candidate is tapped from the set of candidates, a process is carried out to replace, with the character for the candidate that has been tapped, the first candidate that is displayed.

[0057]

Note that this functional structure was explained using, as an example, an implementation of a mobile information device 1; however, there is no limitation thereto. Moreover, the functional structure of the present invention, depicted in FIG. 3, can, of course, be achieved in a general-use terminal (information processing device) such as a personal computer, or the like, that has functions that are equal to or superior to those of a mobile information device.

[0058]

In the present invention, depending on the process type, in the various types of processes that are achieved in a handwritten character recognition environment, an evaluation is performed, in particular, of whether or not appending to the post-processing is possible, in the post-processing

of character string units, by the post-processing continuation determination unit 33. Given this, a post-processing continuation determination table is stored in the RAM 7, for example, in order to achieve this determination by the post-processing continuation determination unit 33.

[0059]

The post-processing continuation determination table will be explained here using FIG. 4.

[0060]

FIG. 4 is a diagram depicting a post-processing continuation determination table of the first embodiment according to the present invention.

[0061]

This post-processing continuation determination table defines correspondence between, and controls, information on the process type and on whether or not it is possible to append thereafter for post-processing (whether or not the entirety of a character string that includes new character, appended to a candidate character string for which a character recognition result has already been obtained, will be a character string that is subject to post-processing).

[0062]

For example, when the process type determined by the process type determination unit 31 is a candidate change process for a handwritten character recognition candidate, the post-processing continuation determination unit 33 references the post-process continuation determination table to evaluate at that time whether to temporarily stop, at that time, post-processing the character that was inputted prior to the candidate selection instruction.

[0063]

Moreover, as another example, when carrying out an operation to break character data of a single character using character cutoff time (a process (timer event) that occurs when the pen-up time (the time over which there is no operation by the stylus 3: a non-input time) has extended beyond a given time period), the post-processing continuation determination unit 33 references the post-processing continuation determination table to evaluate whether to continue post-processing on the characters that had been inputted up to the time of this operation and the character inputted thereafter.

[0064]

In this way, the present invention is able to provide an operating environment that is user-friendly, through determining whether to continue or to stop the post-processing based on the input operation type and event type each time there is an input operation or an event occurs through the stylus 3, or the like.

[0065]

Note that this post-processing continuation determination table is an example, where implementation may be through a variety of different structures depending on the application and purpose. That is, post-processing continuation determination tables can be structured, of course, for a variety of different settings, depending on the process types and events wherein post-processing is to be applied to the character strings obtained through the recognition processes, depending on the intentions and objectives of the user.

[0066]

A specific example of the character recognition process in the first embodiment will be explained next using FIG. 5.

[0067]

FIG. 5 is a diagram for explaining a specific example of a character recognition process of the first embodiment according to the present invention.

[0068]

FIG. 5 depicts an example of a display screen during the operation of a handwritten character input utility program that achieves the character recognition process according to the present invention in an electronic mail application that runs on the mobile information device 1.

[0069]

Note that in, the first embodiment, the explanation will use an example of using a handwritten character input utility program on an application screen generated, for example, by an electronic mail application; however, there is no limitation thereto. That is, the present invention is, of course, effective in other application programs as well that required character input, such as, for example, file browsers, photo viewers, schedulers, word processing programs, and the like.

[0070]

In FIG. 5, area 51 is an area that comprises a variety of input fields for inputting a destination and subject for the electronic mail. For example, a cursor for text input will be displayed in the corresponding input field through pressing an arbitrary position within the area 51 using the stylus 3. The user may, in this state, use the handwritten character input utility program to execute inputting of text into this input field.

[0071]

Area 52 is an area for inputting the body of the electronic mail. Through, for example, pressing the stylus 3 at an arbitrary position within this area 52, the cursor for text input will be displayed in the area 52. The user, in this state, is able to input text into this area through the use of the handwritten character input utility program.

[0072]

Screen 53 is a handwritten character input screen that is realized by the handwritten character input utility program. While the details thereof are explained using FIG. 6, when it is necessary to input text into an arbitrary application program, the user can input characters, through a handwritten input, by starting this handwritten character input utility program.

[0073]

Of course, a configuration is possible wherein application programs that require the handwritten character input utility program are assigned in advance so that the handwritten character input utility program will start simultaneously with launching of those application programs.

[0074]

When handwritten character input (stroke data input) is carried out by the user on the handwritten character input screen 53 of the handwritten character input utility program, the stroke data thereof is recognized by the handwritten character recognition unit 35 of FIG. 3. Given this, the character font corresponding to the character code obtained as the recognition result is displayed temporarily by the handwritten character input screen 53.

[0075]

If the character displayed by the handwritten character input screen 53 is the character that is desired by the user, the user confirms this character to insert, at the cursor position that is displayed on the electronic mail application program, the character that is currently displayed.

[0076]

Area 54 is an area wherein a control group for switching between the various application programs and controls (buttons and icons) for launching the handwritten character input utility program are arranged.

[0077]

(14)

In this area 54, a button 55 is the button for starting the handwritten character input utility program. The handwritten character input utility program will start up, and the handwritten character input screen 53 will be displayed, when this button 55 is pressed using the stylus 3. On the other hand, if the button 55 is pressed in a state wherein the handwritten character input utility program is already running, the handwritten character input utility program will close, and the screen 53 for inputting handwritten character will be removed.

[0078]

Button 56 is a button for starting the file browser. When this button 56 is pressed using the stylus 3, a file browser application for browsing various types of data, under the control of the mobile information device 1, will be started.

[0079]

Button 57 is a button for starting the electronic mail application. The electronic mail application is started when this button 56 is pressed using the stylus 3.

[0080]

Button 58 is a button for starting the photo viewer. The photo viewer (an application for displaying photographs), for viewing image data under the control of the mobile information device 1, is started when this button 56 is pressed using the stylus 3.

[0081]

Button 59 is a button for starting the scheduler. The scheduler application is started when this button 59 is pressed using the stylus 3.

[0082]

The details of the screen 53 for inputting handwritten character, generated by the handwritten character input utility program, will be explained next using FIG. 6.

[0083]

FIG. 6 is a diagram depicting an example of a screen for inputting handwritten character, generated by the handwritten character input utility program of the first embodiment according to the present invention.

[0084]

In FIG. 6, the area 61 is an input character buffer area, and displays a first candidate for the handwritten character recognition result. First candidates for recognition results are displayed sequentially here when the inputting of a single handwritten character is confirmed and the recognition process is terminated. For example, when "今日は" is inputted through handwritten input, if this constitutes characters wherein the stroke data are identical to the shapes that are registered in the recognition dictionary (the reference character dictionary data), then "今日は" will be displayed in this area 61 as the recognition result.

[0085]

Here, if a character that is not the intended candidate is comprised in the set of characters that is displayed in the input character buffer area 61, the user can replace the candidate by selecting, through pressing with the stylus 3, the character that is not the intended candidate.

[0086]

Specifically, when the user selects, through tapping, an arbitrary character within the input character buffer area 61, a group of the first through fourth candidates for that character will be displayed, in the rank order of the candidates, in the candidate display area 66. Given this, if the intended candidate is included in this set of candidates, through pressing that candidate with the stylus 3 the user can replace the character for the first candidate, that is selected in the input character buffer area 61, with the character selected in the candidate display area 66.

[0087]

When the function of the post-processing unit 34 is explicitly disabled through a user instruction, the set of first candidate characters that is displayed in the input character buffer area 61 will reflect, as-is, the rank order of the recognition candidates in the recognition result by the handwritten character recognition unit 35. On the other hand, if the functions of the post-processing unit 34 are explicitly enabled through a user instruction, the rank ordering of the recognition candidates is replaced, and the characters are displayed in the input character buffer area 61, based on the chain probability for the character string, and the degrees of matching of the recognition results, based on the recognition candidates of the identification results by the handwritten character recognition unit 35.

[0088]

The button 62 is the Conversion button. Kana-kanji conversion is carried out for the hiragana character in the input character buffer area 61 when this button 62 is pressed using the stylus 3. For example, if, in a state wherein "あいが" has been inputted into the input character buffer area 61, this button 62 is pressed, it will be converted to, for example "愛が". If the conversion candidate is not what was intended, pressing this button 62 again will display the next candidate.

[0089]

The button 63 is the Backspace key button (the deletion button). If character is displayed in the input character buffer area 61 and this button 63 is pressed using the stylus 3, then one character, preceding the position of the cursor that is displayed in this input character buffer area 61, will be deleted. On the other hand, if not even a single character has been inputted for the text within the input character buffer area 61, one character will be deleted from the left side of the cursor that is displayed in the application (for example, in area 52 of the electronic mail application).

[0090]

The button 64 is the Space Key button (a space character insertion button). If a character string is displayed in the input character buffer area 61 and the insertion position is at the end of the character string, and this button 64 is pressed using the stylus 3, a space character (space code) will be inserted after that character string.

[0091]

The button 65 is an Enter Key button (a candidate character string confirmation/line break input button). If a character string is displayed in the input character buffer area 61 and this button 65 is pressed using the stylus 3, this character string is confirmed and the character string is inserted to the right side of the cursor that is displayed in the application. On the other hand, if no text is present in the input character buffer area 61, a line break code is inserted on the right side of the cursor that is displayed in the application, and the cursor is moved to the next line.

[0092]

Area 66 is the candidate display area for handwritten character recognition. As described above, the first-ranked through fourth-ranked recognition candidates that correspond to the character that is selected in the character string of the input character buffer area 61 will be displayed here.

[0093]

As described above, the selected character will be replaced into the input character buffer area 61 when an arbitrary character within the text that is displayed in the candidate display area 66 is selected through pressing with the stylus 3.

[0094]

For example, if a character string for a recognition result of "今日は良り" is displayed within the input character buffer area 61 and "り" is selected by pressing with the stylus 3, then, as the characters that are the first candidate through the fourth candidates for the "り" will be displayed as, for example, "り", "ㇿ", "ゝ", and "ゝ", or the like in the candidate display area 66. The user is able to select an arbitrary character therefrom. For example, when the "ゝ" is selected by pressing with the stylus 3, the "り" within the input character buffer area 61 will be replaced with the "ゝ," so that the entire character string will be changed from "今日は良り" to "今日は良い".

[0095]

The area 67 is a handwritten character input area. Character input can be carried out through alternately filling in handwritten character into the two regions 67a and 67b.

[0096]

For example, when a "ゝ" is written into the area 67a followed by writing the first stroke for forming "天" in handwriting in area 67b, the input of "ゝ" in area 67a is confirmed and the recognition process on this "ゝ" is commenced. Here, at the instant that recognition of the "ゝ" has been completed, the first candidate for the recognition result of "ゝ" will be displayed in the input character buffer area 61, and at that instant the strokes of the handwritten character that have been inputted into the area 67a will be erased.

[0097]

Processing is next carried out for area 67b. At the instant that inputting of the handwritten strokes for forming the character "天" has been completed, then, if there is a subsequent character to draw and drawing of this subsequent character is started in the area 67a, the recognition process on "天" is commenced.

[0098]

On the other hand, if there is no subsequent character to input, then at the instant that a prescribed time interval (the pen-up time) has elapsed after the moment at which the stylus 3 (for example, pressing) was operated, it is confirmed that inputting of the "天" has been completed, and the recognition process is carried out. Conversely, if the stylus 3 is touched to a part other than the handwritten character input area 67, then the completion of the inputting of the "天" character can be confirmed at that time.

[0099]

Given this, at the instant that the recognition process for "天" has been completed, the first candidate for the recognition result for "天" will be displayed on the right side of the character that is the recognition result for "ゝ" in the input character buffer area 61, and the handwritten strokes that form the public "天" character in area 67b will be erased.

[0100]

Note that the actual processes for inputting the handwritten character may be achieved in the same manner as conventional handwritten character inputting processes.

[0101]

Moreover, while the various operation (instruction) controls (buttons, menus, etc.) for directing execution of various types of processes, depicted in FIG. 5 and FIG. 6, show exemplary implementations on a GUI, these may instead be implemented as physical buttons or keys on the mobile information device, for example.

[0102]

(17)

The processes executed in the first embodiment will be explained next using FIG. 7 through FIG. 9.

[0103]

FIG. 7 is a flowchart depicting a character string post-processing continuation determination process in the first embodiment according to the present invention.

[0104]

Note that this process is executed primarily by the post-processing continuation determination unit 33. When executing this process, the work area, and the like, in the RAM 7, which will be used in the process, is initialized.

[0105]

First, the characters input by the user using the stylus 3 and editing operations are input as position coordinate data by the coordinate input unit 22. Given this, the detail of the operation, such as whether this is an operation for inputting a handwritten character, an editing operation, a candidate selecting operation, or another operation, is identified by the position indicated by the position coordinate data. Note that this determination is achieved through the process type determination unit 31 of FIG. 3, as described above.

[0106]

In the first embodiment, as an example of determining the input type by the processing type determination unit 31, a case where the input type is realized by the handwritten character input utility program shown in FIG. 6 will be described as an example. Note that, as described above, the handwritten character input utility program is started through pressing of the button 55 in the screen for inputting handwritten character in FIG. 5.

[0107]

The flowchart will be described in detail below.

[0108]

In Step S702, a determination is carried out as to whether or not this is a candidate change process that replaces the character string candidate in the input character buffer area 61. This determination evaluates whether or not the stylus 3 has pressed on the character string in the input character buffer area 61 in FIG. 6, for example. In particular, if a position coordinate within that area has been inputted, the determination will be that this is a candidate change process. Given this, if a candidate change process (YES in Step S702), the process advances to Step S703. On the other hand, if not a candidate change process (NO in Step S702), the process advances to Step S704.

[0109]

In Step S703, a character string post-processing completed flag is set. This character string post-processing completed flag is a flag that indicates whether or not character string post-processing has been completed.

[0110]

For example, a user carries out a candidate change process when the first candidate character string that is displayed in the input character buffer area 61 is other than the intended character string. Given this, when the candidate is replaced, the displayed candidate character string after the replacement will be the character string that is desired by the user. Consequently, the character string post-processing completed flag is set at this time in the RAM 7, for example, in order to terminate (stop) the post-processing on the character string that has been thus far. When this character string post-processing completed flag is set, a post-processing completion instruction is issued to the post-processing unit 34 in FIG. 3.

[0111]

In Step S704, a determination is made as to whether or not there is a handwritten character input. This determination is performed based on whether or not there is an input at a position coordinate within a handwritten character input area 67 of FIG. 6. If there is a handwritten character input (YES in Step S704), the process advances to Step S705. On the other hand, if there is no handwritten character input (NO in Step S704), the process advances to Step S706.

[0112]

In Step S705, a process is carried out to add the recognition candidates to the post-processing character string. After one character of the strokes for the handwritten character that has been inputted has been confirmed in the handwritten character input area 67 of FIG. 6, the position coordinate data for the strokes for the handwritten character that has been inputted is transferred to the handwritten character recognition unit 35 of FIG. 3. Following this, distinctive feature data of the character is extracted from the position coordinate data using an online handwritten character recognition algorithm that has existed conventionally.

[0113]

Following this, candidates for a first candidate through fourth candidate other than the inputted handwritten character are determined, in the sequence of the reference character dictionary data with the highest degree of matching of the distinctive features, through a process of matching the distinctive feature data and the standard character dictionary data (standard character distinctive feature data) within the recognition dictionary. Given this, the character codes that correspond respectively to the first candidate through fourth candidate that have been determined, and the evaluation value for the degree of matching, are passed to the post-processing unit 34.

[0114]

The post-processing unit 34 determines, from the character candidate strings, which comprise the character codes and the evaluation values for the recognition candidates, the combinations that have the highest chain probabilities between the character strings, to change the display within the input character buffer area 61 to display the combination with the highest value as the first candidate.

[0115]

For example, let us explain the case wherein the first display in the input character buffer area 61 is "12 ㄣ 4," through the handwritten character recognition unit 35.

[0116]

When examining the "ㄣ" here, in regard to this "ㄣ" the first candidate is "ㄣ", the second candidate is "3", the third candidate is "ㄣ," and the fourth candidate is "ㄣ". In this case, the character string combinations will be "12 ㄣ 4", "1234", "12 ㄣ 4" and "12 ㄣ 4".

[0117]

In this case, with chain probabilities, the combination "1234" will have the highest evaluation value, so if the difference in evaluation values between the "ㄣ" of the first candidate and the "3" of the second candidate is small, the combination "1234" will be selected. The result is that the display in the input character buffer area 61 will switch from "12 ㄣ 4" to "1234".

[0118]

In Step S706, it is determined whether or not the character cutoff time has elapsed.

[0119]

If a prescribed time interval has elapsed for an input by the stylus 3 following input of the most recent stroke of the strokes to draw a single character within the handwritten character input area 67 in FIG. 6, then a character cutoff time interval elapsed event is produced. Note that this

(19)

configuration may be a conventional process that has been used in online handwritten character recognition conventionally.

[0120]

Therefore, when the character cutoff time interval has elapsed, this character cutoff time interval elapsed event is generated, that is, if the character cutoff time interval has elapsed (YES in Step S706), the process advances to Step S707.

[0121]

On the other hand, in another event, or if another position coordinate is inputted (NO in Step S706), the process advances to Step S708.

[0122]

In Step S707, a process for indicating continuation, in regard to the post-processing unit 34 that has added the recognition candidates to the post-processing character string is carried out. Further, in the conventional character recognition process that carries out post-processing, the post-processing is terminated at the instant that there has been a determination that the continuous input has been completed (for example, at the instant that the character cutoff time interval has expired).

[0123]

In relation to this, if inputting of the last stroke of the set of strokes for the handwritten character intended by the user is confirmed through the character cutoff time interval elapsed event in the first embodiment, the stroke data within the handwritten character input area 67 is first transferred to the handwritten character recognition unit 35. Given this, the candidate characters, outputted from the handwritten character recognition unit 35, and the evaluation values thereof, are passed to the post-processing unit 34, a post-processing continuation instruction is sent to the post-processing unit 34 to wait for input.

[0124]

As a result, the present invention enables the post-processing to be continued even after a character cutoff time interval elapsed event has occurred, and even for input thereafter.

[0125]

In Step S708, an evaluation is made as to whether or not the start recognition area has been pressed. In the first embodiment, the area for starting the recognition is defined as any area outside of the handwritten character input area 67 in FIG. 6, so when a position coordinate is inputted in relation to this area, a start recognition event is generated first.

[0126]

Specifically, when a coordinate outside of the handwritten character input area 67 of FIG. 6 has been inputted, then, if there is unprocessed stroke data within the handwritten character input area 67, then a start recognition event is generated first. When the start recognition event has been generated (YES in Step S708), the process advances to Step S709, where the data for the position coordinate input is stored, and a process is executed as the next event, at that position.

[0127]

For example, if, in a state wherein the strokes for the handwritten character "気" have been inputted into the handwritten character input region 67, the button 65 is pressed with the stylus 3, then first a start recognition event is generated through the area outside of the handwritten character input region 67 (the start recognition area) having been pushed, and handwritten character recognition is carried out. Following this, after the recognition result has been displayed in the input character buffer area 61, the process relating to the pressing of the button

65 will be executed. On the other hand, in the case of any other event (NO in Step S708), the process advances to Step S710.

[0128]

Whether or not a single character deleting process has been called is evaluated in Step S710. If a position coordinate within the button 63 has been inputted (YES in Step S710), the determination will be that this is a process for deleting one character, and the process advances to Step S711. On the other hand, if some other position coordinate has been inputted (NO in Step S710), the process advances to Step S712.

[0129]

The single character deleting process is executed in Step S711. That is, if there is a character in the input character buffer area 61, one character on the right end of the input character buffer area 61 will be deleted. Following this, a process is executed to remove the recognition candidates, within the post-processing character string within the post-processing unit 34, that correspond to the deleted character. On the other hand, if there is no character in the input character buffer area 61, the character on the left side of the cursor position in the application will be deleted, and because there is no data subject to processing in the post-processing unit 34, the process is terminated without doing anything.

[0130]

In Step S712 it is evaluated whether or not there is an application switching process. If an application switching command, when the position coordinates within button 56, 59, or the like, in FIG. 5 has been inputted (YES in Step S712), the evaluation will be that of an application switching operation, and the process advances to Step S713. On the other hand, if some other position coordinate is inputted (NO in Step S712), the process advances to Step S714.

