(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property
Organization
International Bureau

(43) International Publication Date
11 July 2013 (11.07.2013)

WIPO | PCT

(10) International Publication Number
WO 2013/102892 Al

(51) International Patent Classification:
G10L 15/18 (2013.01) *G06F 3/01* (2006.01)
G10L 15/28 (2013.01) *G06F 3/00* (2006.01)
G06F 3/048 (2013.01)

(21) International Application Number:
PCT/IB20 13/050 123

(22) International Filing Date:
7 January 2013 (07.01 .2013)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
61/583,775 6 January 2012 (06.01 .2012) US

(71) Applicant *(for all designated States except US):* **TECH¬NOLOGIES OF VOICE INTERFACE LTD** [IL/IL]; 4/9 Yachin, Ramat Gan (IL).

(72) Inventor; and
(71) Applicant *(for US only):* **WEISSBERG, Danny** [IL/IL]; 4/9 Yachin St, Ramat Gan (IL).

(74) Agents: **EITAN, MEHULAL & SADOT** et al; 10 Abba Eban Blvd, Pob 2081, 46120 Herzliya (IL).

(81) **Designated States** *(unless otherwise indicated, for every kind of national protection available):* AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IS, JP, KE, KG, KM, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.

(84) **Designated States** *(unless otherwise indicated, for every kind cf regional protection available):* ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), Eurasian (AM, AZ, BY, KG, KZ, RU, TJ, TM), European (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT, NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

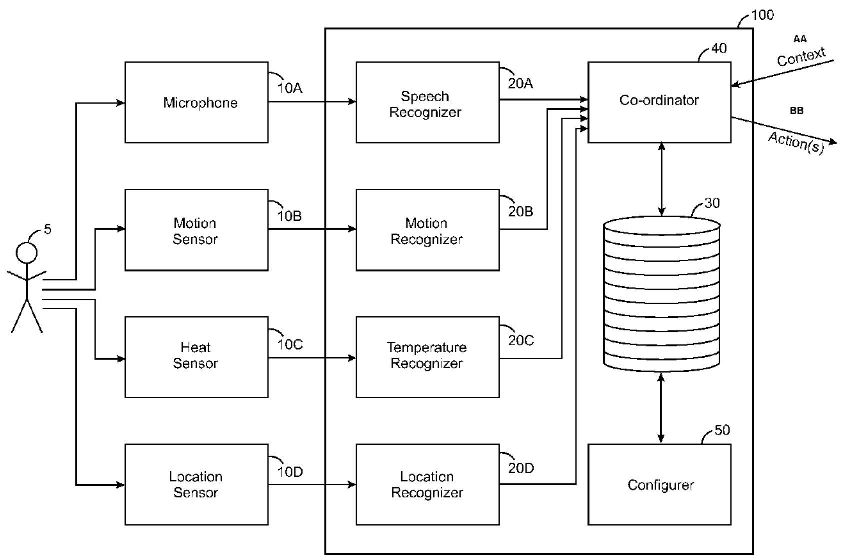
Published:

— *with international search report (Art. 21(3))*

— *before the expiration cf the time limit for amending the claims and to be republished in the event cf receipt cf amendments (Rule 48.2(h))*

(54) **Title:** A SYSTEM AND METHOD FOR GENERATING PERSONALIZED SENSOR-BASED ACTIVATION OF SOFTWARE

Fig. 2



(57) **Abstract: A** system for providing a natural user interface to an application including a personalized natural user interface to at least partially override a user interface of said application and a control element to receive an action from the personalized natural user interface and to provide a control instruction to the application.

TITLE OF THE INVENTION

A SYSTEM AND METHOD FOR GENERATING PERSONALIZED SENSOR-BASED ACTIVATION OF SOFTWARE

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims benefit from U.S. Provisional Patent Applications No. 61/583,775, filed January 6, 2012 which is hereby incorporated in its entirety by reference.

FIELD OF THE INVENTION

[0002] The present invention relates to sensor-based activation of software.