[0131]

In Step S713, the post-processing character string is extended. In the first embodiment, the configuration is such that inputting of the handwritten character input utility program is continued even if the application is switched. This is envisioning a case wherein there will be a request to change the application during inputting. Because of this, a post-processing continuation instruction is sent to the post-processing unit 34 to wait for input.

[0132]

In Step S714, an evaluation is carried out whether or not there is a call for a kana-kanji converting process.

[0133]

If a position coordinate within the button 62 in FIG. 6 is inputted (YES in Step S714), then the evaluation is that of a kana-kanji converting process, and the process advances to Step S715. If some other position coordinate is inputted (NO in Step S714), the process advances to Step S716.

[0134]

A kana-kanji converting process is executed in Step S715. The kana-kanji converting process itself may be one of a variety of kana-kanji converting processes that have been carried out conventionally, and thus detailed explanations thereof are omitted. Explaining simply, when text has been inputted into the input character buffer area 61, for example, when the hiragana of "てんき" has been inputted, kana-kanji conversion is carried out on that character string, to convert into "天気", or the like. When this kana-kanji converting process is executed, a process is carried out to convert from the inputted character string into a character string with different character

(21)

codes. Because of this, the character string post-processing completed flag is set here for the post-processing unit 34, because it would be meaningless to continue with the post-processing.
[0135]

Whether or not there has been a call to insert the character string is evaluated in Step S716. If a command, or the like, for inserting a character string has been produced through a user instruction (YES in Step S7164), the process advances to Step S717. The same process as in Step S715 is executed in this Step S717. On the other hand, if some other position coordinate is inputted or there is a different event (NO in Step S716), the process will be terminated.
[0136]

The processes explained thus far make it possible to control the continuation or cancellation of the post-processing, which is the processing of the character string units of handwritten characters, depending on the inputting process or editing process for each of the handwritten character inputs.
[0137]

Note that the sequence of the various evaluations shown in FIG. 7 (Step S702, Step S704, Step S706, Step S708, Step S710, Step S712, Step S714, and Step S716) is not limited to the sequence depicted in FIG. 7, but rather the sequence may be varied depending on the application and the purpose.
[0138]

The handwritten character recognition process will be explained next using FIG. 8.
[0139]

FIG. 8 is a flowchart depicting a handwritten character recognition process according to the first embodiment according to the present invention.
[0140]

Note that when the handwritten character recognition process is started, the work areas, and the like, in the RAM 7, used by the processes, are first initialized, and these processes are executed primarily by the handwritten character recognition unit 35. For the position coordinate data for the strokes for a single character that is to be recognized, stroke data that is inputted into the respective areas, through areas 67a and 67b in the handwritten character input area 67 of FIG. 6, are inputted as stroke data for a single character.
[0141]

In Step S902, the stroke data for a single character is read into a work area. For example, stroke data for a single character, would consist of stroke data for a single stroke if a single-stroke character. If a character of two or more strokes, it would consist of a plurality of stroke data for individual strokes. The stroke data for a single stroke starts from, for example, XY position coordinate data for a starting point, which determines the position thereof, with XY position coordinate data following the shape of the stroke, continuing to the XY position coordinate of the endpoint. Given this, feature points (distinctive feature data) are extracted through performing normalization of the position coordinate data using a conventional algorithm.
[0142]

In Step S903, a recognition process is executed on the stroke data that has been obtained, where candidate characters (character codes) that are the recognition results are stored in the RAM 7. Here a matching process is carried out between the distinctive feature data for the stroke data, extracted in Step S902, and reference character dictionary data of the recognition dictionary. Given this, the character codes, and the values for the degree of matching (the evaluation values), corresponding to the reference character dictionary data that have the highest

(22)

degrees of matching, are stored as the recognition results, where character codes corresponding to the reference character dictionary data that have sequentially the highest degrees of matching are stored as recognition result candidates.

[0143]

In Step S904, the recognition result candidates are saved into the post-processing unit 34. That is, the character codes and evaluation values of the recognition results that were stored in Step S903 are saved into the post-processing unit 34. The post-processing unit 34 stores the character codes, and the evaluation values thereof, that are the recognition results that were recognized sequentially.

[0144]

Here the explanation will be for a case wherein the handwritten character "工場" has been inputted. First the first candidate character in the recognition result for "工", by the handwritten character recognition unit 35, is "工," and the evaluation value is 100, where the second candidate character is "𠄎", with an evaluation value of 100, and the third candidate character is "𠄎", with an evaluation value of 90.

[0145]

Next the first candidate character for the recognition result for "場" is "揚", with an evaluation value of 98, where the second candidate character is "場", with the evaluation value for the degree of matching being 96.

[0146]

In this case, in the post-processing unit 34 the chain probabilities for the character strings combining these candidates are calculated using a dictionary for post-processing. The combinations are, for example, "工場", "工場", "𠄎揚", "𠄎場", "𠄎揚", "𠄎場". Given this, the post-processing unit 34 determines the candidate character string with the highest chain probability among these accommodations.

[0147]

In Step S905, the candidate character string is read out from the post-processing unit 34.

[0148]

As described above, when "工場", is inputted as the handwritten characters, then even if the recognition results by the handwritten character recognition unit 35 are "工揚", the process in the post-processing unit 34 in Step S904 will determine that the candidate character string with the highest chain probability is "工場," so "工場" is read out as the candidate character string for the handwritten character. In this case, here the "工揚" that is the recognition result by the handwritten character recognition processing unit 35 may be replaced with "工場".

[0149]

In Step S906, the recognition result is displayed. For example, the "工場" is produced as the initial recognition result will be displayed in the input character buffer area 61 of FIG. 6, with "工場" through the process in Step S905.

[0150]

When this series of processes is explained as a time series, when first "工" is inputted and then the first stroke of the next character, "場" is inputted, at that instant the recognition result of "工" by the handwritten character recognition unit 35 for the "工" that has been inputted will be displayed in the input character buffer area 61.

[0151]

(23)

Next, when the character "場" is inputted, and is confirmed through, for example, the character cutoff time, recognition by the handwritten character recognition unit 35 is started on the "場", and if the first-ranked recognition result is "揚", the display will be "工場" as the recognition result in the input character buffer area 61.

[0152]

Next the candidate character string that has the highest chain probability, for a combination among the recognition candidates by the handwritten character recognition unit 35, is recognized, and if that candidate character string is "工場," then the "工場" of the display of the input character buffer area 61 will be replaced with "工場" by the post-processing unit 34.

[0153]

Moreover, if the user instruction that is inputted next is, for example, the handwritten character "が", then post-processing is executed with "工場が" as the subject for post-processing, as the inputted character string.

[0154]

On the other hand, if the "工場" post-processing result was not the intended recognition result, and there is an instruction for a candidate change process to another candidate, an instruction for a single-character deleting process, or the like, then the character string at that instant is handled as a confirmed character string, and continuation of the post-processing is terminated (that is, the character string post-processing completed flag is set at that instant). Thereafter, the handwritten character that is inputted next will be handled as the first character to be subjected to post-processing (in which case the character string post-processing completed flag is cleared).

[0155]

Thereafter, the handwritten character recognition process is terminated. Here the stroke data that was stored for the individual character is erased, and the work area, and the like, is released.

[0156]

As explained above, a post-processing continuation determination unit for evaluating whether or not to continue post-processing on a character string that has been inputted, based on the process type that is determined by an instruction input from the user, is provided through the first embodiment. This makes it possible to achieve a handwritten character recognition environment wherein character input and editing operations are carried out naturally without the user having to be aware of continuing or confirming the post-processing on the inputted character string (that is, without having to execute an operation to designate breaks in the post-processing unit). This enables a handwritten character recognition environment wherein the user can input the intended text easily.

[0157]

< EMBODIMENT 2 >

While in the first embodiment the explanation was for a case wherein the character string that is subjected to post-processing is handled as recognition results of handwritten characters, in normal mobile information devices there are many cases wherein text character strings copied from other applications are inserted, rather than handwritten character input. In such cases there will be mixtures, in a single text, of inserted text character strings and characters that are recognition results for handwritten characters. In this case, with handwritten character recognition devices that have existed in the past, post-processing cannot be carried out in such a state.

[0158]

(24)

Given this, the second embodiment adds an insertion unit 101 (FIG. 9), for inserting a inserted text character string into the post-processing unit 34, in addition to the structures of the first embodiment. Through this, it is possible to achieve good efficiency in post-processing in a state wherein there is a mixture, in a single character, of an inserted text character string and characters that are the results of recognition of handwritten characters.

[0159]

Note that an inserted text character string refers to, for example, a character string that has been copied from an arbitrary application, a character string that has been inputted in the past, or a character string wherein hiragana, after handwriting recognition, has been subjected to kana-kanji conversion.

[0160]

The functional structure of the mobile information device according to the second embodiment will be explained first using FIG. 9.

[0161]

FIG. 9 is a diagram depicting the functional structure of the mobile information device according to the second embodiment according to the present invention.

[0162]

Note that those structures that are identical to those of FIG. 3 of the first embodiment will have identical reference numbers applied thereto, and detailed explanations thereof will be omitted.

[0163]

As described above, in the second embodiment there is an insertion unit 101 in order to insert an inserted text character string into the post-processing unit 34. This insertion unit 101 is a processing unit that is called when a text character string, from another application program or that has been stored in the past, is to be inserted into the input character buffer area 61 within the handwritten character input utility program of FIG. 6.

[0164]

The insertion unit 101 executes a process for handling, as character strings that are subject to post-processing, candidate character strings that are the result of recognition of handwritten characters, and text character strings that have been inserted, inputted into the input character buffer area 61. The processed candidate character string during is sent to the post-processing unit 34.

[0165]

A specific example of a character recognition process according to the second embodiment will be explained next using FIG. 10 through FIG. 12.

[0166]

FIG. 10 through FIG. 12 are diagrams for explaining a specific example of a character recognition process according to the second embodiment according to the present invention.

[0167]

FIG. 10 through FIG. 12 depict examples of display screens when using the handwritten character input utility program on a word processor application.

[0168]

In FIG. 10, 111 depicts a text input area for the word processor application. 112 is a display for displaying, conceptually, a text character string that is inserted into the post-processing unit 34. A state is depicted wherein the characters "都市" are stored in a clipboard area (a storage area secured by the operating system in the internal memory) for text that is a function of the operating system on which the word processor application runs.

[0169]

FIG. 11 depicts a state wherein the character string for the characters "都市" has been inserted, from the state in FIG. 10, into the input character buffer area 61. Moreover, FIG. 12 depicts a state wherein a handwritten "思い" is inputted after the character string of "失われた都市の記憶を" in the input character string buffer area 61 of FIG. 11.

[0170]

The insertion process (which has been added) for inserting, into the post-processing unit 34, the character string that was inserted, by the insertion unit 101, will be explained next using FIG. 13.

[0171]

FIG. 13 is a flowchart depicting the insertion process in the second embodiment according to the present invention.

[0172]

Note that the transitions between states in FIG. 10 through FIG. 12 will also be explained following this flowchart.

[0173]

A character string is selected through a conventional operation, using a menu, operating keys, or the like, in another application, or the like, and copied into a specific storage area (a system character string storage area).

[0174]

Following this, the character string that has been stored in the specific storage area (the system character string storage area) is inserted, through an instruction, such as through a command or a menu, or the like, at a location wherein a text cursor is displayed.

[0175]

The process thus far is an ordinary process in a typical computer or mobile information device, so will not be explained in detail.

[0176]

The cursor is moved to after the "失われた" in the character string of the "失われたの記憶を" in the input character buffer area 61 of FIG. 10, and there is an instruction to insert there the "都市" that is stored in the system character string storage area.

[0177]

When there is a command for insertion, on the screen the character string will be "失われたの記憶を" of the input character buffer area 61 of FIG. 11. Note that the actual text insertion process is the same process as with a conventional mobile information device.

[0178]

This process is executed primarily by the insertion unit 101. In executing this process, the work area in the RAM 7 that is used by this process, and the like, is initialized.

[0179]

Following this, in Step S142, a stored text character string that is subject to insertion (the insertion text character string) is read into the system character string storage area. For example, the character codes for the insertion text character string "都市", shown in FIG. 10, for example, are stored in the RAM 7. Additionally, location information is stored for the position of the cursor when the insertion instruction is issued.

[0180]

(26)

In Step S143, a processing operation for processing the character string data into the same structure (format) as the character data that is processed by the post-processing unit 34 is executed in order to insert the insertion text character string, as a confirmed character string, into the post-processing unit 34. The character string data processed by the post-processing unit 34 (the post-processing character string data) is structured from a character code and an evaluation value. Given this, the insertion text character string is originally defined as character codes, and thus, in this processing operation, post-processing character string data is created after setting, for these character codes, the evaluation values to the maximum score of 100.

[0181]

For example, explaining for the case of "都市", post-processing character string data of (the character code for "都" with the evaluation value 100, and the character code for "市" with the evaluation value 100) is created, and outputted together with the data for the position of the cursor, to the post-processing unit 34.

[0182]

Thereafter, a process is executed to release the work area, and the like, used by this process, terminating the process.

[0183]

The post-processing unit 34, based on the location information that indicates the position of the insertion cursor, evaluates the insertion position for the post-processing character string data that corresponds to the insertion text character string. Given this, based on the evaluation result the post-processing character string data is inserted at a data position corresponding to the cursor position in the post-processing character string storing area of the post-processing unit 34.

[0184]

For example, explaining the case in FIG. 10 and FIG. 11, the cursor is at a position to the right of "た" in the "失われた記憶を" character string. Because of this, the cursor position is to the right of the fourth character, so the data for the insertion cursor position is 4, indicating the fourth character.

[0185]

If the "失われた記憶を" character string in FIG. 10 is the result of recognition in a handwritten character input, then the data for "失", which is the first character in post-processing, is structured from the first candidate "失" with the evaluation value of 98 thereof, and the second candidate "矢" with the evaluation value 90 thereof. Thereafter, similarly, the "た" that is the fourth character, for example, is the first candidate "た" with the evaluation value 97 thereof, the second candidate "E" with the evaluation value 60 thereof, and the third candidate "太" with the evaluation value 59 thereof. Moreover, the fifth character "の" is the first candidate "の" with the evaluation value of 97 thereof, the second candidate "0" with the evaluation value 80 thereof, and a third candidate "o" with the evaluation value 78 thereof.

[0186]

For the post-processing character string data that is in this state, the "都" character code and with evaluation value 100 thereof, and "市" character code with the evaluation value 100 thereof, are inserted. Because here the insertion position is after the data for the fourth character, the insertion will be after the third candidate "太" that is the last data for the fourth character "た", where the data is the first candidate "都" with the evaluation value 100 thereof.

[0187]

(27)

Following this, the sixth character will be "市", where the data thereof is the first candidate "市" with the evaluation value 100 thereof. The seventh character will be "の", with the data thereof being the first candidate "の" with the evaluation value 97 thereof, the second candidate "0" with the evaluation value 80 thereof, and the third candidate "o" with the evaluation value 78 thereof. The data thereafter follows sequentially from there.

[0188]

The content of the post-processing character string data that is to be processed by the post-processing unit 34 is updated as described above. That is, here, post-processing character string data is generated in a state wherein a character as a result of handwritten character recognition and an inserted text character string are mixed in one sentence.

[0189]

An example of inputting handwritten characters after the post-processing character string data has been updated will be explained next using FIG. 12.

[0190]

In FIG. 12, when the handwritten character "思い" is inputted into regions 67a and 67b for the handwritten character input 67, then, in the same process explained in the first embodiment, a handwritten character recognition process is executed first. In this case, a first candidate "思" with the evaluation value 98 thereof, a second candidate "恵" with the evaluation value 67 thereof, are obtained for the recognition result for the "思" and the first candidate "い" and the evaluation value of 95 thereof and the second candidate "ゝ" and the evaluation value 90 thereof are obtained for the recognition result for "い".

[0191]

When the handwritten "思い" is inputted, the post-processing character string data in the post-processing unit 34 will be "失われた都市の記憶を", where the cursor position for inserting the recognition result of the handwritten character will be after the "を".

[0192]

In this case, the data for the first candidate "思" (evaluation value = 98) and second candidate "恵" (evaluation value = 67), before the "思", and the first candidate "い" (evaluation value = 95) and the second candidate "ゝ" (evaluation value = 90) for the recognition result for "い" will be inserted after the first candidate "を" (evaluation value = 96) and the second candidate "あ" (evaluation value = 77) for the "を" that is the 10th character in the post-processing character string data.

[0193]

Based on the post-processing character string data that has been processed in this way (the "失われた都市の記憶を思い"), the post-processing explained in the first embodiment is performed to replace the displayed candidate for the recognition candidate, and is displayed in the input character buffer areas 61.

[0194]

Explaining briefly the post-processing on the post-processing character string data thereafter, in the "失われた都市の記憶を思い" post-processing character string data, the "失われた", "記憶を", and "思い" are subject to the processing because they are the recognition results for handwritten inputs, and there are sets of character codes, comprising a plurality of candidates for each, where "都市" is inputted as an inserted text character string data, so there is only a single character code for a single character.

[0195]

That is, the post-processing character string data is produced as a combination of character string (character code) candidate strings, as depicted in FIG. 14. Additionally, the post-processing unit 34 determines, based on the post-processing dictionary for chain probabilities, the candidate character string that has the highest chain probability. Note that for the method of calculating the chain probabilities a conventional well-known method may be used, so the explanation thereof will be omitted here.

[0196]

As explained above, the provision of the structure, in the second embodiment, for inserting a handwritten character into a character recognition result editing (confirming) region (input character buffer region) for input enables insertion of a text character string that is generated by another application, or a text character string that is stored in a system character string storage area, during the process of inputting a handwritten character string.

[0197]

Through this, it is possible to provide an information processing device equipped with a handwritten character recognition function that enables post-processing to function, without problems, for character strings that are a mixture of character strings consisting of recognition results for handwritten character input together with text character strings that are inserted separately.

[0198]

More specifically, even when, for example, a kanji character string through hiragana-kanji conversion, created in a word processor, for example, is inputted after inputting a character string that is inputted through handwriting, after which a character string is further inputted through handwriting, post-processing for example, a chain probability calculating process in the second embodiment, can be executed on an inputted character string as a single character string that is to be subjected to processing. This enables post-processing to be achieved efficiently and by an appropriate processing unit, on the character string that is intended by the user, making it possible to provide a character recognition environment with a higher recognition rate.

[0199]

While in the above exemplary embodiments were described in detail, the present invention may be formed as, for example, a system, a device, a method, a program, a storage medium, or the like, and, specifically, may also be applied as a system that is structured from a plurality of devices, and may be applied as a device that is constituted from a single device.

[0200]

Note that the present invention includes a case that is achieved by a software program for achieving the functions of the embodiments, set forth above (a program corresponding to the flowchart that is shown in a diagram in an embodiment), supplied either directly or remotely to a system or device, where a computer of the system or device reads out and executes the supplied program code.

[0201]

Consequently, in order to achieve, using a computer, the functional processes of the present invention, the program code itself that is installed in the computer is an implementation of the present invention, that is, the present invention includes also the computer program itself for achieving the functional processes of the present invention.

[0202]

In this case, insofar as it has the functions of the program, the form may be that of object code, a program that is executed by an interpreter, script data that is supplied to an operating system, or the like.

[0203]

The recording medium for supplying the program may be, for example, a Floppy (registered trademark) disk, a hard disk, an optical disc, a magneto-optical disc, an MO, a CD-ROM, a CD-R, a CD-RW, a magnetic tape, a nonvolatile memory card, a ROM, a DVD (a DVD-ROM, DVD-R), or the like.

[0204]

Additionally, the method for supplying the program may be that of using a browser of a client computer to connect to a webpage on the Internet, where the computer program itself of the present invention, or a compressed file that includes an automatic installation function, may be supplied from the webpage through downloading to a recording medium such as a hard disk, or the like. Moreover, the embodiment may be achieved through dividing the program code that structures the program of the present invention may be divided into a plurality of files and each of these files may be downloaded from different webpages. That is, a web server for downloading, for a plurality of users, program files for achieving, on a computer, the functional processes of the present invention, is also included in the present invention.