BACKGROUND OF THE INVENTION

[0003] Conventional computerized devices, such as personal computers, laptop computers, smartphones, tablets and the like, utilize graphical user interfaces in applications, enabling users to quickly provide input to applications. In general, using a graphical user interface, a user operates an input device such as a mouse, keyboard or touchscreen to manipulate graphical objects on a computer display. The graphical objects are often represented as icons, elements or widgets such as buttons, check boxes, toolbars, hyperlinks etc., and the user can operate an input device, such as a mouse, to activate the application through the graphical object. For example, the user can activate a required function in the application by positioning a mouse pointer over any form of user interface element such as a button and clicking the mouse or by using a touchpad.

[0004] Today there is an increasing demand to use a more natural way of communicating with such computerized devices in order to keep up with our busy lives and schedules. Natural User Interface (NUI) (voice, motion etc.) is known in the art where input (activation) is provided in a more natural form such as voice and motion. For example voice instructions may be given to a mobile phone while driving in order to remain "hands-free". Such instructions may include playing music, dialing a specific phone number etc. Other examples can include tapping or shaking a device to activate it. NUI is becoming increasingly popular and may be used in diverse applications such as games, healthcare (doctors, paramedics) systems, social networks applications, inventory, field sales-force applications, computer support applications, interactive menu systems, games, educational software, etc.

SUMMARY OF THE PRESENT INVENTION

[0005] There is provided, in accordance with a preferred embodiment of the present invention a system for providing a natural user interface to an application. The system includes a personalized natural user interface to at least partially override a user interface of the application and a control element to receive an action from the personalized natural user interface and to provide a control instruction to the application.

[0006] Moreover, in accordance with a preferred embodiment of the present invention, the system for providing a natural user interface to an application also includes an input device to receive natural input and where the input device is at least one of a microphone, a motion sensor, an accelerometer, a gyroscope, a magnetometer, a camera, a location sensor and a temperature sensor.

[0007] Further, in accordance with a preferred embodiment of the present invention, the personalized natural user interface includes at least one recognizer to recognize at least one natural input as a formatted output.

[0008] Still further, in accordance with a preferred embodiment of the present invention, the recognizer is at least one of a speech recognizer, a voice recognizer, a motion recognizer, a location recognizer, a temperature recognizer and a facial expression recognizer.

[0009] Additionally, in accordance with a preferred embodiment of the present invention, the recognizer includes an external support analyzer to analyze if the recognizing is to be performed on at least one of a local server and a remote server.

[0010] Moreover, in accordance with a preferred embodiment of the present invention, the personalized natural user interface also includes a database to store associations of formatted output and associated actions.

[0011] Further, in accordance with a preferred embodiment of the present invention, the associations are one of: pre-determined and user generated.

[0012] Still further, in accordance with a preferred embodiment of the present invention, the database also stores application context and properties of elements of the user interface of the application.

[0013] Additionally, in accordance with a preferred embodiment of the present invention, the personalized natural user interface includes a coordinator to receive the formatted output, to generate at least one action associated with the formatted output from the database and to provide at least one action to the control element.

[0014] Moreover, in accordance with a preferred embodiment of the present invention, the personalized natural user interface also includes a configurer to receive the natural input from a user and associate the natural input with at least one action.

[0015] Further, in accordance with a preferred embodiment of the present invention, the control element includes an agent to communicate with the application.

[0016] Still further, in accordance with a preferred embodiment of the present invention, the agent is one of an emulator, a native user interface agent and a web browser add-on.

[0017] Additionally, in accordance with a preferred embodiment of the present invention, the control element is a software development kit (SDK).

[0018] Moreover, in accordance with a preferred embodiment of the present invention, the application is one of a web-based application and a non-web-based application.

[0019] There is provided, in accordance with a preferred embodiment of the present invention a method for converting natural input into an associated action or chain of actions and providing the associated action or chain of actions as input to the application.

[0020] Moreover, in accordance with a preferred embodiment of the present invention, the converting also includes receiving natural input via an input device and where the input device is at least one of a microphone, a motion sensor, an accelerometer, a gyroscope, a magnetometer, a camera, a location sensor and a temperature sensor.

[0021] Further, in accordance with a preferred embodiment of the present invention, the converting also includes recognizing via a recognizer said natural input as a formatted output.

[0022] Still further, in accordance with a preferred embodiment of the present invention, the recognizer is at least one of a speech recognizer, a voice recognizer, a motion recognizer, a location recognizer, a temperature recognizer and a facial expression recognizer.

[0023] Additionally, in accordance with a preferred embodiment of the present invention, the converting also includes saving the associated action or chains of actions in a database.

[0024] Moreover, in accordance with a preferred embodiment of the present invention, the associations are one of: pre-determined and user generated.

[0025] Further, in accordance with a preferred embodiment of the present invention, the converting also includes generating the action or chain of actions associated with the formatted output.

[0026] Still further, in accordance with a preferred embodiment of the present invention, the converting also includes configuring the natural input to be associated with the action or chain of actions.

[0027] Further, in accordance with a preferred embodiment of the present invention, the providing includes communicating with the application via an agent.

[0028] Additionally, in accordance with a preferred embodiment of the present invention, the providing also includes communicating with the application via an SDK.

[0029] Moreover, in accordance with a preferred embodiment of the present invention, the agent is one of an emulator, a native user interface agent and a web browser add-on.

[0030] Further, in accordance with a preferred embodiment of the present invention, the application is one of a web-based application and a non-web-based application.

BRIEF DESCRIPTION OF THE DRAWINGS

[0031] The subject matter regarded as the invention is particularly pointed out and distinctly claimed in the concluding portion of the specification. The invention, however, both as to organization and method of operation, together with objects, features, and advantages thereof, may best be understood by reference to the following detailed description when read with the accompanying drawings in which:

[0032] Figs. 1A and IB are schematic illustrations of unit for providing a control instruction from a personalized natural user interface, constructed and operative in accordance with the present invention;

[0033] Fig. 2 is a block diagram of personalized natural interface activation system in accordance with the present invention;

[0034] Fig. 3 is a flowchart of the runtime functionality of the personalized natural interface activation system of Fig. 2; in accordance with the present invention;

[0035] Fig. 4 is a schematic illustration of how more than one input or event may be configured to map to one or more actions for a particular application context in accordance with the present invention;

[0036] Fig. 5 is a schematic illustration of how associations are held in the database in accordance with the present invention;

[0037] Fig. 6 is a flowchart of the lookup functionality of the coordinator of Fig. 2 in accordance with the present invention;

[0038] Fig. 7 is a flowchart of the action retrieval functionality of the coordinator of Fig. 2. in accordance with the present invention;

[0039] Fig. 8 is a flowchart of the configuration functionality of the configurer of Fig. 2. in accordance with the present invention;

[0040] Fig. 9 is a schematic illustration of the communication methods between the unit of Figs. 1 and 2 and an application, in accordance with the present invention;

[0041] Fig. 10 is a schematic illustration of a user interface wizard in accordance with the present invention;

[0042] Fig. 11 is a schematic illustration of a seamless UI in accordance with the present invention;

[0043] Fig. 12 is a schematic illustration of storage and sharing of the database of Fig. 2 on a remote server, in accordance with the present invention;

[0044] Fig. 13 is a block diagram of the system of Fig. 2 together with an external recognizer in accordance with the present invention.

[0045] It will be appreciated that for simplicity and clarity of illustration, elements shown in the figures have not necessarily been drawn to scale. For example, the dimensions of some of the elements may be exaggerated relative to other elements for clarity. Further, where considered appropriate, reference numerals may be repeated among the figures to indicate corresponding or analogous elements.

DETAILED DESCRIPTION OF THE PRESENT INVENTION

[0046] In the following detailed description, numerous specific details are set forth in order to provide a thorough understanding of the invention. However, it will be understood by those skilled in the art that the present invention may be practiced without these specific details. In other instances, well-known methods, procedures, and components have not been described in detail so as not to obscure the present invention.

[0047] Applicants have realized that current methods of natural user interface (NUI) and in particular the use of NUI to provide input to 3rd party applications require substantial time and effort in the development of applications and platforms in order to implement such functionality. In order to support NUI for an application, it is necessary to create the correct content and to develop code to support NUI for each application separately. It will further be appreciated, that these forms of NUI may also be limiting. For example, a soccer voice control game may be limited to the control word "kick' to kick a ball, whereas the user may wish to use the word "go" or a sound "boom" instead.

[0048] Applicants have also realized that that are other forms of computerized devices aside from those discussed in the background such large household appliances - air conditioner units, ovens, refrigerators etc. as well as smaller appliances such as vacuum cleaners, televisions etc. Such appliances may be considered computerized since they also have a user interface which is controlled by some form of control software. Input is usually provided to the control software of these appliances through the use of touch/switches, remote controls etc. to change appliance settings.