[0205]

Moreover, the program of the present invention can also be implemented through encrypting and storing onto a storage medium, such as a CD-ROM, and distributing to users, where a user that satisfies prescribed conditions is allowed to download, from a webpage on the Internet, key information for decryption, and using the key information to execute the encrypted program to install it on a computer.

[0206]

Moreover, the functions of the embodiments, described above, may be achieved through not only achieving the functions of the embodiments through a computer reading out and executing the program, but also, based on instructions of that program, achieving the functions of the embodiments described above through the operating system that runs on the computer carrying out some or all of the actual processes.

[0207]

Moreover, the functions of the embodiments described above may be implemented through writing, to a memory that is equipped in a function expansion board that is inserted in a computer or a function expansion unit that is connected to a computer, a program that is read out from a recording medium, and then, based on instructions of that program, carrying out some or all of the actual processes by CPUs, or the like, equipped in the function expansion board or function expansion unit.

[BRIEF DESCRIPTIONS OF THE DRAWINGS]

[0208]

FIG. 1 is a diagram depicting an example of a form of use of a character recognition device according to a first embodiment according to the present invention.

FIG. 2 is a diagram depicting a hardware structure for a mobile information device that functions as the character recognition device according to the first embodiment according to the present invention.

FIG. 3 is a diagram depicting a functional structure for the mobile information device of the first embodiment according to the present invention.

FIG. 4 is a diagram depicting a post-processing continuation determination table of the first embodiment according to the present invention.

FIG. 5 is a diagram for explaining a specific example of a character recognition process in the first embodiment according to the present invention.

FIG. 6 is a diagram depicting an example of a handwritten character input screen generated by a handwritten character input utility program according to the first embodiment according to the present invention.

FIG. 7 is a flowchart depicting a character string post-processing continuation evaluating process according to the first embodiment according to the present invention.

FIG. 8 is a flowchart depicting a handwritten character recognition process according to the first embodiment according to the present invention.

FIG. 9 is a diagram depicting a functional structure for a mobile information device according to a second embodiment according to the present invention.

FIG. 10 is a diagram for explaining a specific example of a character recognition process in the second embodiment according to the present invention.

FIG. 11 is a diagram for explaining a specific example of a character recognition process in the second embodiment according to the present invention.

FIG. 12 is a diagram for explaining a specific example of a character recognition process in the second embodiment according to the present invention.

FIG. 13 is a flowchart depicting the insertion process in the second embodiment according to the present invention.

FIG. 14 is a diagram depicting a combination string of a character string that is to be subject to post-processing in the second embodiment according to the present invention.

[Explanations of Reference Symbols]

[0209]

1: Mobile Information Device

2: Displaying Unit

3: Stylus

4: Camera Unit

4a: CCD

5: Coordinate Inputting Unit

6: CPU

7: RAM

8: ROM

9: Media Interface

10: Keyboard

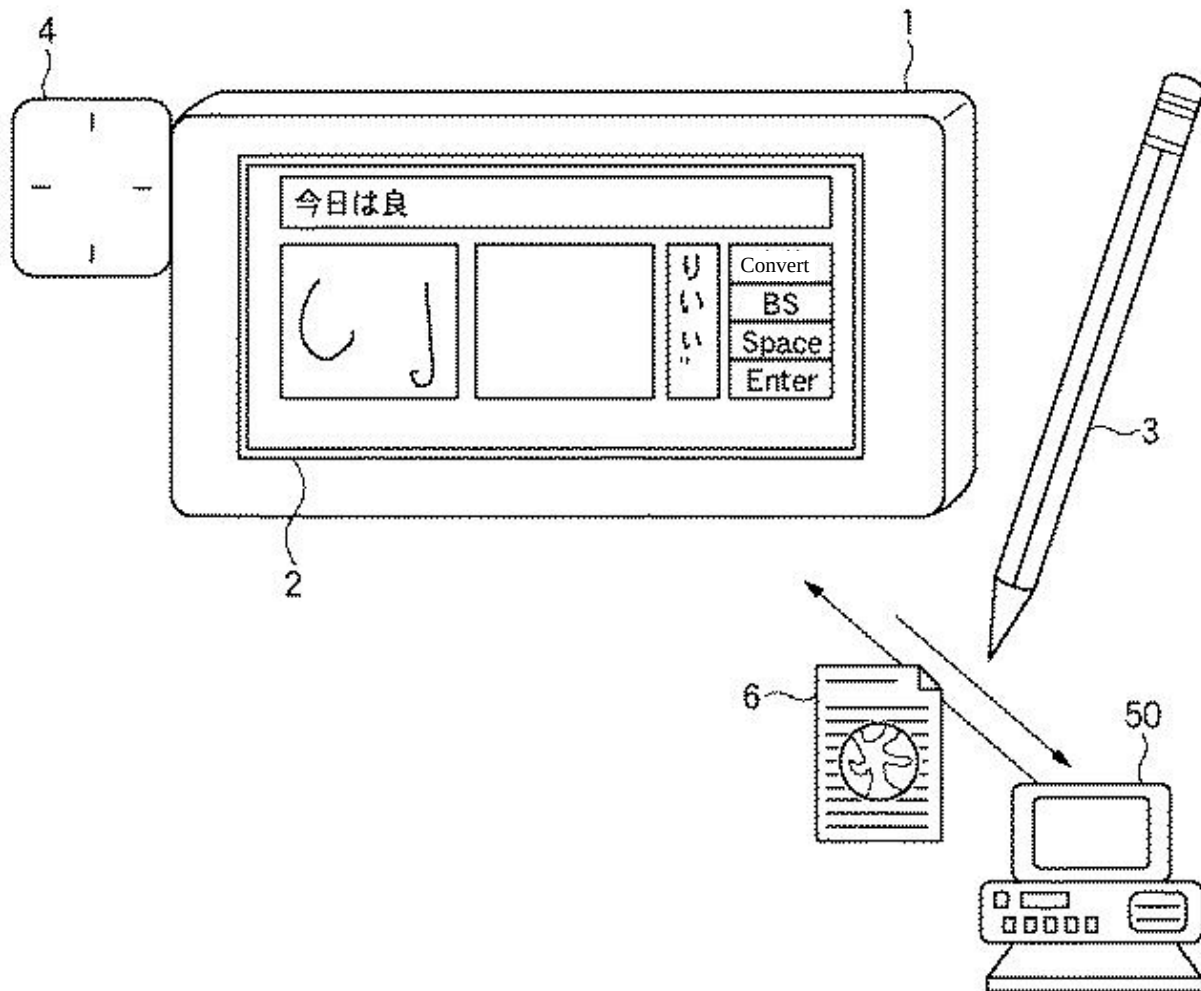
11: External Interface

12: Network Interface

13: System Bus

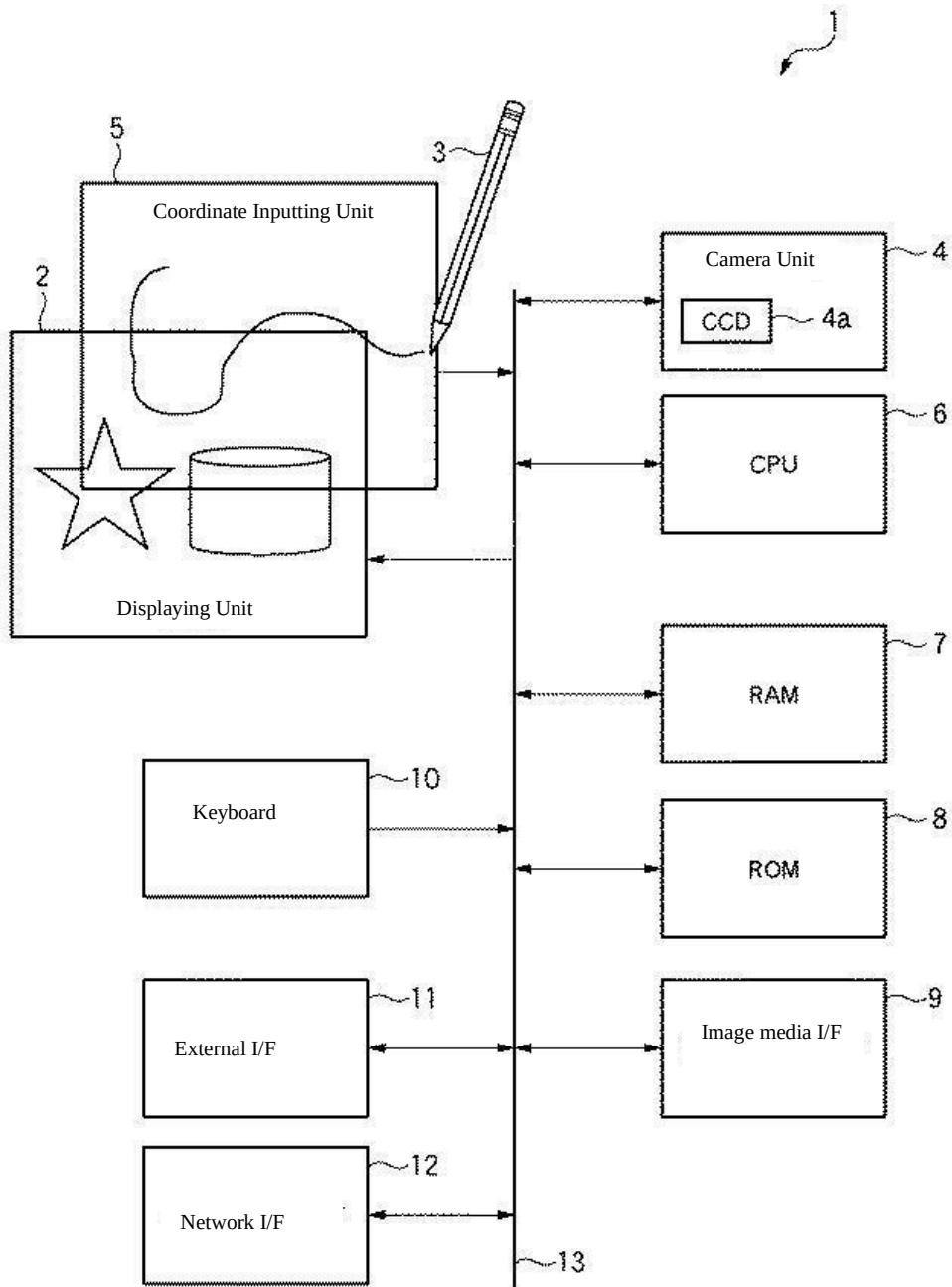
(31)

[FIG. 1]



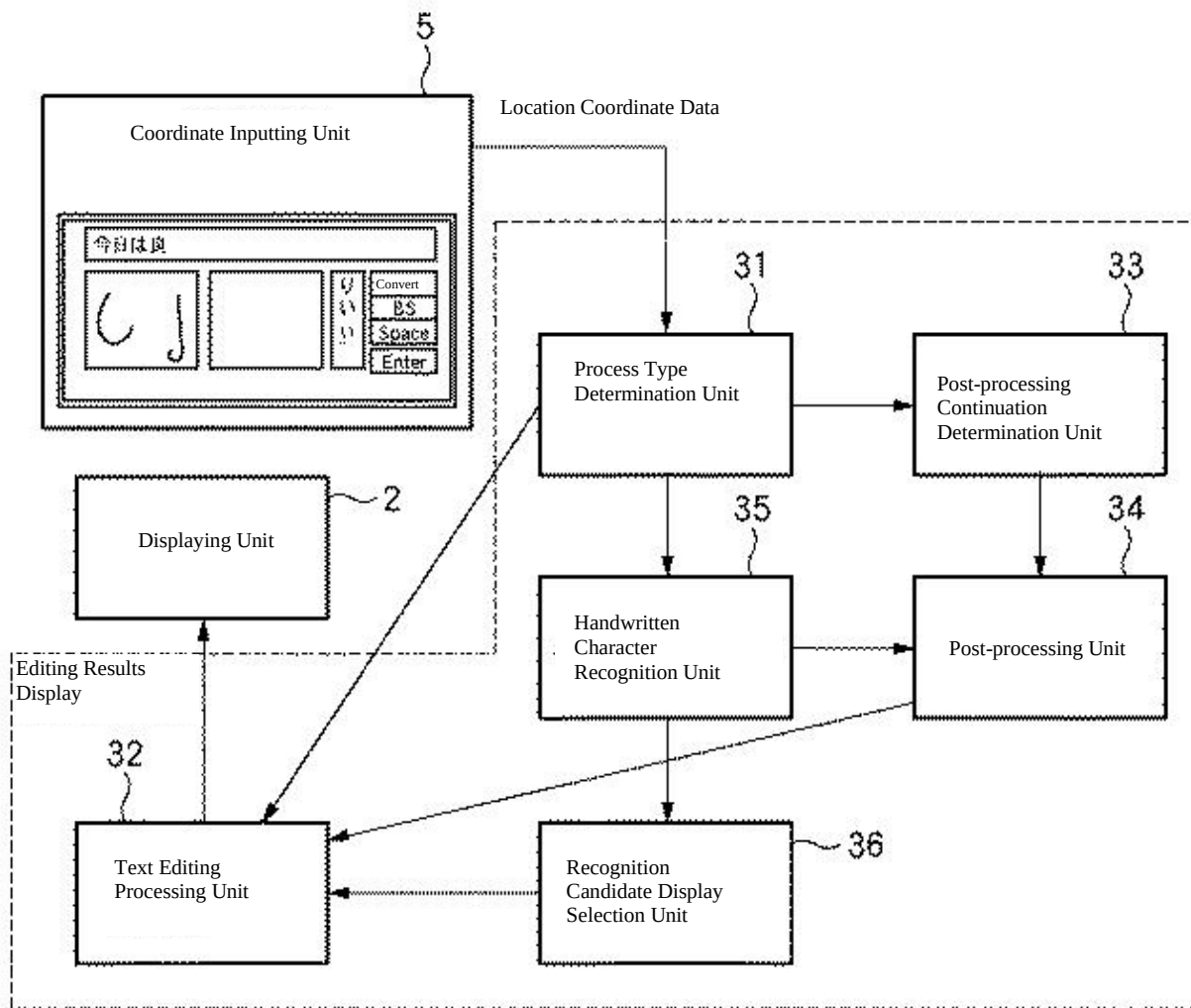
(32)

[FIG. 2]



(33)

[FIG. 3]



(34)

[FIG. 4]

Process Type	Post-processing Continuation
Candidate Change process	Post-processing Character String Continuation Termination
Handwritten Character Input	Post-processing Character String Continuation
Character Cutoff Operation	Post-processing Character String Continuation
Kana-Kanji Conversion Process Instruction	Post-processing Character String Continuation Termination
Single-Character Deleting Process	Post-processing Character String Continuation
Space Input	Post-processing Character String Continuation Termination
Enter Input	Post-processing Character String Continuation Termination
Switch Application	Post-processing Character String Continuation

(35)

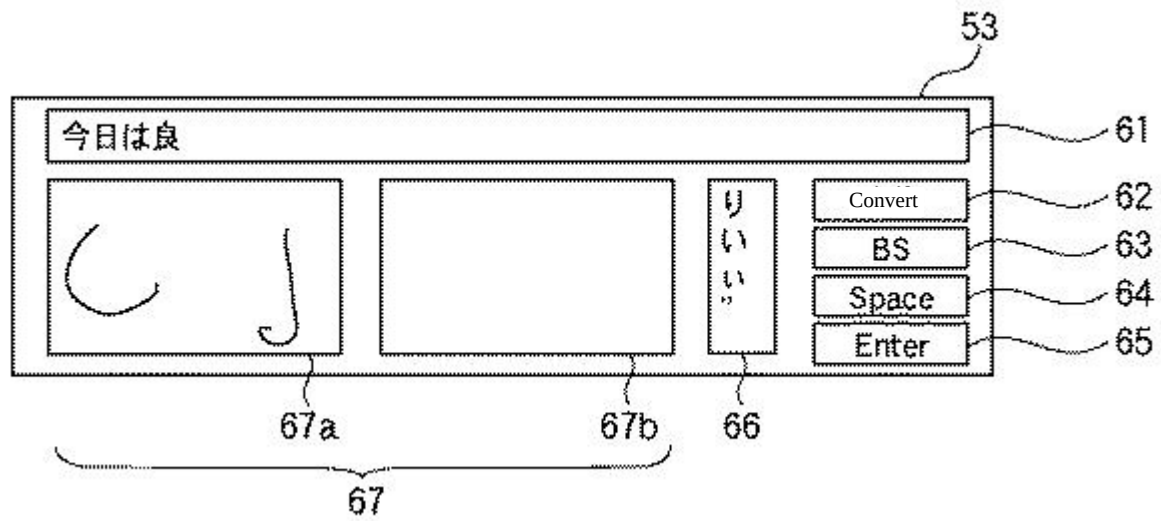
[FIG. 5]

The diagram illustrates a 'New Message Creation Screen' (51) with the following components:

- Header (51):** Contains email fields: To: asa@uso.com, From: ara@uso.com, Subject: Greetings, Cc:, Bcc:, and Attached:.
- Body (52):** Contains the text 'Good morning.' followed by a cursor '|'. Below this is a large text input area (53) containing the handwritten Japanese text '今日は良' (Today is good).
- Bottom Bar (54-59):** A horizontal row of icons and buttons:
 - 54: A square icon with a diagonal hatching pattern and the letter 'A'.
 - 55: A square icon with a diagonal hatching pattern.
 - 56: A button labeled 'File Browser'.
 - 57: A button labeled 'Email'.
 - 58: A button labeled 'Photo Viewer'.
 - 59: A button labeled 'Scheduler'.
- Input Area (53):** A detailed view of the text input area showing:
 - A large rectangular box for text entry.
 - A smaller box below it containing the handwritten characters 'C' and 'J'.
 - A vertical column of three small boxes containing the Japanese characters 'り', 'い', and 'い'.
 - A vertical stack of four buttons on the right: 'Convert', 'BS', 'Space', and 'Enter'.