[0049] Applicants have further realized that an extra interface to provide a NUI may be added between a user and any control software or 3rd party application installed on a computing device and that this NUI may be configured and personalized by any user with or without extensive computer knowledge.. It will be appreciated that this personalized NUI (PNUI) may be used to override or partially override the pertinent user interface of an application to be controlled by providing an automatic user interface control instruction that may replace the current control. The application in question may receive a pertinent control from the NUI instead of its own user interface. Reference is now made to Fig. 1A which illustrates a unit 105A for providing a control instruction from personalized natural user interface (PNUI) 100 via a UI controller 8 to an application 15. For example, a user 5 may say the word "JUMP" with the intention of making a character in an avatar game jump. The speech input may be captured by PNUI 100 and matched to an associated action which may be provided to controller 8. Controller 8 may then forward a control instruction to match the associated action to application 15 which may replace the physical activation of the 'jump' button 3 of application 15.

[0050] It will also be appreciated that an alternative method of overriding the pertinent user interface of application 15 may be through the use of a software development kit (SDK) 9 in conjunction with PNUI 100 as is illustrated in Fig. IB which illustrates a unit 105B to be used with application 15 and to which reference is now made. Application 15 may seek a mapping match for a UI control or other application functionality from PNUI 100 via SDK 9 instead of from its own user interface. In another embodiment, PNUI 100 may override the UI the application 15 using its application programming interface (API).

[0051] Reference is now made to Fig .2 which illustrates a PNUI activation system 100 which may be installed on a computerized device in accordance with an embodiment of the present invention. Reference is also made to Fig. 3 which illustrates the runtime flow of events for PNUI 100. PNUI 100 may comprise one or more input devices 10 (such as a microphone 10A, a motion sensor 10B, a heat sensor IOC and/or a location sensor 10D), one or more input recognizers 20 (such as speech recognizer 20A, a motion recognizer 20B, a temperature recognizer 20C and a location recognizer 20D), a database 30, a coordinator 40 and a configurer 50.

[0052] It will be appreciated that PNUI 100 may have three main flows, at application start up, during user configuration and at run time. For the sake of clarity, the functionality of PNUI 100 is described with a run-time flow.

[0053] Once application 15 has been activated (step 110 of Fig. 3), an input (an event) may be received by the relevant input device 10 (step 120) and sent to the relevant recognizer 20 (step 130). It will be appreciated that for each input device 10 in use, there may be one or more matching recognizers 20. For example a speech signal may be received by microphone 10A and may be forwarded to speech recognizer 10B. Likewise, motion may be detected using motion sensor 12 and may be forwarded to motion recognizer IOC etc.

[0054] Each recognizer 20 may interpret the pertinent input (step 140) (audio signal, motion, temperature etc.) and may convert it to a formatted output such as an alphanumeric or text format according to methods known in the art. It will be appreciated that for each type of recognizer 20, the formatted output may vary according to the type of recognizer. Examples of speech recognizers include the speech recognizers, implemented as Application Programming Interfaces (APIs) and commercially available from Google Inc. or from Nuance Communications, Inc. An exemplary motion detector is Kinect (commercially available from Microsoft Corporation Inc.). It will be appreciated that recognizers 20 may be potentially implemented by a third party API as described herein above or may have a proprietary implementation such as a voice tag recognizer. Voice tags are known in the art and may be used in automated speech recognition in a voice control device to allow users to "speak" commands. For the sake of clarity, recognizers that provide a text output are used in this discussion. For example, for the above mentioned example, the speech signal corresponding to the word "jump" may be picked up by microphone 11 and may be converted into a text output of the word "jump". In an alternative embodiment, predefined voice tags may be recognized by speech recognizer 21 thus providing a different form of output for association. For example, a user can record a voice tag such as "boo" to kick a ball in a soccer game.

[0055] It will also be appreciated that PNUI 100 may be active for a particular application context or state only. It will be appreciated that some PNUI 100 configurations may be global, i.e. active for all applications in any state of the application such as the functionality of up, down, right and left keys in a gaming application. Other configurations may be active for a specific application, but on any screen and state of the application, for example, the connection between the "LIKE" voice command and the like button may only be active in the context of the Facebook application and a Facebook screen showing the "like' button'. Another context may be the sub-state where the application context may be divided into sub states. For example, state may be a particular screen, and every change may be identified as a sub-state. Therefore when the screen changes, a particular configuration may no longer be valid. It will be appreciated that context for a particular event may be set during configuration as is described herein below.