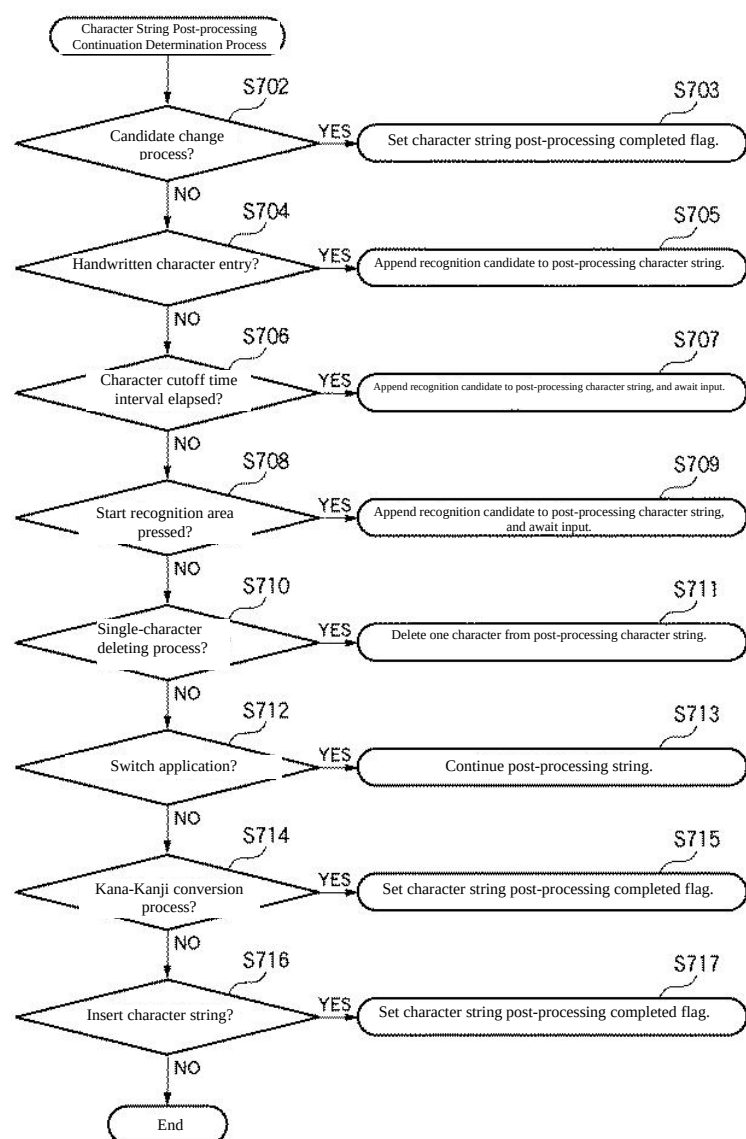
(36)

[FIG. 6]



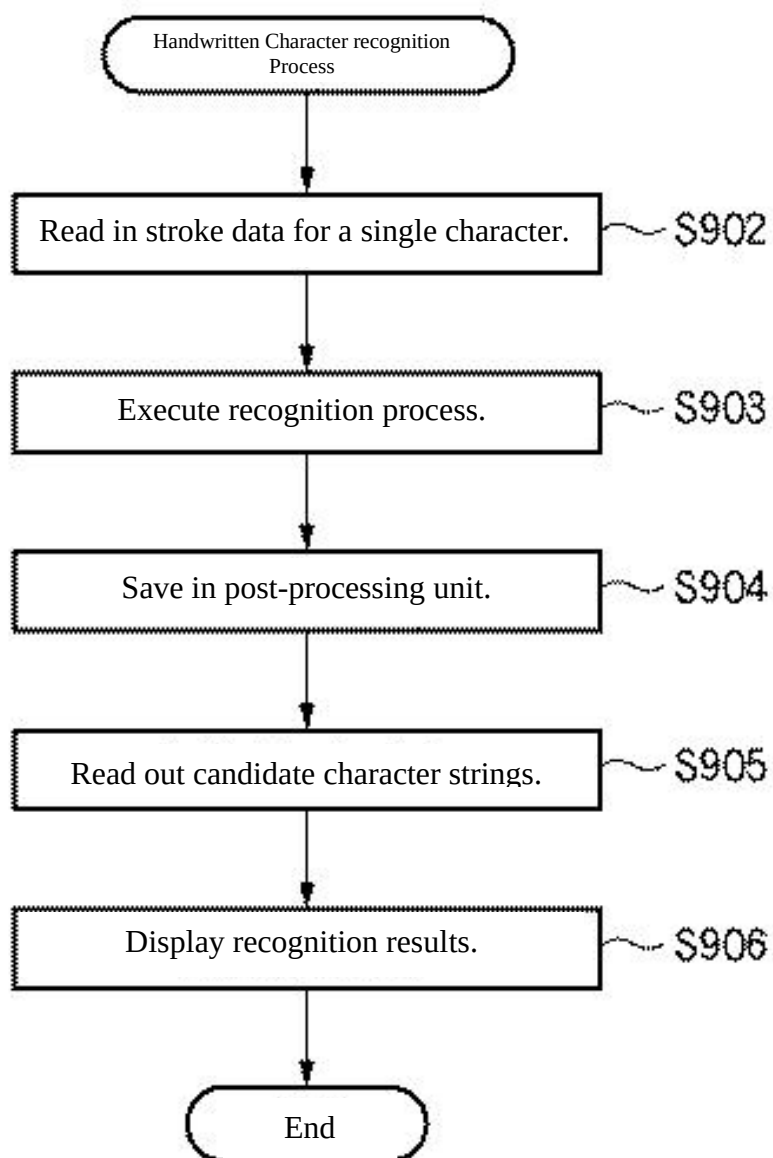
(37)

[FIG. 7]



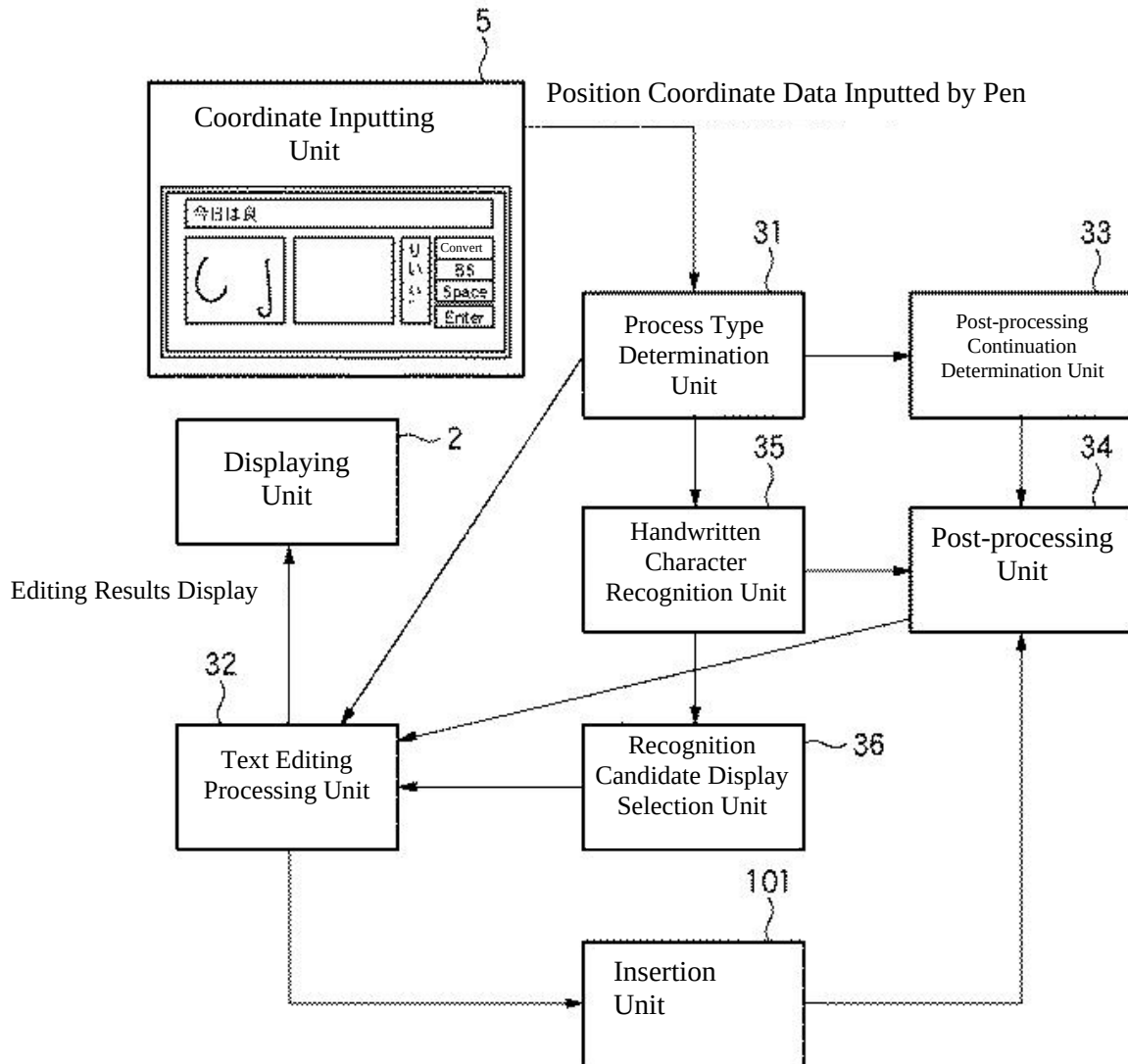
(38)

[FIG. 8]



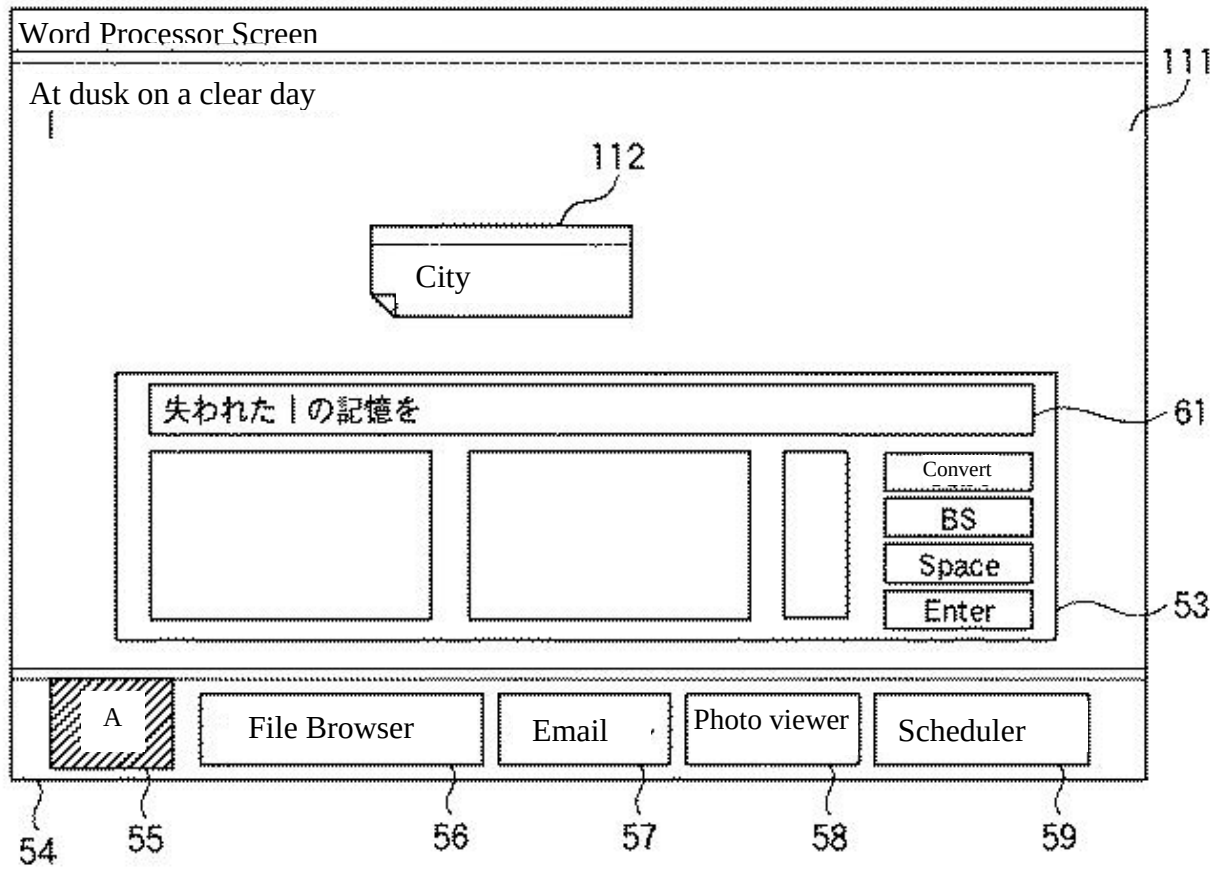
(39)

[FIG. 9]



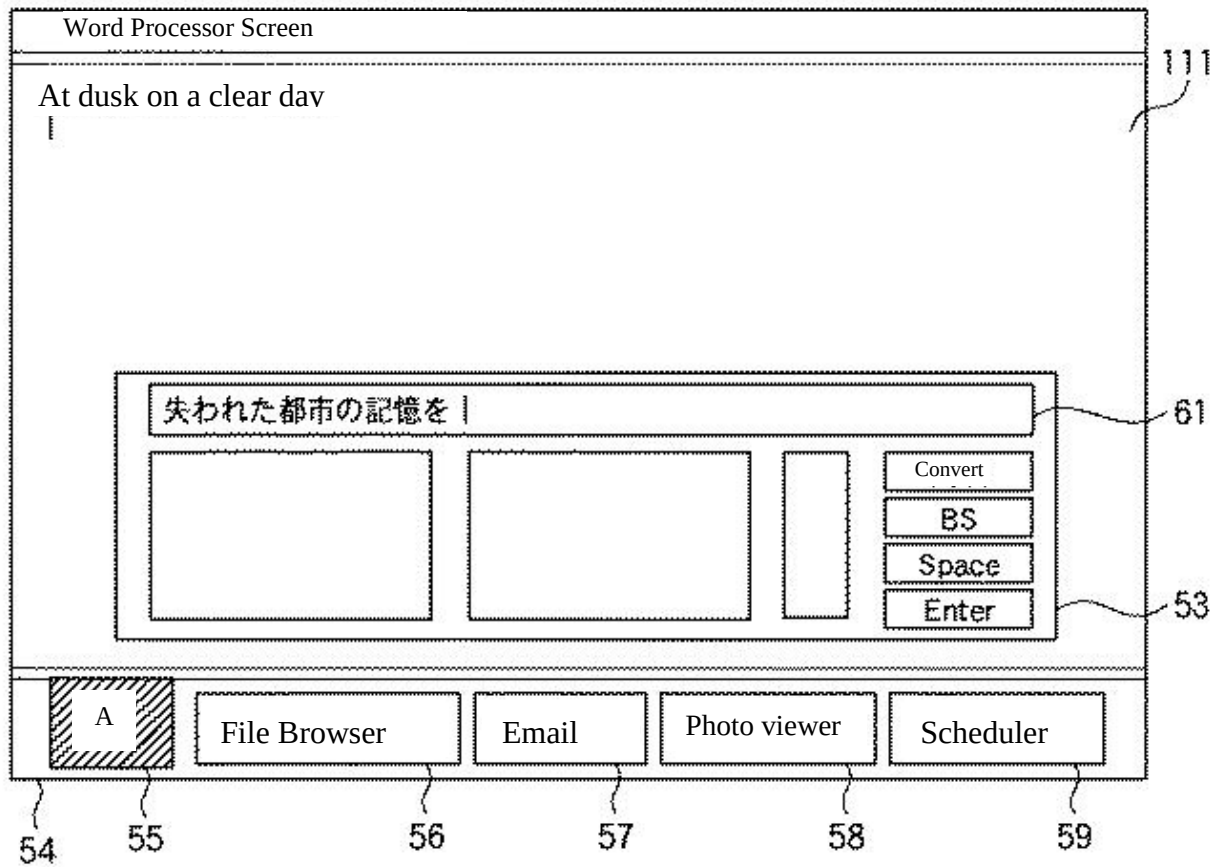
(40)

[FIG. 10]



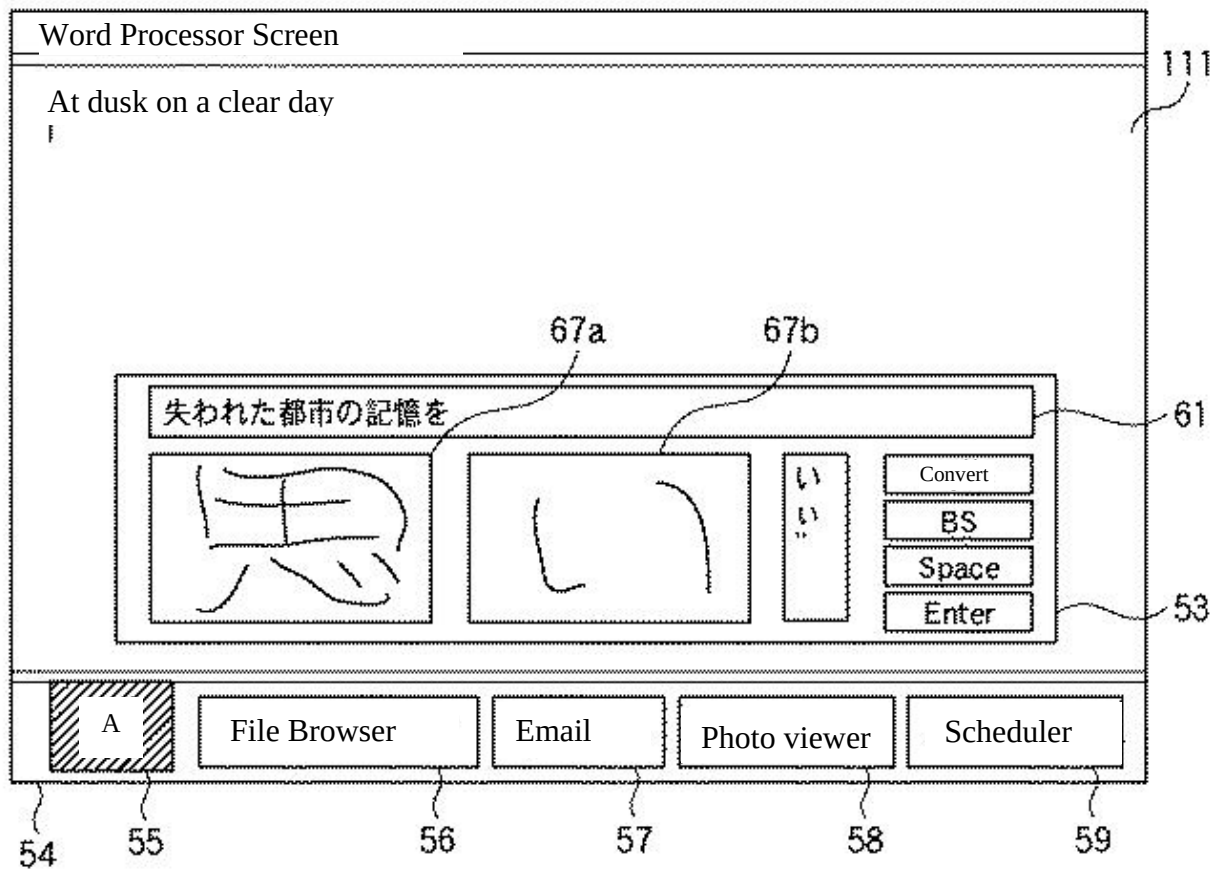
(41)

[FIG. 11]



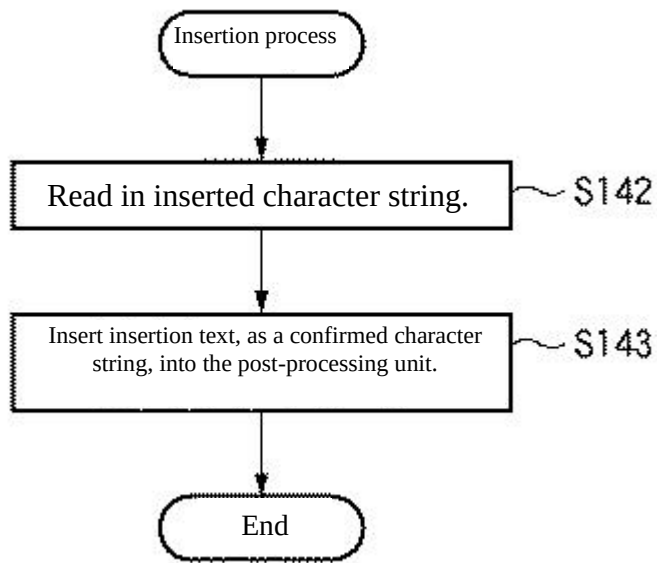
(42)

[FIG. 12]

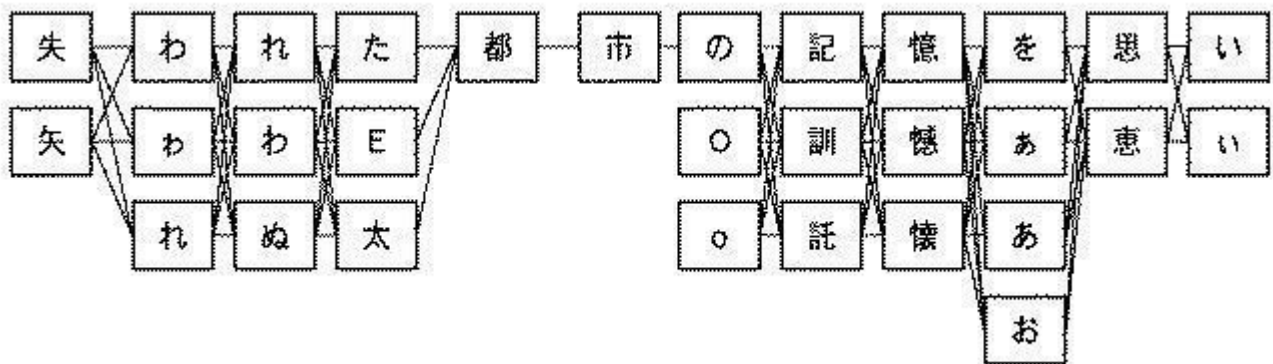


(43)

[FIG. 13]



[FIG. 14]



(44)

Continued from front page.

F Terms (Reference)

5B064 AB04 BA05 BA06 CA11 DD02 DD11 DD16 EA08 EA16 EA18

FA04 FA05 FA13

5B068 BB06 CC19

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2006-323502

(P2006-323502A)

(43) 公開日 平成18年11月30日(2006.11.30)

(51) Int. Cl.	F I	テーマコード (参考)
G06K 9/03 (2006.01)	G06K 9/03 B	5B009
G06F 3/041 (2006.01)	G06F 3/03 38OR	5B064
G06F 17/22 (2006.01)	G06F 17/22 514A	5B068
G06K 9/62 (2006.01)	G06K 9/62 G	
G06K 9/72 (2006.01)	G06K 9/72 Z	
審査請求 未請求 請求項の数 10 O L (全 28 頁)		

(21) 出願番号 特願2005-144227 (P2005-144227)
 (22) 出願日 平成17年5月17日 (2005.5.17)

(71) 出願人 000001007
 キヤノン株式会社
 東京都大田区下丸子3丁目30番2号
 (74) 代理人 100076428
 弁理士 大塚 康德
 (74) 代理人 100112508
 弁理士 高柳 司郎
 (74) 代理人 100115071
 弁理士 大塚 康弘
 (74) 代理人 100116894
 弁理士 木村 秀二
 (72) 発明者 新井 常一
 東京都大田区下丸子3丁目30番2号 キ
 ヤノン株式会社内
 Fターム(参考) 5B009 LA01 MB03

最終頁に続く

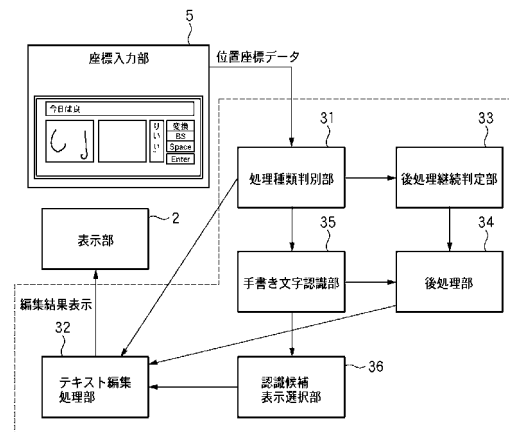
(54) 【発明の名称】 情報処理装置及びその制御方法、プログラム

(57) 【要約】

【課題】 効率的にかつ精度良く手書き文字認識を実行することができる情報処理装置及びその制御方法、プログラムを提供する。

【解決手段】 手入力された軌跡データに対応する文字を認識する。その文字認識結果の候補文字列を後処理する。入力処理、文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示による処理種類に基づいて、新たに入力される文字を、後処理対象の文字列に継続させるか否かを判定する。

【選択図】 図3



【特許請求の範囲】

【請求項 1】

手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置であって、

前記軌跡データを入力する入力手段と、

前記入力手段で入力された軌跡データに対応する文字を認識する文字認識手段と、

前記文字認識手段による文字認識結果の候補文字列を後処理する後処理手段と、

前記入力手段による入力処理、前記文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示手段と、

前記指示手段によって実行が指示される処理種類に基づいて、該指示手段による指示以降に入力される新たな文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させるか否かを判定する判定手段と

を備えることを特徴とする情報処理装置。

【請求項 2】

前記新たに入力される文字として、それに対応する文字コードが確定している文字を挿入する挿入手段とを更に備える

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】

前記判定手段は、前記処理種類が前記入力手段による入力処理である場合には、該入力手段で入力された軌跡データに対する前記文字認識手段による文字認識結果として新たに入力される候補文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させる

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 4】

前記入力手段と、前記指示手段と、及び前記文字認識手段による文字認識結果の候補文字を表示する候補表示領域を含む操作画面と、当該情報処理装置上で動作するアプリケーションが生成するアプリケーション画面と、当該情報処理装置上で動作する複数種類のアプリケーションを切替るための切替操作画面を表示する表示手段とを更に備える

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 5】

前記入力手段は、前記軌跡データを入力する入力領域を備え、

前記指示手段は、

前記候補表示領域上の候補文字を変更するための変更部と、

前記候補表示領域上の候補文字を仮名漢字変換するための変換部と、

前記候補表示領域上で表示されるカーソルが位置する前の候補文字を削除するための削除部と、

前記候補表示領域上で空白文字を挿入するための空白文字挿入部と、

前記候補表示領域上の候補文字列を確定して前記アプリケーション画面上に、該候補文字列を挿入するための候補文字列確定部と

を備えることを特徴とする請求項 4 に記載の情報処理装置。

【請求項 6】

前記判定手段は、前記処理種類が、前記入力領域に対する入力処理、前記入力領域に対する未入力時間が所定時間経過した場合に発生するタイマーイベント処理、前記削除部による削除処理、前記切替操作画面による切替処理のいずれかである場合には、該入力領域に入力された軌跡データに対する前記文字認識手段による文字認識結果として新たに入力される候補文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させる

ことを特徴とする請求項 5 に記載の情報処理装置。

【請求項 7】

当該情報処理装置を制御するオペレーティングシステムが内部メモリに確保する記憶領域

域に記憶された文字を挿入する挿入手段を更に備え、

前記判定手段は、前記処理手種類が前記挿入手段による挿入処理である場合には、その挿入処理で新たに入力される文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させる

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 8】

前記後処理手段は、前記文字認識結果の候補文字列間の連鎖確率を算出する

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 9】

手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置の制御方法であって、

前記軌跡データを入力する入力工程と、

前記入力工程で入力された軌跡データに対応する文字を認識する文字認識工程と、

前記文字認識工程による文字認識結果の候補文字列を後処理する後処理工程と、

前記入力工程による入力処理、前記文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示工程と、

前記指示工程によって実行が指示される処理種類に基づいて、該指示工程による指示以降に入力される新たな文字を、前記後処理工程で既に後処理対象となっている後処理文字列に継続させるか否かを判定する判定工程と

を備えることを特徴とする情報処理装置の制御方法。

【請求項 10】

手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置の制御を実現するプログラムであって、

前記軌跡データを入力する入力工程のプログラムコードと、

前記入力工程で入力された軌跡データに対応する文字を認識する文字認識工程のプログラムコードと、

前記文字認識工程による文字認識結果の候補文字列を後処理する後処理工程のプログラムコードと、

前記入力工程による入力処理、前記文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示工程のプログラムコードと、

前記指示工程によって実行が指示される処理種類に基づいて、該指示工程による指示以降に入力される新たな文字を、前記後処理工程で既に後処理対象となっている後処理文字列に継続させるか否かを判定する判定工程のプログラムコードと

を備えることを特徴とするプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置及びその制御方法、プログラムに関するものである。

【背景技術】

【0002】

従来より、入力された手書き文字の軌跡（ストローク）を認識する文字認識装置が存在する。この種の文字認識装置（例えば、特許文献 1）では、単語の区切りを指定させるために、ユーザに、文字入力枠近傍の特定エリアにストロークを入力させ指定させることで、その区切りの位置を明確に指示させる必要がある。

【0003】

また、文字列単位で後処理を実行するオンライン手書き文字認識装置では、認識開始ボタンが押されるまでに入力された手書き文字列を、後処理の対象とする構成としている。

【特許文献 1】特開平 6-208655 号公報

【発明の開示】

【発明が解決しようとする課題】

【0004】

しかしながら、従来の文字認識装置では、ユーザが都度、単語の区切りを入力指示しない限り、装置側は単語の区切りを認識できない。そのため、ユーザは、入力時には、本来意図する文字入力以外の入力作業を強いられ、自然な使い勝手が損なわれていた。

【0005】

また、文字列単位での後処理を行う場合は、認識開始ボタンが押されるまでに入力された手書き文字列を後処理対象としている。そのため、文字列の入力途中で認識開始ボタンを押してしまった場合、その後に入力された文字は、認識されてしまった文字とは関係されず、認識率の低下あるいは意図する認識結果が得られない場合がある。

10

【0006】

本発明は上記の課題を解決するためになされたものであり、効率的にかつ精度良く手書き文字認識を実行することができる情報処理装置及びその制御方法、プログラムを提供することを目的とする。

【課題を解決するための手段】

【0007】

上記の目的を達成するための本発明による情報処理装置は以下の構成を備える。即ち、手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置であって、

20

前記軌跡データを入力する入力手段と、

前記入力手段で入力された軌跡データに対応する文字を認識する文字認識手段と、

前記文字認識手段による文字認識結果の候補文字列を後処理する後処理手段と、

前記入力手段による入力処理、前記文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示手段と、

前記指示手段によって実行が指示される処理種類に基づいて、該指示手段による指示以降に入力される新たな文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させるか否かを判定する判定手段と

を備える。

【0008】

また、好ましくは、前記新たに入力される文字として、それに対応する文字コードが確定している文字を挿入する挿入手段とを更に備える。

30

【0009】

また、好ましくは、前記判定手段は、前記処理種類が前記入力手段による入力処理である場合には、該入力手段で入力された軌跡データに対する前記文字認識手段による文字認識結果として新たに入力される候補文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させる。

【0010】

また、好ましくは、前記入力手段と、前記指示手段と、及び前記文字認識手段による文字認識結果の候補文字を表示する候補表示領域を含む操作画面と、当該情報処理装置上で動作するアプリケーションが生成するアプリケーション画面と、当該情報処理装置上で動作する複数種類のアプリケーションを切替るための切替操作画面を表示する表示手段を更に備える。

40

【0011】

また、好ましくは、前記入力手段は、前記軌跡データを入力する入力領域を備え、前記指示手段は、

前記候補表示領域上の候補文字を変更するための変更部と、

前記候補表示領域上の候補文字を仮名漢字変換するための変換部と、

前記候補表示領域上で表示されるカーソルが位置する前の候補文字を削除するための削除部と、

前記候補表示領域上で空白文字を挿入するための空白文字挿入部と、

50

前記候補表示領域上の候補文字列を確定して前記アプリケーション画面上に、該候補文字列を挿入するための候補文字列確定部と
を備える。

【0012】

また、好ましくは、前記判定手段は、前記処理種類が、前記入力領域に対する入力処理、前記入力領域に対する未入力時間が所定時間経過した場合に発生するタイマーイベント処理、前記削除部による削除処理、前記切替操作画面による切替処理のいずれかである場合には、該入力領域に入力された軌跡データに対する前記文字認識手段による文字認識結果として新たに入力される候補文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させる。

10

【0013】

また、好ましくは、当該情報処理装置を制御するオペレーティングシステムが内部メモリに確保する記憶領域に記憶された文字を挿入する挿入手段を更に備え、

前記判定手段は、前記処理種類が前記挿入手段による挿入処理である場合には、その挿入処理で新たに入力される文字を、前記後処理手段で既に後処理対象となっている後処理文字列に継続させる。

【0014】

また、好ましくは、前記後処理手段は、前記文字認識結果の候補文字列間の連鎖確率を算出する。

【0015】

上記の目的を達成するための本発明による情報処理装置の制御方法は以下の構成を備える。即ち、

20

手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置の制御方法であって、

前記軌跡データを入力する入力工程と、

前記入力工程で入力された軌跡データに対応する文字を認識する文字認識工程と、

前記文字認識工程による文字認識結果の候補文字列を後処理する後処理工程と、

前記入力工程による入力処理、前記文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示工程と、

前記指示工程によって実行が指示される処理種類に基づいて、該指示工程による指示以降に入力される新たな文字を、前記後処理工程で既に後処理対象となっている後処理文字列に継続させるか否かを判定する判定工程と

30

を備える。

【0016】

上記の目的を達成するための本発明によるプログラムは以下の構成を備える。即ち、

手書き文字の軌跡データを入力して文字認識し、その文字認識結果に対して後処理を実行する情報処理装置の制御を実現するプログラムであって、

前記軌跡データを入力する入力工程のプログラムコードと、

前記入力工程で入力された軌跡データに対応する文字を認識する文字認識工程のプログラムコードと、

40

前記文字認識工程による文字認識結果の候補文字列を後処理する後処理工程のプログラムコードと、

前記入力工程による入力処理、前記文字認識結果の候補に対する処理を含む各種処理の実行を指示するための指示工程のプログラムコードと、

前記指示工程によって実行が指示される処理種類に基づいて、該指示工程による指示以降に入力される新たな文字を、前記後処理工程で既に後処理対象となっている後処理文字列に継続させるか否かを判定する判定工程のプログラムコードと

を備える。

【発明の効果】

【0017】

50

本発明によれば、効率的にかつ精度良く手書き文字認識を実行することができる情報処理装置及びその制御方法、プログラムを提供できる。

【発明を実施するための最良の形態】

【0018】

以下、本発明の実施の形態について図面を用いて詳細に説明する。

【0019】

本発明では、手書き入力による文字列を、その文字列単位で後処理を実行する情報処理装置において、ユーザの指示入力の種類に応じた後処理継続判定部を設ける。そして、この後処理継続判定部によって、ユーザが文章の終了や継続等の指示入力を、特に意識して指示しなくても、自然にかつ適切な処理単位で後処理を実現する構成について説明する。これにより、本発明では、ユーザが入力する手書き入力に対する認識結果となる文字が、ユーザが意図する認識結果となる確率を高めることが実現できる。 10

【0020】

また、文字コードで構成される一般的なテキスト文字列（文字コード列）を、後処理対象の文字列の一部として挿入する挿入部を設ける。そして、この挿入部によって、後処理対象の文字認識結果編集領域にテキスト文字列を挿入した場合でも、正常に後処理を実現する構成について説明する。これにより、手書き文字の認識結果と、一般的なテキスト文字列が混在する場合でも、ユーザの意図する文章の入力効率を向上させることができる。

【0021】

<実施形態1>

20

図1は本発明の実施形態1の文字認識装置の使用形態の一例を示す図である。また、図2は本発明の実施形態1の文字認識装置として機能する情報携帯機器のハードウェア構成を示す図である。

【0022】

1は、PDA（パーソナルデジタルアシスタント）、携帯電話、デジタルカメラ、ノートPC（パーソナルコンピュータ）等の携帯情報機器である。

【0023】

この携帯情報機器1は、表示部2、指示具3、カメラ部4、座標入力部5、CPU6、RAM7、ROM8、記憶メディア（例えば、スマートメディア、コンパクトフラッシュ（登録商標）等）を差し込むためのスロットを備えるメディアI/F（インタフェース）9、キーボード（表示部2上で表示されるソフトキーボード、あるいは通常のキーボード）10、他の情報機器と接続するための外部I/F（USB、Bluetooth等）11、ネットワークI/F12を備えている。 30

【0024】

携帯情報機器1は、カメラ部4での撮影画像を液晶表示部2に表示し、その画像に付加するメモ情報等の付加情報を、座標入力部5によって入力する軌跡データ（ユーザの手書き入力操作による手書き軌跡データ）によって作成することができる。

【0025】

CPU6は、システムバス13を介して、RAM7、ROM8、座標入力部5、表示部2等の携帯情報機器1の各種構成要素と接続している。また、CPU6は、ROM8に記憶されているプログラムによって各種制御を実行する。 40

【0026】

RAM7は、画像データ、入力された軌跡データ及びその位置情報等の各種データを記憶するとともに、データの作業領域、一時待避領域として機能する。

【0027】

ROM8は、本発明を実現する各種プログラムを含む携帯情報機器の制御プログラムが記憶されている。より詳しくは、ROM8は、後述する、手書き文字入力時の後処理の制御処理プログラムや、カメラ部制御プログラム、表示部制御プログラム、手書き軌跡データの認識プログラム等の各種プログラムに加えて、手書き文字認識用の認識用辞書データ、後処理用辞書データ（連鎖確率算出用辞書データ）、各種テーブル、設定データ等のデ 50

ータを記憶している。

【0028】

入力された軌跡データは、従来からある手書き文字認識機能によって、対応する文字コードに認識されて、携帯情報機器1に入力することができる。また、入力された軌跡データは、認識されずにそのままの状態の画像（イメージデータ）を付加情報として撮影画像に付加したり、文字認識機能によって得られるテキストデータを付加情報として撮影画像に付加することができる。

【0029】

この付加情報が付加された撮影画像の画像データはユーザの指示操作によって、外部I/F11を介して、パーソナルコンピュータ等の外部端末50へ送信することができる。または、その画像データを記憶メディア上に記憶した後、メディアスロットを備える外部端末50やプリンタに装着することで、画像データの保存や印刷を実行することができる。

【0030】

更に、携帯情報機器1で実現される電子メール機能によって作成された電子メールに画像データを添付した上で、ネットワークI/F12を介して接続されるローカルネットワークや公衆回線等のネットワークを通じて、ネットワーク上の他の外部端末へ送信することも可能である。

【0031】

2は表示部であり、例えば、液晶やLEDで構成される。表示部2が液晶表示部で構成される場合には、液晶表示素子、液晶制御回路、表示メモリを有する。この表示部2は、システムバス13経由でCPU6に接続している。また、CPU6からの指示で、画像データや軌跡データを表示したり、各種操作画面を表示する。

【0032】

この表示部2の上面に座標入力部5を重ねて配置することで、座標入力を行う一般的な透明抵抗膜デジタイザ（いわゆる、タッチパネル）を実現することができる。ユーザは、座標入力部5の座標入力面（表示部2の表示画面）上を指示具3で押圧することにより、軌跡データを入力することができる。

【0033】

より詳しくは、座標入力部5は内部の制御回路（不図示）を介して、システムバス13経由でCPU6に接続している。これにより、指示具3を使って、座標入力部5の座標入力面上を押圧することにより、押圧した位置の位置座標データがCPU6によって通知される。

【0034】

この位置座標データに基づいて、携帯情報機器1は、表示部2上で実現するGUI（グラフィックユーザインタフェース）によるファイル一覧表示画面等の操作画面を、ROM8内の処理手順に従って表示する。また、その入力された位置座標データが示す位置における操作画面上のデータファイルアイコン（データファイルを表す表示）に対応するデータファイルの内容を表示したり、その入力された位置に対応する画像を編集する等の各種処理を実行することができる。

【0035】

また、指示具3を用いることで、携帯情報機器1で実現される各種機能の実行指示や画面操作を実現することができ、例えば、上述の電子メール機能も、この指示具3を用いて使用することができる。

【0036】

3は指示具であり、手書きの軌跡データや、表示画面上のソフトボタン、ソフトキーボード等の入力に利用される。

【0037】

4はカメラ部であり、一般的なデジタルカメラの構成要素である、レンズ、CCD4a、CCD制御回路（不図示）等の構成要素を備えている。また、このカメラ部4は、携帯

情報機器 1 内の CPU 6 によって制御され、撮影された画像データは CPU 6 の制御の下、RAM 7 に一旦記憶される。このカメラ部 4 での撮影画像を、電子メールに添付して送信することができるのは、上述した通りである。

【0038】

より詳しくは、カメラ部 4 の CCD 制御回路がシステムバス 11 経由で CPU 6 に接続されている。そのため、この CPU 6 によって CCD 制御回路を介して CCD 6 の動作が制御され、CCD 6 による撮影画像が RAM 7 に記憶される。

【0039】

メディア I / F 9 は、デジタルカメラ等で一般的な記憶メディア、例えば、コンパクトフラッシュ（登録商標）カードのインタフェース部である。特に、メディア I / F 9 は、接続されたメディア中のデータの読出／書込を実現し、画像データ等のデータを記憶することができる。

10

【0040】

キーボード 10 は、一般的なキーボードであり、キーを押すことにより、英数字等のキーコードが入力される。入力されたキーコードは、システムバス 13 経由で CPU 6 が読み取る。

【0041】

外部 I / F 11 は、例えば、USB インタフェース部である。これは、CPU 6 によって制御され、パーソナルコンピュータ等の外部端末と接続して、軌跡データや画像データ等の各種データのやり取りを行うことができる。

20

【0042】

ネットワーク 1 / F 12 は、例えば、通信カードで構成される。これは、内部に PH S 回線等の公衆回線への接続コントローラやアンテナを内蔵している。通信カード内のコントローラに規定のコマンドを渡すことによって、公衆回線へ接続し、電子メールの送受信を実行できる。

【0043】

また、この通信カードを、ローカルエリアネットワークカード等のネットワークカードに変更すれば、ローカルエリアネットワーク接続に対応することも可能である。ここでの電子メールの送受信のプロトコル自体は、従来の方法と同様である。