[0056] It will be appreciated that the current context of application 15 may be obtained during runtime via controller 8 or via SDK 9 which may receive from application 15 information regarding the active application itself and any current context including pertinent states and sub-states as described in more detail herein below.

[0057] The context of the application 15 may be obtained by coordinator 40 from controller 8 or via SDK 9 as is described hereinabove. The text output of each recognizer 20 may then be passed to coordinator 40 (step 160), which may be connected to database 30. It will be appreciated that database 30 may store associations between the text output of the appropriate recognizers 20 and desired actions, within a particular context. For example, the word "right" may be associated with the action "move right" for a character within the first three levels of a particular game. Coordinator 40 may attempt to retrieve from database 30 an associated action or actions for the word "right" (step 170) within the correct context and if a suitable match is found (180), may provide action instructions to controller 8 (step 190). If no match is found, the process may be repeated.

[0058] It will be appreciated that there may be multiple forms of input and that for each form of input there may be an appropriate input device 10. For example, microphone 10A may capture external sound. The raw sound captured may be interpreted into an event using either a speech recognizer 20A (either proprietary or 3rd party or a hybrid of both as described in more detail herein below) or another form of voice recognizer 20 which may recognize voice tags. It will also be appreciated that another form of input device may be motion sensor 10B which may detect motion and may be an accelerometer, a gyroscope, a magnetometers etc. The motion captured may be analyzed by motion recognizer 20B. Other forms of input may include facial expression, location and temperature which may be captured by optical and image sensors such as device cameras, global positioning systems, thermometers, compass sensors, RFID readers and clocks and which all may be analyzed by using a pertinent recognizer 20.. It will be appreciated that some of the above mentioned input devices may be built in to the operating system of the pertinent device or may be 3rd party software.

[0059] It will be appreciated that coordinator 40 may receive as input a single or chain of events from multiple recognizers 20. For example, PNUI 100 may be configured (as described in more detail herein below) to match the motion of a hand wave with the speech input of the word "BYE" to close a particular 3rd party application. Reference is now made to Fig. 4 which illustrates how more than one input or event may be configured to map to one or more actions for a particular application context. It will also be appreciated that each chain of events may comprise multiple events (such as the hand wave and the "BYE") with the relationship between them set with pre-defined order together with a predefined time delay. It will be appreciated that the events may also be configured to happen simultaneously. It will further be appreciated that one or many events, may be mapped to one or many actions.

[0060] It will also be appreciated that database 30 may hold the matches between "events" (input received via devices 10) and "actions" (instructions to controller 8) within a particular context. As described hereinabove, a chain of events can be one event or more. It will be further appreciated that the relationship between different events in the same chain may also be stored in database 30, together with instructions as to whether multiple events should happen simultaneously or in a particular order with time delays between them etc. The same information may also be held for actions. Database 30 may be configured by default values pre-configured by the system, by user configuration using configurer 50 or by sharing configurations as part of a social community (described in more detail herein below).

[0061] It will be further appreciated that an association between an incoming event such as the word "Dad" and the outgoing action "instruct cellphone to call 123456" may be broadened. The command "Brian" may also be associated with the same action "instruct cellphone to call 123456". Thus the words "Dad" and "Brian" may both be used as triggers for the action "instruct cellphone to call "12345 as is illustrated in Fig.5 which shows examples of different associations that may be held in database 30 and to which reference is now made. As discussed hereinabove, an incoming event or chain of events may be configured to map to a particular action or chain of actions.