【0044】

システムバス 13 は、CPU 6 と RAM 7、ROM 8 を含む携帯情報機器 1 の各種構成要素を相互に接続してデータの送受信を可能にする通信経路である。

30

【0045】

50 はパーソナルコンピュータ等の外部端末である。この外部端末 50 には、汎用コンピュータに搭載される標準的な構成要素（例えば、CPU、RAM、ROM、ハードディスク、外部記憶装置、ネットワークインタフェース、外部インタフェース、ディスプレイ、キーボード、マウス等）を有している。

【0046】

また、実施形態 1 では、外部端末 50 は、一般的な USB 等の外部インタフェースや、ネットワークを介して携帯情報機器 1 と接続し、画像データ 6 を送受信することができる。

40

【0047】

次に、実施形態 1 の携帯情報機器の機能構成について、図 3 を用いて説明する。

【0048】

図 3 は本発明の実施形態 1 の携帯情報機器の機能構成を示す図である。

【0049】

尚、図 3 では、携帯情報機器の各種機能構成（破線部で囲まれる）に加えて、説明の関係上、ハードウェアである座標入力部 5 及び表示部 2 を示している。この機能構成は、例えば、CPU 6 の制御の下、ROM 8 に記憶される制御プログラムによって実現されても良いし、その一部／全部が専用のハードウェア回路で実現されても良い。

50

【0050】

図3において、31は処理種類判別部であり、座標入力部2から受信する位置座標データの範囲に応じて処理種類の判別を行う。そして、その判別した処理種類に応じて、入力座標データを各処理部に振り分ける処理を行う。

【0051】

32はテキスト編集処理部であり、手書き文字認識部35によって得られるテキスト文字列に対して、削除や、仮名漢字変換や、アプリケーションの文字入力領域への挿入等の編集処理を実行する。

【0052】

33は後処理継続判定部であり、処理種類判別部31から受信する各処理種類に応じて、後処理文字列（後処理対象の文字列）の継続及び終了（中止）を判定し、その判定結果を後処理部34に送信する。 10

【0053】

34は後処理部である。ここでは、特に、文字列単位の後処理を実行し、そのアルゴリズムとしては、例えば、特開平10-198759号等に記載されるアルゴリズムと同様のアルゴリズムを使用することができる。具体的には、手書き文字認識部35で認識された候補文字列を、その文字同士の連鎖確率と、認識用辞書中の標準文字辞書データとの一致度等を加味して、入力された手書き文字列の認識候補を決定する。

【0054】

尚、後処理としては、これに限定されず、例えば、形態素解析や、統計処理等の、認識処理によって得られる文字列に対して後処理されるものであれば、どのようなものでも良い。 20

【0055】

35は手書き文字認識部であり、処理種類判別部31から、手書き文字の位置座標データ列を受信して、それと認識用辞書中の標準文字辞書データとのマッチング処理により、候補文字コードを出力する。

【0056】

36は認識候補表示選択部であり、手書き文字認識候補の選択指示処理を行う。この処理自体は、従来からある一般的な処理で、例えば、ある手書き文字の第1候補の表示を指示具3でタップすると、その文字の第1候補から第5候補までの候補群を候補表示エリア 30
に表示させる。そして、この候補群から、その任意の候補をタップすると、そのタップした候補の文字と表示されている第1候補と入れ替える処理を行う。

【0057】

尚、この機能構成は、携帯情報機器1上で実現される場合を例に挙げて説明しているが、これに限定されるものではない。つまり、この携帯情報機器と同等の機能あるいは、それ以上の機能を有するパーソナルコンピュータ等の汎用端末（情報処理装置）においても、本発明の図3の機能構成を実現できることは言うまでもない。

【0058】

本発明では、手書き文字認識環境において実現される各種処理において、その処理種類に応じて、特に、文字列単位の後処理における後処理の継続の可否を、後処理継続判定部 40
33において判定する。そして、この判定を、後処理継続判定部33で実現するために、例えば、後処理継続可否判定テーブルをRAM7上に記憶する。

【0059】

ここで、この後処理継続可否判定テーブルについて、図4を用いて説明する。

【0060】

図4は本発明の実施形態1の後処理継続可否判定テーブルを示す図である。

【0061】

この後処理継続可否判定テーブルは、処理種類と、その後の後処理を継続するか否か（既に得られている文字認識結果の候補文字列に継続する新たな文字を含む文字列全体を、後処理対象の文字列とするか否か）を示す情報とを対応づけて管理している。 50

【0062】

例えば、処理種類判別部31で判定された処理種類が手書き文字認識候補の候補変更処理である場合は、後処理継続判定部33は、後処理継続可否判定テーブルを参照することで、その時点で、その候補選択指示以前に入力していた文字に対する後処理はその時点で一旦中止（停止）すると判定する。

【0063】

また、別の例としては、文字切り時間（ペンアップ時間（指示具3による操作がなされていない時間：未入力時間）が一定時間経過した場合に発生する処理（タイマーイベント））等による1文字の文字データの切り分け操作が行われた場合は、後処理継続判定部33は、後処理継続可否判定テーブルを参照することで、その操作が行われた時点で入力していた文字及びそれ以降に入力される文字に対する後処理を継続すると判定する。

10

【0064】

このように、本発明では、指示具3等による入力操作やイベント発生毎に、その入力操作種類やイベント種類に基づいて、後処理の継続／中止を決定することで、ユーザにより使いやすい操作環境を提供することができる。

【0065】

尚、この後処理継続可否判定テーブルは一例であり、用途や目的に応じて、様々な構成を実現することができる。つまり、ユーザの意図や目的に応じて、認識処理によって得られる文字列に対して後処理を適用したい処理種類やイベントに応じて様々な設定内容の後処理継続可否判定テーブルを構成できることは言うまでもない。

20

【0066】

次に、実施形態1の文字認識処理の具体例について、図5を用いて説明する。

【0067】

図5は本発明の実施形態1の文字認識処理の具体例を説明するための図である。

【0068】

図5では、情報携帯機器1で動作する電子メールアプリケーション上で、本発明の文字認識処理を実現する手書き文字入力ユーティリティプログラムが動作する場合の表示画面例を示している。

【0069】

尚、実施形態1では、例として、電子メールアプリケーションが生成するアプリケーション画面上で手書き文字入力ユーティリティプログラムを動作する例を挙げて説明するが、これに限定されるものではない。つまり、本発明は、文字入力が必要な他のアプリケーションプログラム、例えば、ファイルブラウザ、フォトビューワ、スケジューラ、ワープロプログラム等の他のプログラムについても有効であることは言うまでもない。

30

【0070】

図5において、領域51は、電子メールの宛先やサブジェクトを入力するための各種入力フィールドから構成される領域である。例えば、指示具3で、領域51内の任意の位置を押下することにより、対応する入力フィールドに、テキスト入力用のカーソルが表示される。ユーザは、この状態で、手書き文字入力ユーティリティプログラムを使用して、その入力フィールド内の文字入力を実行することができる。

40

【0071】

領域52は、電子メールの本文を入力するための領域である。例えば、指示具3で、その領域52の任意の位置を押下することにより、テキスト入力用のカーソルが領域52に表示される。ユーザは、この状態で、手書き文字入力ユーティリティプログラムを使用して、その領域内の文字入力を実行することができる。

【0072】

画面53は、手書き文字入力ユーティリティプログラムによって実現される手書き文字入力用画面である。その詳細については、図6で説明するが、任意のアプリケーションプログラム上で、文字入力が必要な場合は、この手書き文字入力ユーティリティプログラムを起動することで、ユーザは、手書き入力による文字入力を実行することができる。

50

【0073】

もちろん、手書き文字入力ユーティリティプログラムを必要とするアプリケーションプログラムに予め割り付けておいて、そのアプリケーションプログラムの起動と同時に、手書き文字入力ユーティリティプログラムが起動される構成とすることも可能である。

【0074】

この手書き文字入力ユーティリティプログラムによる手書き文字入力用画面53に対して、ユーザによって手書き文字入力（軌跡データの入力）がなされると、図3の手書き文字認識部35によってその軌跡データが認識される。そして、認識結果として得られる文字コードに対応する文字フォントが、一旦、手書き文字入力用画面53で表示される。

【0075】

ユーザは、手書き文字入力用画面53が表示された文字が所望の文字であれば、その文字を確定することにより、現在表示されている文字が、電子メールアプリケーションプログラム上で表示されているカーソル位置に挿入される。

【0076】

領域54は、各種アプリケーションプログラムを切り替えるためのコントロール群と、手書き文字入力ユーティリティプログラムの起動を指示するためのコントロール（ボタンやアイコン）が配置されている領域である。

【0077】

領域54内において、ボタン55は、手書き文字入力ユーティリティプログラムの起動ボタンである。このボタン55が、指示具3で押下されると、手書き文字入力ユーティリティプログラムが起動され、その手書き文字入力用画面53が表示される。一方、手書き文字入力ユーティリティプログラムが起動されている状態で、ボタン55が押下された場合は、手書き文字入力ユーティリティプログラムが終了され、手書き文字入力用画面53が消去される。

【0078】

ボタン56は、ファイルブラウザの起動ボタンである。このボタン56が、指示具3で押下されると、携帯情報機器1で管理されている各種データを閲覧するためのファイルブラウザアプリケーションが起動される。

【0079】

ボタン57は、電子メールアプリケーションの起動ボタンである。このボタン56が、指示具3で押下されると、電子メールアプリケーションが起動される。

【0080】

ボタン58は、フォトビューワの起動ボタンである。このボタン56が、指示具3で押下されると、携帯情報機器1で管理されている画像データを閲覧するためのフォトビューワ（フォト表示アプリケーション）が起動される。

【0081】

ボタン59は、スケジューラ起動ボタンである。このボタン59が、指示具3で押下されると、スケジューラアプリケーションが起動される。

【0082】

次に、手書き文字入力ユーティリティプログラムが生成する手書き文字入力用画面53の詳細について、図6を用いて説明する。

【0083】

図6は本発明の実施形態1の手書き文字入力ユーティリティプログラムが生成する手書き文字入力用画面の一例を示す図である。

【0084】

図6において、領域61は入力文字バッファ領域であり、手書き文字認識結果の第1候補を表示する。ここでは、手書き文字の1文字の入力が確定し、認識処理が終了すると、その認識結果の第1候補が順次表示される。例えば、手書き入力で「今日は」と入力すると、その軌跡データが認識用辞書に登録されている形状（標準文字辞書データ）と同じ文字であれば、その認識結果として「今日は」が、この領域61に表示される。

【0085】

ここで、入力文字バッファ領域61に表示されている文字群の内、意図する候補でない文字が含まれている場合には、ユーザは、その意図する候補でない文字を指示具3で押下して選択することで、その候補を変更することができる。

【0086】

具体的には、ユーザが、入力文字バッファ領域61中の任意の文字を押下して選択すると、候補表示領域66に、その文字に対する第1位から第4位の候補群がその候補順に表示される。そして、その候補群に、意図する候補がある場合には、ユーザは、その候補を指示具3で押下して選択することにより、入力文字バッファ領域61で選択した文字を、候補表示領域66で選択した文字に、第1候補の文字として入れ替えることができる。

10

【0087】

尚、入力文字バッファ領域61に表示される第1候補の文字群は、後処理部34の機能がユーザ指定によって明示的に無効にされている時は、手書き文字認識部35の認識結果の認識候補順位をそのまま反映している。一方、後処理部34の機能がユーザ指定によって明示的に有効にされている時は、手書き文字認識部35の認識結果の認識候補を元に、その文字列の連鎖確率と、認識結果との一致度に基づいて、認識候補の順位を入れ替えて、入力文字バッファ領域61内の文字の表示を制御する。

【0088】

ボタン62は変換ボタンである。このボタン62が、指示具3で押下されると、入力文字バッファ領域61内の平仮名文字の仮名漢字変換を行う。例えば、入力文字バッファ領域61に、「あいが」が入力されている状態で、このボタン62が押下されると、例えば、「愛が」に変換される。この変換候補が意図するものでない場合は、再度、ボタン62を押下することにより、次候補が表示される。

20

【0089】

ボタン63はBack Spaceキーボタン（削除ボタン）である。このボタン63が、指示具3で押下されると、入力文字バッファ領域61内に文字が表示されている場合は、例えば、その入力文字バッファ領域61内で表示されるカーソルが位置する前の1文字が削除される。一方、入力文字バッファ領域61内に文字が1文字も入力されていない場合は、アプリケーション（例えば、電子メールアプリケーション上の領域52）上で表示されているカーソルの左側の1文字が削除される。

30

【0090】

ボタン64はSpaceキーボタン（空白文字挿入ボタン）である。このボタン64が、指示具3で押下されると、入力文字バッファ領域61内に文字列が表示されていて、文字列の後ろに挿入位置がある時は、その文字列の後ろに空白文字（空白コード）が挿入される。

【0091】

ボタン65はEnterキーボタン（候補文字列確定／改行入力ボタン）である。このボタン65が指示具3で押下されると、入力文字バッファ領域61内に文字列が表示されている場合は、その文字列を確定して、アプリケーション上で表示されているカーソルの右側に、その文字列が挿入される。一方、入力文字バッファ領域61内に文字が存在しない場合は、アプリケーション上で表示されているカーソルの右側に、改行コードが挿入され、カーソルが改行される。

40

【0092】

領域66は手書き文字認識の候補表示領域である。ここでは、上述のように、入力文字バッファ領域61内の文字列中の選択された文字に対応する認識候補の第1位から第4位までを表示する。

【0093】

上述したように、候補表示領域66内に表示された文字の中から任意の文字を指示具3で押下して選択すると、入力文字バッファ領域61内で選択した文字との入替が行われる。

50

【0094】

例えば、入力文字バッファ領域61内に「今日は良り」と言う認識結果の文字列が表示されている場合に、「り」を指示具3で押下して選択すると、候補表示領域66内には、「り」に対する第1候補から第4候補である文字として、その順で「り」、「い」、「い」、「ん」等が表示される。ユーザは、その中から任意の文字を選択できる。例えば、「い」を指示具3で押下して選択すると、入力文字バッファ領域61内の「り」が「い」に入れ替えられ、全体の文字列は、「今日は良り」から「今日は良い」に変更されることになる。

【0095】

領域67は手書き文字入力領域である。ここでは、2つの領域67a及び67bに対して、手書き文字を交互に記入して行くことによって文字入力を行うことができる。 10

【0096】

例えば、領域67aに、手書きで「い」を記入し、次に、領域67bに手書きで「天」を構成する1筆（ストローク）目を書き出すと、領域67aの「い」の入力を確定し、「い」に対する認識処理を開始する。ここで、「い」の認識が終了した時点で、入力文字バッファ領域61内に「い」の認識結果の第1候補が表示され、その時点で、領域67aに入力された手書き文字のストロークは消去される。

【0097】

次に、領域67bに対する処理を実行する。これは、「天」を構成する手書きのストロークが入力し終った時点で、次に書く文字が存在する場合は、領域67aに次の文字を書き出すと、「天」の認識処理が開始される。 20

【0098】

一方、次に入力する文字がない場合は、指示具3からの操作（例えば、押下）時間が一定時間（ペンアップ時間）経過した時点で、「天」に対する入力が終了したと確定して、認識処理を行う。または、手書き文字入力領域67以外の部分を、指示具3で触ると、その時点で、「天」に対する入力が終了したと確定しても良い。

【0099】

そして、「天」の認識処理が終了した時点で、入力文字バッファ領域61の「い」の認識結果の文字の右側に、「天」の認識結果の第1候補が表示され、領域67bの「天」を構成するの手書きのストロークが消去される。 30

【0100】

尚、これらの手書き文字の入力処理自体は、従来の手書き文字入力処理と同様にして実現することもできる。

【0101】

また、図5及び図6で示される、各種処理の実行を指示するための各種操作（指示）コントロール（ボタン、メニュー等）は、GUI上で実現する例を示しているが、例えば、携帯情報機器上の物理的なボタンやキーによって実現されても良い。

【0102】

次に、実施形態1で実行する処理について、図7～図9を用いて説明する。

【0103】

図7は本発明の実施形態1の文字列後処理継続判定処理を示すフローチャートである。 40

【0104】

尚、この処理は、主に、後処理継続判定部33によって実行される。この処理の実行にあたっては、この処理が使用するRAM7上のワーク領域等の初期化を行う。

【0105】

まず、座標入力部22によって、ユーザが、指示具3を使用して入力した文字や、編集操作が位置座標データとして入力される。そして、その位置座標データが示す位置によって、手書き文字の入力であるか、編集操作であるか、候補選択操作であるか、その他の操作であるか等の各種操作内容を判別する。尚、この判別は、上述したように、図3の処理種類判別部31によって実現される。 50

【0106】

実施形態1では、この処理種類判別部31による入力種類の判別例として、図6に示した手書き文字入力ユーティリティプログラムで実現する場合を例に挙げて説明する。尚、この手書き文字入力ユーティリティプログラムは、上述したように、図5の手書き文字入力用画面内のボタン55を押下することにより起動される。

【0107】

以下、フローチャートについて、具体的に説明する。

【0108】

ステップS702において、入力文字バッファ領域61内の文字列候補の変更を実行する候補変更処理である否かを判定する。この判定は、例えば、図6の入力文字バッファ領域61内の文字列上に対して指示具3が押下されたか否かを判定する。特に、その領域内の位置座標が入力された場合には、候補変更処理であると判定する。そして、候補変更処理である場合（ステップS702でYES）、ステップS703に進む。一方、候補変更処理でない場合（ステップS702でNO）ステップS704に進む。

【0109】

ステップS703において、文字列後処理完了フラグをセットする。この文字列後処理完了フラグは、文字列後処理完了の有無を示すフラグである。

【0110】

例えば、ユーザが候補変更処理を行う場合は、その時点で、入力文字バッファ領域61内に表示されている第1候補の文字列が、意図した文字列と異なる時である。そして、候補が変更された場合、変更された後の候補表示文字列が、ユーザが所望する文字列となる。従って、この時点で、今まで入力された文字列の後処理を終了（中止）させるため、文字列後処理完了フラグを、例えば、RAM7にセットする。そして、この文字列後処理完了フラグがセットされると、図3の後処理部34に後処理完了を指示する。

【0111】

ステップS704において、手書き文字入力の有無を判定する。この判定は、図6の手書き文字入力領域67内の位置座標の入力の有無に基づいて判定する。手書き文字入力がある場合（ステップS704でYES）、ステップS705に進む。一方、手書き文字入力がない場合（ステップS704でNO）、ステップS706へ進む。

【0112】

ステップS705において、後処理文字列に認識候補を追加する処理を行う。図6の手書き文字入力領域67に、入力された手書き文字の軌跡の1文字が確定した後で、図3の手書き文字認識部35に入力された手書き文字の軌跡の位置座標データを渡す。次に、従来からあるオンライン手書き文字認識アルゴリズムを用いて、その位置座標データから、文字の特徴データを抽出する。

【0113】

次に、その特徴データと、認識用辞書内の標準文字辞書データ（標準文字特徴データ）とのマッチング処理により、特徴が一致している度合いが高い標準文字辞書データの順で、入力された手書き文字に他する第1候補として第4候補までの候補を決定する。そして、決定した第1候補から第4候補それぞれに対応する文字コードと、一致度の評価値を、後処理部34に渡す。

【0114】

後処理部34は、この認識候補の文字コードと評価値からなる文字候補列から、その文字列同士の連鎖確率の高い組み合わせを決定して、その高い組み合わせを第1候補として、入力文字バッファ領域61内の表示を変更する。

【0115】

例えば、入力文字バッファ領域61内の表示が最初は、手書き文字認識部35によって、「12ろ4」であった場合で説明する。

【0116】

ここで、「ろ」に注目すると、「ろ」に対し、その第1候補が「ろ」、第2候補が「3

」、第3候補が「了」。第4候補が「る」であったとする。この場合、文字列の組み合わせとしては、「1 2 ろ 4」、「1 2 3 4」、「1 2 了 4」、「1 2 る 4」の組み合わせがある。

【0 1 1 7】

この場合、連鎖確率では、「1 2 3 4」の組み合わせが高い評価値となり、第1候補の「ろ」と第2候補の「3」の評価値間の差が少なければ、「1 2 3 4」の組み合わせが選ばれる。その結果、入力文字バッファ領域61内の表示が、「1 2 ろ 4」から「1 2 3 4」に切り替わる。

【0 1 1 8】

次に、ステップS706において、文字切り時間タイムアップの有無を判定する。

10

【0 1 1 9】

図6の手書き文字入力領域67で、1つの文字を構成する軌跡の内、最後の軌跡が入力された後で、指示具3での入力がある設定時間（所定時間）以上経過した場合、文字切り時間タイムアップイベントを発生させる。尚、この構成は、従来からのオンライン手書き文字認識で採用される一般的な処理でもある。

【0 1 2 0】

そこで、文字切り時間が経過すると、この文字切り時間タイムアップイベントが発生する、つまり、文字切り時間タイムアップである場合（ステップS706でYES）、ステップS707に進む。

【0 1 2 1】

一方、それ以外のイベントや、それ以外の位置座標入力である場合（ステップS706でNO）、ステップS708に進む。

20

【0 1 2 2】

ステップS707において、後処理文字列に認識候補を追加した後処理部34に対しては、継続指示を行う処理を行う。尚、従来の後処理を行う文字認識処理では、連続入力が終了したと判定される時点（例えば、文字切り時間タイムアップ時点）で、後処理を終了している。

【0 1 2 3】

これに対し、実施形態1では、文字切り時間タイムアップイベントによって、ユーザが意図する手書き文字の軌跡群の最後の軌跡の入力が確定した場合に、一旦、手書き文字入力領域67内の軌跡データを手書き文字認識部35に渡す。そして、この手書き文字認識部35から出力される候補文字とその評価値を後処理部34に渡し、後処理部34に後処理継続指示を出して入力待ちとする。

30

【0 1 2 4】

これにより、本発明では、文字切り時間タイムアップが発生しても、それ以降の入力に対しても、後処理を継続することができる。

【0 1 2 5】

ステップS708において、認識開始領域が押下されたか否かを判定する。実施形態1では、認識開始領域は、図6の手書き文字入力領域67外の領域を定義しており、この領域に対する位置座標が入力された場合には、まず、認識開始イベントが発生する。

40

【0 1 2 6】

具体的には、図6の手書き文字入力領域67外の座標が入力された時に、手書き文字入力領域67内に未処理の軌跡データが存在する場合は、まず、認識開始イベントが発生する。この認識開始イベントが発生した場合（ステップS708でYES）、ステップS709に進み、その位置座標入力のデータはそのまま保持され、その位置における処理を、次のイベントとして実行する。

【0 1 2 7】

例えば、手書き文字入力領域67内に、「気」の手書き文字の軌跡が入力されている状態で、ボタン65が指示具3で押下されると、まず、手書き文字入力領域67外の領域（認識開始領域）が押されたことによる認識開始イベントが発生して、手書き文字認識が行

50

われる。次に、その認識結果が入力文字バッファ領域 6 1 に表示された後で、ボタン 6 5 の押下に対する処理が実行される。一方、それ以外のイベントの場合（ステップ S 7 0 8 で NO）、ステップ S 7 1 0 に進む。

【0 1 2 8】

ステップ S 7 1 0 において、1 文字消去処理が指示されたか否かを判定する。これは、ボタン 6 3 上の位置座標が入力された場合（ステップ S 7 1 0 で YES）、1 文字消去処理と判定して、ステップ S 7 1 1 に進む。一方、それ以外の位置座標入力である場合（ステップ S 7 1 0 で NO）、ステップ S 7 1 2 に進む。

【0 1 2 9】

ステップ S 7 1 1 において、1 文字削除処理を実行する。これは、入力文字バッファ領域 6 1 内に文字がある場合は、その入力文字バッファ領域 6 1 内の右端の 1 文字を消去する。次に、その消去した文字に対応する、後処理部 3 4 内の後処理文字列内の認識候補を削除する処理を実行する。一方、入力文字バッファ領域 6 1 内に文字がない場合は、アプリケーション上のカーソル位置左側の文字を消去し、後処理部 3 4 内には処理対象のデータが存在しないため、何もしないでこの処理を終了する。

【0 1 3 0】

ステップ S 7 1 2 において、アプリケーションの切替処理の有無を判定する。アプリケーション切替指示となる、図 5 のボタン 5 6 や 5 9 等上の位置座標が入力された場合（ステップ S 7 1 2 で YES）、アプリケーション切替動作であると判定して、ステップ S 7 1 3 に進む。一方、それ以外の位置座標入力である場合（ステップ S 7 1 2 で NO）、ステップ S 7 1 4 に進む。

【0 1 3 1】

ステップ S 7 1 3 において、後処理文字列を継続する。実施形態 1 では、アプリケーションの切替を行っても、手書き文字入力ユーティリティプログラムの入力が継続する構成となっている。これは、入力途中でアプリケーションを変更する要求が発生する場合もあることを想定したものである。そのため、後処理部 3 4 に対して後処理継続指示を出して入力待ちとする。

【0 1 3 2】

ステップ S 7 1 4 において、仮名漢字変換処理の指示の有無を判定する。

【0 1 3 3】

これは、図 6 のボタン 6 2 の位置座標が入力された場合（ステップ S 7 1 4 で YES）、仮名漢字変換処理であると判定して、ステップ S 7 1 5 へ進む。それ以外の位置座標入力である場合（ステップ S 7 1 4 で NO）、ステップ S 7 1 6 へ進む。

【0 1 3 4】

ステップ S 7 1 5 において、仮名漢字変換処理を実行する。仮名漢字変換処理自体は、従来から行われている各種仮名漢字変換処理で構わないので、その詳細な説明は省略する。簡単に説明すると、入力文字バッファ領域 6 1 内に文字が入力されている場合で、例えば、「てんき」の平仮名が入力されている場合は、その文字列の仮名漢字変換を行い、「天気」等に変換する。この仮名漢字変換処理が実行される場合、入力文字列から、異なる文字コードの文字列に変換される処理がなされる。そのため、ここでは、後処理を継続しても意味をなさないので、後処理部 3 4 に対する文字列後処理完了フラグをセットする。

【0 1 3 5】

ステップ S 7 1 6 において、文字列挿入が指示されたか否かを判定する。文字列挿入のコマンド等がユーザ指示により発生した場合（ステップ S 7 1 6 4 で YES）、ステップ S 7 1 7 に進む。このステップ S 7 1 7 は、ステップ S 7 1 5 と同様の処理を実行する。一方、それ以外の位置座標入力やそれ以外のイベントである場合（ステップ S 7 1 6 で NO）、処理を終了する。

【0 1 3 6】

ここまで説明した処理により、各手書き文字入力の入力処理や編集処理に応じて、手書き文字の文字列単位の処理である後処理の継続や中止を制御することができる。

【0137】

尚、図7で示される各種判定（ステップS702、ステップS704、ステップS706、ステップS708、ステップS710、ステップS712、ステップS714、ステップS716）の順序は、図7で示される順序に限定されるものではなく、用途や目的に応じて、その順序を変更することができる。

【0138】

次に、手書き文字認識処理について、図8を用いて説明する。

【0139】

図8は本発明の実施形態1の手書き文字認識処理を示すフローチャートである。

【0140】

尚、手書き文字認識処理を開始する場合には、まず、この処理が使用するRAM7上のワーク領域等の初期化を行う。また、この処理は、主に、手書き文字認識部35によって実行される。認識対象とする1文字分の軌跡の位置座標データは、図6の手書き文字入力領域67上の領域67a及び67bによって、それぞれの領域に入力された軌跡データが1文字分の軌跡データとして入力される。

【0141】

ステップS902において、1文字分の軌跡データをワーク領域に読み込む。例えば、1文字分の軌跡データは、1画の文字であれば、1筆の軌跡データで構成されている。また、2画以上の文字であれば、複数の1筆の軌跡データから構成されている。1筆の軌跡データは、その位置を規定する、例えば、始点のXY位置座標データからはじまり、軌跡の形状に沿ったXY位置座標データが終点のXY位置座標まで続く。そして、従来からあるアルゴリズムにより、位置座標データの正規化を行い、特徴点（特徴データ）を抽出する。

【0142】

ステップS903において、得られる軌跡データに対して、認識処理を実行し、その認識結果である候補文字（文字コード）をRAM7に記憶する。ここでは、ステップS902において抽出した軌跡データの特徴データと、認識用辞書の標準文字辞書データとのマッチング処理を行う。そして、一番一致度が高い標準文字辞書データに対応する文字コードと一致度のスコア（評価値）を認識結果として記憶し、順次一致度の高い順に標準文字辞書データに対応する文字コードを認識結果候補として記憶する。

【0143】

ステップS904において、認識結果候補を後処理部34に登録する。つまり、ステップS903において記憶した認識結果の文字コードとその評価値を、後処理部34に登録する。後処理部34は、順次認識された認識結果である文字コードとその評価値を記憶して行くことになる。

【0144】

ここで、例えば、「工場」という手書き文字が入力された場合について説明する。まず、手書き文字認識部35によって、「工」に対する認識結果の第1候補の文字が「工」でその評価値が100であり、第2候補の文字が「エ」でその評価値が100であり、第3候補の文字が「ェ」でその評価値が90であるとする。

【0145】

次に、「場」に対する認識結果の第1候補の文字が「揚」でその評価値が98で、第2候補の文字が「場」で一致度の評価値が96であるとする。

【0146】

この場合、後処理部34では、その候補の組み合わせ文字列の連鎖確率を後処理用辞書によって算出する。組み合わせは、「工場」、「工場」、「エ揚」、「エ場」、「ェ揚」、「ェ場」等になる。そして、後処理部34は、これらの組み合わせの中で、一番連鎖確率が高い候補文字列を決定する。

【0147】

ステップS905において、後処理部34から候補文字列を読み出す。

【0148】

上述のように、例えば、手書き文字として「工場」を入力した際の手書き文字認識処理部35での認識結果が「工場」である場合でも、ステップS904の後処理部34の処理によって、一番連鎖確率が高い候補文字列「工場」として決定されると、手書き文字の候補文字列として、「工場」が読み出されることになる。この場合、手書き文字認識処理部35での認識結果が「工場」が「工場」に置き換えることができる。

【0149】

ステップS906において、認識結果を表示する。例えば、図6の入力文字バッファ領域61において、当初の認識結果として得られた「工場」が、ステップS905の処理によって「工場」に置き換わって表示されることになる。

10

【0150】

この一連の処理を、時系列で説明すると、最初に「工」が入力され、次の文字「場」の1筆目が入力されると、その時点で、入力された「工」に対する手書き文字認識部35による認識結果の「工」が、入力文字バッファ領域61に表示される。

【0151】

次に、「場」の文字の入力が、例えば、文字切り時間で確定すると、「場」に対する手書き文字認識部35による認識が開始され、その認識結果の1位が「場」であれば、入力文字バッファ領域61には、その認識結果として「工場」が表示されることになる。

【0152】

次に、後処理部34によって、手書き文字認識部35による認識候補内の組み合わせで一番連鎖確率が高い候補文字列を決定し、その候補文字列が「工場」であれば、入力文字バッファ領域61の表示を「工場」から「工場」に置き換える。

20

【0153】

更に、次に入力されるユーザ指示が手書き文字、例えば、「が」であれば、入力文字列としては、「工場が」後処理対象となって後処理が実行される。

【0154】

一方、後処理結果の「工場」が意図した認識結果でなく、他の候補への候補変更処理や、1文字消去処理等の指示がなされる場合は、その時点までの文字列を確定文字列として取り扱い、後処理の継続を中止する（つまり、一旦、この時点で、文字列後処理完了フラグがセットされる）。その後、次に、入力される手書き文字は、後処理対象の先頭の文字として取り扱うことになる（この場合、文字列後処理完了フラグがリセットされる）。

30

【0155】

その後、手書き文字認識処理を終了する。ここで、記憶した一文字分の軌跡データを消去し、ワーク領域等を開放する。

【0156】

以上説明したように、実施形態1によれば、ユーザの指示入力によって規定される処理種類に応じて、入力された文字列に対する後処理の継続の有無を判定する後処理継続判定部を設ける。これによって、ユーザが、入力した文字列に対する後処理の継続や、確定を意識することなく（つまり、後処理単位の区切りを指定する操作を実行することなく）、文字入力や編集操作を自然に行える手書き文字認識環境を実現することができる。これにより、ユーザが意図する文字の入力が容易な手書き文字認識環境を実現することができる。

40

【0157】

<実施形態2>

実施形態1では、後処理対象の文字列は、手書き文字の認識結果を取り扱う例で説明を行っているが、通常の携帯情報機器では、手書き文字入力以外にも他のアプリケーションからコピーしたテキスト文字列を挿入して使う場合も多く存在する。このような場合は、1つの文の中に手書き文字の認識結果の文字と、挿入されたテキスト文字列とが混在することになる。この場合、今までの手書き文字認識装置では、このような状況での後処理を実現することはできない。

50

【0158】

そこで、実施形態2では、実施形態1の構成に加えて、テキスト挿入文字列を後処理部34へ挿入するための挿入部101（図9）を追加する。これにより、1つの文の中に手書き文字の認識結果の文字と、挿入されたテキスト文字列とが混在する状態での、後処理を効率良く実現することができる。

【0159】

尚、挿入されたテキスト文字列とは、例えば、任意のアプリケーション上でコピーされた文字列や、過去に入力された文字列や、手書き認識された平仮名を仮名漢字変換した文字列がある。

【0160】

まず、実施形態2の携帯情報機器の機能構成について、図9を用いて説明する。

【0161】

図9は本発明の実施形態2の携帯情報機器の機能構成を示す図である。

【0162】

尚、実施形態1の図3と同一の構成については、同一の参照番号を付加して、その詳細説明については省略する。

【0163】

実施形態2では、上述したように、テキスト挿入文字列を後処理部34へ挿入するための挿入部101を有している。この挿入部101は、図6の手書き文字入力ユーティリティプログラム内の入力文字バッファ領域61に、他のアプリケーションプログラムや過去に保存したテキスト文字列を挿入する時に呼ばれる処理部である。

【0164】

この挿入部101では、入力文字バッファ領域61に入力されている手書き文字の認識結果である候補文字列と、挿入されたテキスト文字列を後処理の対象文字列として取り扱うための処理を実行する。そして、この処理された候補文字列を、後処理部34に送信する。

【0165】

次に、実施形態2の文字認識処理の具体例について、図10～図12を用いて説明する。

【0166】

図10～図12は本発明の実施形態2の文字認識処理の具体例を説明するための図である。

【0167】

図10～図12は、ワープロアプリケーション上で、手書き文字入力ユーティリティプログラムが動作する場合の表示画面例を示している。

【0168】

図10において、111は、ワープロアプリケーションのテキスト入力領域を示している。112は、後処理部34に挿入されるテキスト文字列を概念的に表している表示である。「都市」というテキストが、そのワープロアプリケーションが動作するOS（オペレーティングシステム）の機能であるテキスト用のカットバッファ領域（クリップボード領域：OSが内部メモリに確保する記憶領域）に記憶されている状態を示している。

【0169】

図11は、図10の状態から、「都市」のテキスト文字列が入力文字列バッファ領域61に挿入された状態を示している。また、図12は、図11の入力文字列バッファ領域61中の「失われた都市の記憶を」の文字列の後ろに、手書きの「思い」を入力しようとする状態を示している。

【0170】

次に、挿入部101による、挿入された文字列を後処理部34へ挿入する（追加する）挿入処理について、図13を用いて説明する。

【0171】

図 1 3 は本発明の実施形態 2 の挿入処理を示すフローチャートである。

【0172】

尚、このフローチャートに沿って、図 1 0 ～図 1 2 の状態遷移についても説明する。

【0173】

他のアプリケーション等で、メニューや、操作キー等による従来からある操作によって文字列を選択し、特定記憶領域（システム文字列記憶領域）にコピーする。

【0174】

次に、その特定記憶領域（システム文字列記憶領域）に記憶された文字列を、テキストカーソルが表示している場所にコマンドやメニュー等の指示により挿入する。

【0175】

ここまでの処理は、一般的なコンピュータや携帯情報機器での一般的な処理なので詳細な説明は行わない。

【0176】

図 1 0 の入力文字列バッファ領域 6 1 の「失われたの記憶を」の文字列の「失われた」の後ろにカーソルを移動させ、そこで、システム文字列記憶領域に記憶している「都市」の挿入が指示される。

【0177】

挿入が指示されると、画面上は、図 1 1 の入力文字列バッファ領域 6 1 の「失われた都市の記憶を」の文字列になる。尚、テキストの挿入処理自体は、従来の携帯情報機器と同一の処理である。

【0178】

この処理は、主に、挿入部 1 0 1 によって実行される。この処理の実行にあたっては、この処理が使用する R A M 7 上のワーク領域等の初期化を行う。

【0179】

次に、ステップ S 1 4 2 において、システム文字列記憶領域に、記憶された挿入対象のテキスト文字列（挿入テキスト文字列）を読み込む。例えば、図 1 0 で表している、挿入テキスト文字列「都市」の文字コードを R A M 7 内に記憶する。また、挿入が指示された時のカーソルの位置の位置情報を記憶する。

【0180】

ステップ S 1 4 3 において、その挿入テキスト文字列を確定文字列として後処理部 3 4 に挿入するために、その文字列データを、後処理部 3 4 で処理する文字データと同じ構造（フォーマット）に加工する加工処理を実行する。後処理部 3 4 で処理する文字列データ（後処理用文字列データ）は、文字コードとその評価値で構成されている。そこで、この挿入テキスト文字列は、もともと文字コードで規定されているので、加工処理では、その文字コードに対して、その評価値を最高点である 1 0 0 に設定した後処理用文字列データを作成する。

【0181】

例えば、「都市」の例で説明すると、（「都」の文字コードとその評価値 1 0 0、「市」の文字コードとその評価値 1 0 0）の後処理用文字列データを作成し、カーソルの位置のデータと一緒に、後処理部 3 4 に出力する。

【0182】

その後、この処理が使用したワーク領域等を開放する処理を実行して、処理を終了する。

【0183】

後処理部 3 4 は、挿入カーソル位置を示す位置情報に基づいて、挿入テキスト文字列に対応する後処理用文字列データの挿入位置を判定する。そして、その判定結果に基づいて、その後処理用文字列データを、後処理部 3 4 の後処理文字列記憶領域内のカーソル位置に対応するデータ位置に挿入する。

【0184】

例えば、図 1 0 及び図 1 1 の例で説明すると、文字列「失われたの記憶を」において、

「た」の右の位置にカーソルが存在する。そのため、カーソル位置は、4文字目の右となり、挿入カーソル位置のデータは4文字目を示す4というデータになる。

【0185】

図10の文字列「失われた記憶を」が、手書き文字入力における認識結果であれば、後処理の1文字目の「失」のデータは、第1候補「失」とその評価値98、第2候補「矢」とその評価値90で構成されている。以降、同様にして、例えば、4文字目の「た」は、第1候補「た」とその評価値97、第2候補「E」とその評価値60、第3候補「太」とその評価値59である。また、5文字目の「の」は、第1候補「の」とその評価値97、第2候補「0」とその評価値80、第3候補「o」とその評価値78である。

【0186】

この状態の後処理用文字列データに対して、「都」の文字コードとその評価値100、「市」の文字コードとその評価値100を挿入する。ここで、挿入位置は、4文字目のデータの後ろであるから、4文字目の「た」の最後のデータである第3候補「太」の後ろに挿入され、5文字目が「都」となり、そのデータは第1候補「都」とその評価値100となる。

【0187】

続いて、6文字目が「市」となり、そのデータは第1候補「市」とその評価値100となる。7文字目は「の」となり、そのデータは第1候補「の」とその評価値97、第2候補「0」とその評価値80、第3候補「o」とその評価値78になる。以下のデータは、順次後ろにずれることになる。

【0188】

以上のようにして、後処理部34で処理する後処理用文字列データの内容が更新される。つまり、ここでは、1つの文の中に手書き文字の認識結果の文字と、挿入テキスト文字列とが混在する状態の後処理用文字列データが生成されることになる。

【0189】

次に、その後処理文字列データが更新された後に、手書き文字が入力される例を、図12を用いて説明する。

【0190】

図12において、手書き文字入力67の領域67a及び67bとにおいて、手書き文字「思い」が入力されると、実施形態1で説明した処理と同様に、まず、手書き文字認識処理が実行される。この場合、例えば、「思」に対する認識結果の第1候補「思」とその評価値98、第2候補「恵」とその評価値67と、「い」の認識結果の第1候補「い」とその評価値95、第2候補「い」とその評価値90が得られるとする。