[0062] Reference is now made to Fig. 6 which illustrates a flow chart of the lookup functionality of coordinator 40. Coordinator 40 may receive an event (step 220) from a recognizer 20. It may then add the event to a current chain of events (step 230). Once the complete chain of events has been received, a match is made with each event with database 30 (step 240). If a complete match is made of all the events in database 30 (step 250), then the pertinent action or chain of actions is activated (step 270). If no match is made, then a check is made to see if a particular event is missing from the current chain (step 260). If an event is missing, then the event is retrieved (step 220). If a partial match is made (some events match up and not others), then a second check may be made within database 30 to see whether there is a similar chain of events stored in database 30 (step 220). .If no similar match is found, the current chain of events is cleared and the entire process may be repeated (step 280). It will also be appreciated that matches are also made according to the application context. It will be further appreciated that each 'action' may be considered an instruction to controller 8. It will be further appreciated that each single event may be configured to match an individual action or a chain of actions. Similarly, a chain of events may be configured to match a single action or a chain of actions. There may be many permutations of matches between events and actions (two events -> three actions etc.)

[0063] Reference is now made to Fig. 7 which illustrates a flow chart of the activation of a chain of actions (or single action) once they have been activated as described herein above. An action is retrieved (either first or subsequent) (step 310) and added to the current list of actions (step 320) together with any pre-defined relationship between the current action and the previous action (step 330). If the relationship has been defined that both actions are to be performed simultaneously then the next action is retrieved immediately and added to the list. It will be appreciated that the list may hold more than one action if they are to be performed simultaneously. It will be also be appreciated that if the actions are not defined as simultaneous, then only one action will be held on the list at any time. If the relationship between subsequent actions has been defined with a time delay between them, another action is not retrieved, the pertinent time delay is waited and the current action (or actions that are on the current list of actions) is sent to controller 8 (step 340). Once controller 8 has received pertinent action, it is removed from the list (step 350) and the next action is sought. Steps 310-350 and are repeated until controller 8 has been instructed of all the pertinent actions. Once there are no more actions to be retrieved (step 310), the process is stopped.

[0064] It will be appreciated that there may be four types of actions that may be configured to be activated by an event or series of events. The first type action may be to send information to the pertinent device on which PNUI 100 is installed such as showing a particular picture on a screen, changing its brightness etc. Another type of action in this category may be to prepare voice feedback to user 5 via the device speaker or to vibrate the device. The second type of action may be to activate the device functionalities of the pertinent device. It will be appreciated that all device functionalities have a programmable API. The built in functionalities may include sending an SMS, turning Wi-Fi/ Bluetooth etc. on and off, creating a call, browsing multimedia etc.

[0065] The third type of action may be instructing a 3rd party application, i.e. overriding a particular control such as pressing a button or in the case of using an SDK to control the 3rd party application (as described herein above), the activation may be through SDK callback. The fourth type of action may be to activate a secondary 3rd party application via a primary 3rd party application.

[0066] In an alternative embodiment to the present invention, coordinator 40 may not receive input via input device 10 but from a different form of trigger such as an application based event. It will also be appreciated that the application to be controlled 15 by such an application based event may be the same application providing the input. For example, changes in application resources may be configured to be recognized as suitable input. A significant change in CPU usage by application 15 may be configured to cause the pertinent device on which it is installed to vibrate when a threshold level is reached.

[0067] It will be appreciated that configurer 50 may enable user 5 to personalize his own PNUI 100. Configurer 50 may set the above mentioned relationships between events and actions as well as between events in chains of events and actions in chains of actions. Reference is now made to Fig. 8 which illustrates a flowchart showing the functionality of configurer 50. Configurer 50 may be activated (step 500) and the pertinent application that is to be controlled may be selected (step 505). Next the input type is selected (step 510), for example whether the pertinent event is triggered by voice, motion etc. It will be appreciated that a particular input type may be pre-defined as the default such as all input is voice. Once the input and context has been determined, a value is set (step 520) for the particular event (as described in more detail herein below). As is described herein above, the pertinent recognizer 20 may convert the input from the device 10 in question into text output. It is this text output that is assigned a value (label or tag). For example the incoming speech signal for the word "BYE" may be converted into a text output by speech recognizer 21 and then assigned a label 'bye" It is this label that is held in database 30 for further associations with actions.

[0068] If there is more than one event that requires configuring, a relationship may be set between the current input and the next input (step 540), such as sequence or timing between them and then the next event is selected (step 510). This looping (step 530) may continue until all pertinent inputs have been received, assigned a label and had the relationships between them defined. Next the action/command type is selected (step 550) such as voice output, screen output, control of application 15 etc. Once the output type has been determined, the appropriate command may be set (step 560). It will also be appreciated that a particular output type may be defined as a default. For example for the event that has been labeled "bye", the matching command may be "close down" a particular 3rd party application. If there are additional commands that need to be configured (step 570), the relationship is set between the current command and the command event (step 580), such as sequence or timing and then the next event is selected (step 370). Once again, "looping' may occur until all the pertinent commands are configured. Once they have all been configured, the pertinent configuration may be stored in database 30.