【0191】

この手書きの「思い」が入力される時に、後処理部34における後処理用文字列データは「失われた都市の記憶を」であり、また、手書き文字の認識結果を挿入するカーソル位置は「を」の後ろに存在する。

【0192】

この場合、後処理用文字列データの10文字目の「を」の第1候補「を」（評価値96）、第2候補「あ」（評価値77）の後ろに、「思」に対する認識結果の第1候補「思」（評価値98）、第2候補「恵」（評価値67）と、「い」の認識結果の第1候補「い」（評価値95）と、第2候補「い」（評価値90）のデータを挿入する。

【0193】

このように処理された、後処理用文字列データ（「失われた都市の記憶を思い」）を元に、実施形態1で説明した後処理を行い、認識候補の表示候補を変更し、入力文字バッファ領域61に表示する。

【0194】

この後処理用文字列データでの後処理を簡単に説明すると、後処理用文字列データ「失われた都市の記憶を思い」の内、「失われた」、「記憶を」、「思い」は手書き入力に対する認識結果であるので、その処理対象としてはそれぞれ複数の候補からなる文字コード

群が存在し、「都市」は挿入テキスト文字列データとして入力されたので、1つの文字に対して1文字コードしか存在しない。

【0195】

即ち、後処理用文字列データは、図14に示すような文字列（文字コード）候補列の組み合わせが得られることになる。そして、後処理部34は、一番連鎖確率が高い候補文字列を連鎖確率用の後処理用辞書を元に決定する。尚、連鎖確率の算出方法については、従来より周知の方法を用いれば良いので、ここでは、その説明は省略する。

【0196】

以上説明したように、実施形態2によれば、テキスト文字を手書き文字入力に対する文字認識結果編集（確認）領域（入力文字バッファ領域）に挿入する構成を設けると、手書き文字列の入力過程で、別のアプリケーションで生成されたテキスト文字列やシステム文字列記憶領域に記憶されたテキスト文字列を挿入することができる。

10

【0197】

これにより、手書き文字入力に対する認識結果からなる文字列と、別途挿入されたテキスト文字列とが混在する文字列に対する後処理を問題無く機能させることができる手書き文字認識機能を搭載する情報処理装置を実現することができる。

【0198】

より具体的には、手書き入力による文字列を入力した後に、例えば、ワープロアプリケーションで作成した平仮名漢字変換による漢字文字列を入力し、更に、手書き入力による文字列を入力した場合でも、これらの入力した文字列を1つの処理対象として後処理（実施形態2では、例えば、連鎖確率算出処理）を実行することができる。これにより、ユーザが意図する文字列を、効率的にかつ適切な処理単位で後処理を実現することができ、より認識率の高い文字認識環境を提供することができる。

20

【0199】

以上、実施形態例を詳述したが、本発明は、例えば、システム、装置、方法、プログラムもしくは記憶媒体等としての実施態様をとることが可能であり、具体的には、複数の機器から構成されるシステムに適用しても良いし、また、一つの機器からなる装置に適用しても良い。

【0200】

尚、本発明は、前述した実施形態の機能を実現するソフトウェアのプログラム（実施形態では図に示すフローチャートに対応したプログラム）を、システムあるいは装置に直接あるいは遠隔から供給し、そのシステムあるいは装置のコンピュータが該供給されたプログラムコードを読み出して実行することによっても達成される場合を含む。

30

【0201】

従って、本発明の機能処理をコンピュータで実現するために、該コンピュータにインストールされるプログラムコード自体も本発明を実現するものである。つまり、本発明は、本発明の機能処理を実現するためのコンピュータプログラム自体も含まれる。

【0202】

その場合、プログラムの機能を有していれば、オブジェクトコード、インタプリタにより実行されるプログラム、OSに供給するスクリプトデータ等の形態であっても良い。

40

【0203】

プログラムを供給するための記録媒体としては、例えば、フロッピー（登録商標）ディスク、ハードディスク、光ディスク、光磁気ディスク、MO、CD-ROM、CD-R、CD-RW、磁気テープ、不揮発性のメモ리카ード、ROM、DVD（DVD-ROM、DVD-R）などがある。

【0204】

その他、プログラムの供給方法としては、クライアントコンピュータのブラウザを用いてインターネットのホームページに接続し、該ホームページから本発明のコンピュータプログラムそのもの、もしくは圧縮され自動インストール機能を含むファイルをハードディスク等の記録媒体にダウンロードすることによっても供給できる。また、本発明のプログ

50

ラムを構成するプログラムコードを複数のファイルに分割し、それぞれのファイルを異なるホームページからダウンロードすることによっても実現可能である。つまり、本発明の機能処理をコンピュータで実現するためのプログラムファイルを複数のユーザに対してダウンロードさせるWWWサーバも、本発明に含まれるものである。

【0205】

また、本発明のプログラムを暗号化してCD-ROM等の記憶媒体に格納してユーザに配布し、所定の条件をクリアしたユーザに対し、インターネットを介してホームページから暗号化を解く鍵情報をダウンロードさせ、その鍵情報を使用することにより暗号化されたプログラムを実行してコンピュータにインストールさせて実現することも可能である。

【0206】

また、コンピュータが、読み出したプログラムを実行することによって、前述した実施形態の機能が実現される他、そのプログラムの指示に基づき、コンピュータ上で稼動しているOSなどが、実際の処理の一部または全部を行ない、その処理によっても前述した実施形態の機能が実現され得る。

【0207】

さらに、記録媒体から読み出されたプログラムが、コンピュータに挿入された機能拡張ボードやコンピュータに接続された機能拡張ユニットに備わるメモリに書き込まれた後、そのプログラムの指示に基づき、その機能拡張ボードや機能拡張ユニットに備わるCPUなどが実際の処理の一部または全部を行ない、その処理によっても前述した実施形態の機能が実現される。

【図面の簡単な説明】

【0208】

【図1】本発明の実施形態1の文字認識装置の使用形態の一例を示す図である。

【図2】本発明の実施形態1の文字認識装置として機能する情報携帯機器のハードウェア構成を示す図である。

【図3】本発明の実施形態1の携帯情報機器の機能構成を示す図である。

【図4】本発明の実施形態1の後処理継続可否判定テーブルを示す図である。

【図5】本発明の実施形態1の文字認識処理の具体例を説明するための図である。

【図6】本発明の実施形態1の手書き文字入力ユーティリティプログラムが生成する手書き文字入力用画面の一例を示す図である。

【図7】本発明の実施形態1の文字列後処理継続判定処理を示すフローチャートである。

【図8】本発明の実施形態1の手書き文字認識処理を示すフローチャートである。

【図9】本発明の実施形態2の携帯情報機器の機能構成を示す図である。

【図10】本発明の実施形態2の文字認識処理の具体例を説明するための図である。

【図11】本発明の実施形態2の文字認識処理の具体例を説明するための図である。

【図12】本発明の実施形態2の文字認識処理の具体例を説明するための図である。

【図13】本発明の実施形態2の挿入処理を示すフローチャートである。

【図14】本発明の実施形態2の後処理対象の文字列の組み合わせ例を示す図である。

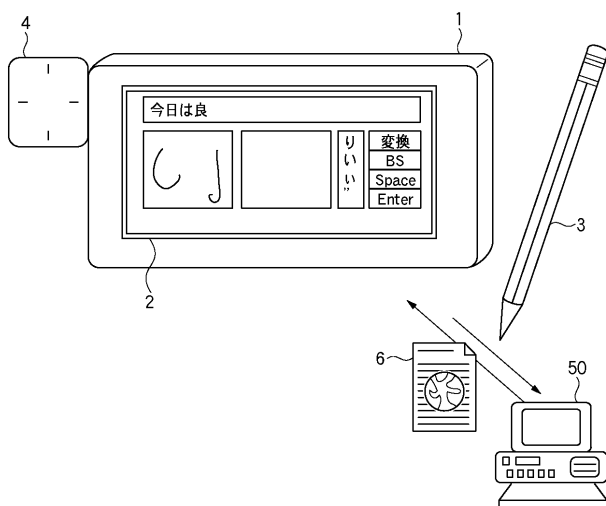
【符号の説明】

【0209】

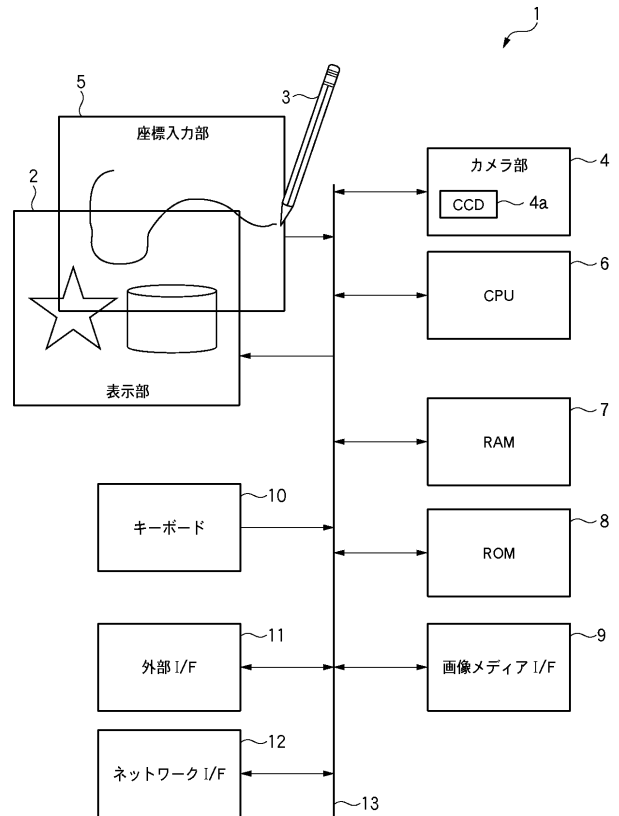
- 1 携帯情報機器
- 2 表示部
- 3 指示具
- 4 カメラ部
- 4a CCD
- 5 座標入力部
- 6 CPU
- 7 RAM
- 8 ROM
- 9 メディアI/F

- 1 0 キーボード
- 1 1 外部 I / F
- 1 2 ネットワーク I / F
- 1 3 システムバス

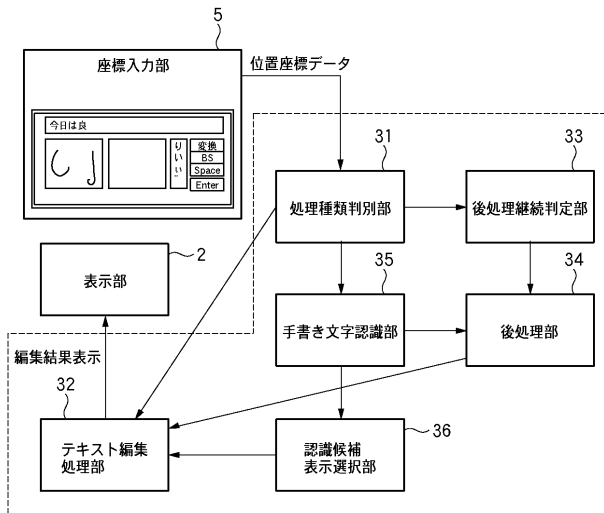
【図 1】



【図 2】



【図 3】

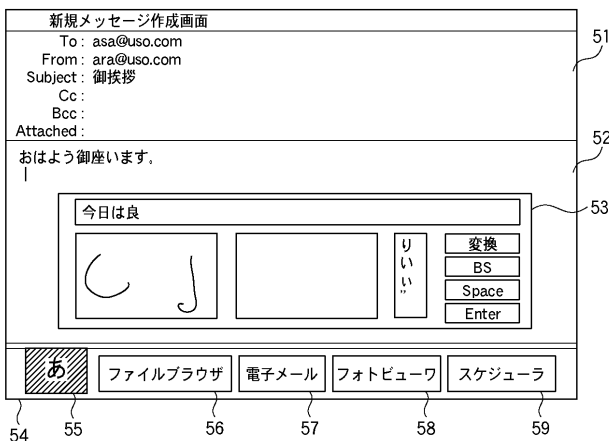


31: Determination Unit;

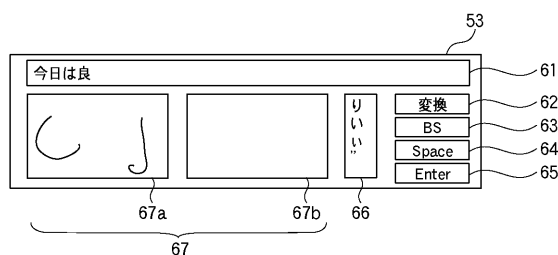
【図 4】

処理種類	後処理継続可否
候補変更処理	後処理文字列の継続を中止
手書き文字入力	後処理文字列の継続
文字切り操作	後処理文字列の継続
仮名漢字変換処理の指示	後処理文字列の継続を中止
1文字消去処理	後処理文字列の継続
スペース入力	後処理文字列の継続を中止
Enter入力	後処理文字列の継続を中止
アプリケーションの切替	後処理文字列の継続

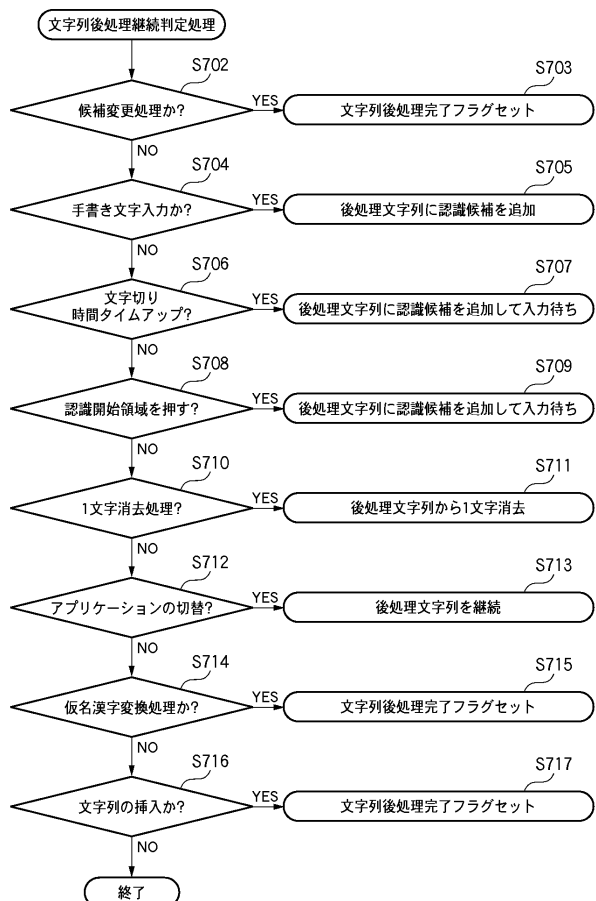
【図 5】



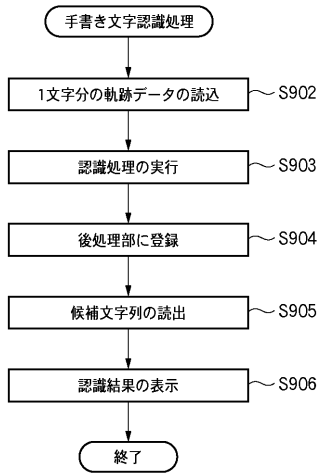
【図 6】



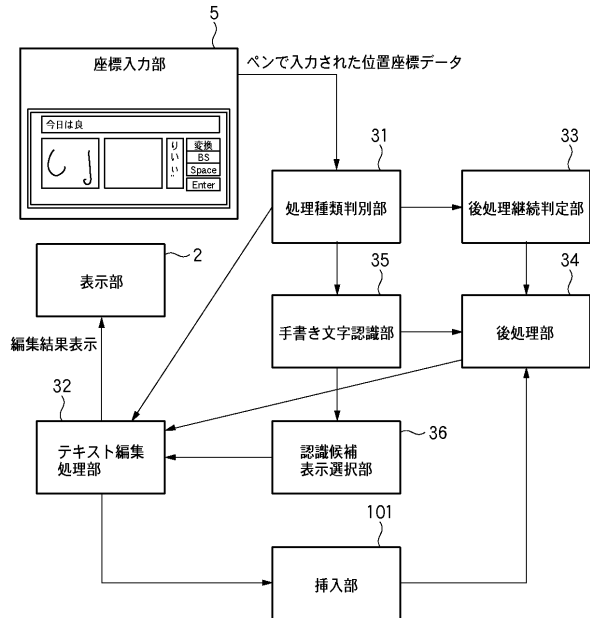
【図 7】



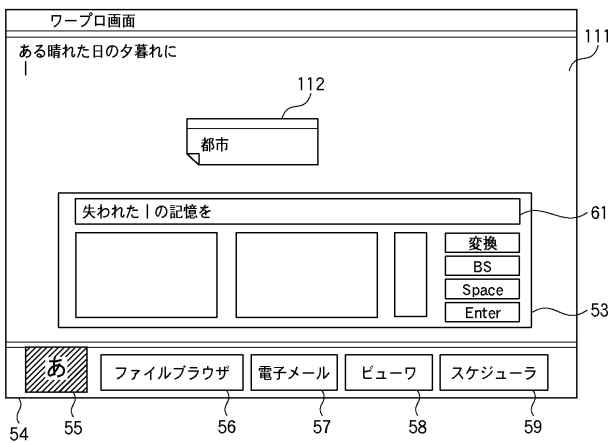
【図 8】



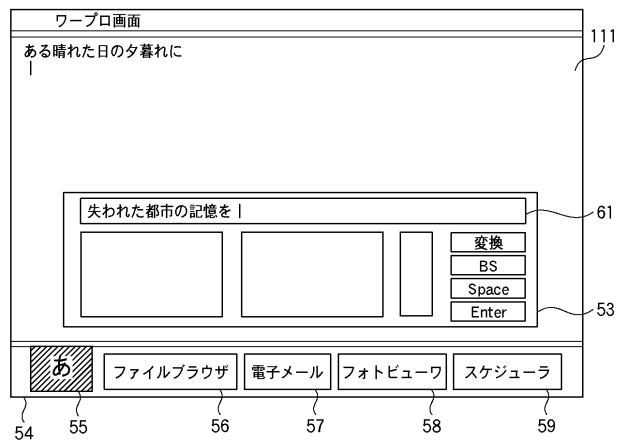
【図 9】



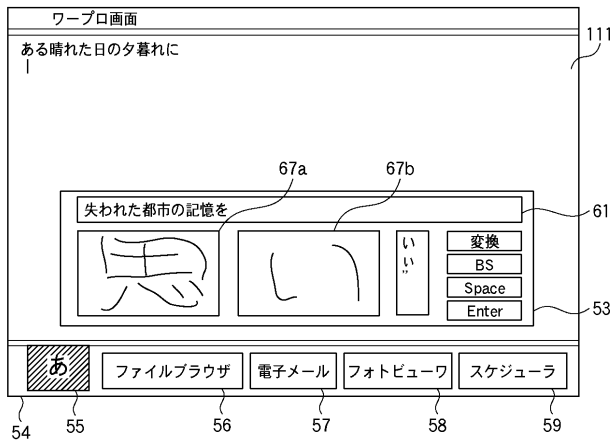
【図 10】



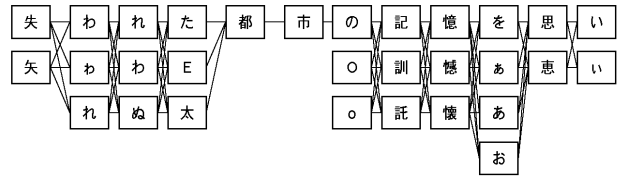
【図 11】



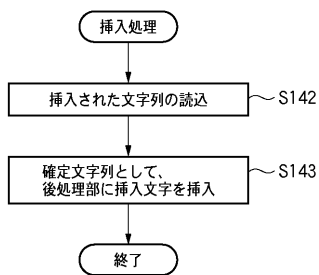
【図 1 2】



【図 1 4】



【図 1 3】



フロントページの続き

Fターム(参考) 5B064 AB04 BA05 BA06 CA11 DD02 DD11 DD16 EA08 EA16 EA18
FA04 FA05 FA13
5B068 BB06 CC19