[0069] It will be appreciated, that in order for unit 105 to fully control application 15 and in order to allow user 5 to configure PNUI 100, controller 8 may comprise an agent 85 as a communication medium between controller 8 and the UI of application 15. As discussed herein above, there may be four types of control instruction to application 15 including activating the device functionalities of the pertinent device on which application 15 is installed and instructing a 3rd party application by overriding a particular control such as pressing a button. It will be appreciated that overriding or partially overriding the device functionalities of the pertinent device as well as 3rd party application controls may require identifying the UI elements from application 15 together with their properties as well as 'hooking' on to them in order to modify them or to emulate their functionality. It will also be appreciated that agent 85 may have the ability to do this during configuration. Agent 85 may also receive any updates as to changes in functionality and/or context from application 15 during runtime such as the creation of a new window in a voice control game. It will also be appreciated, that agent 85 may also instruct application 15 to load itself into a particular context of the application. It will be further appreciated that in the case of an SDK when no controller 8 is present, context information may be received by coordinator 40 via the API of application 15.

[0070] It will also be appreciated that UI elements may come in various shapes and sizes and may range from buttons, check boxes, labels, progress bars etc. to avatars in gaming, hyperlinks and widgets. It will also be appreciated all UI elements may have multiple properties such as type, caption, color, size etc. It will be further appreciated that the activation of a UI element may include, pressing a button, checking a checkbox, selecting an item in a pull down menu etc. Modification of a UI element may include changing different parameters such as color or size or may include creating or destroying an element.

[0071] Reference is now made to Fig. 9 which illustrates how agent 85 may communicate with the UI elements of application 15 in order to identify, 'hook' and control them. It will be further appreciated that there may be different methods of implementation depending on the type of application 15 and its environment. If application 15 is a 3rd party web application (71), a standard method may be to extend the current capabilities and features of the web browser in use (Explorer, Safari etc.) Add-on software may be used that may sit inside the browser and search for UI elements to activate. An alternative method may be using a customized web browser (72) added to agent 85. This may be very useful in devices where there is no add-on support.

[0072] For certain non-web based applications 15 (73) using technologies such as Win32; agent 85 may be required to run in the context of application 15 in order to communicate with its UI elements and to either extract property information for storage in database 30 or to modify their properties. In another embodiment, agent 85 may also run within the context of unit 105 and not within application 15. Another embodiment may be through an emulator (74). Such a program may be native to the pertinent device in use and may aid users to develop their own applications by emulating the pertinent operating system. This emulation capability may then be extended by using an add-on agent 85 in order to control application 15. It will be appreciated that all mobile platforms may have emulator software. In some cases, the emulator maybe SDK 9.

[0073] It will also be appreciated that agent 85 may understand the pertinent technology of the UI of application 15 such as Win 32, .Net, Java, ObjectiveC etc. It will also be appreciated that agent 85 may be generic at the technology level and that once agent 85 supports a particular technology; all other applications running with the same technology may also be supported.

[0074] Once user 5 has selected and configured a UI element, its properties may be stored in database 30 in a manner that will enable the pertinent UI element to be uniquely identified for future use. For example if user 5 says the word "hey", and the word 'hey' has been associated with turning on a particular button, coordinator 40 may need to know which button to instruct controller 8 to activate. It will be appreciated that certain properties of UI elements may be obligatory and certain properties may be optional. For example obligatory properties for a button may include type - button and caption -'hey' in order to provide a unique identification. Optional parameters may include its position (since the button may be moved) and its color (which may be changed)... It will be appreciated that during configuration, all pertinent properties both obligatory and optional may be stored in database 30 for a particular UI element. It will be further appreciated that when controller 8 may seek to identify to which UI element to send his control instruction, coordinator 40 may find a match in database 30 by first seeking obligatory elements. If the obligatory elements searched do not provide a unique element, then the optional properties are also searched. For example, if there are two buttons both stored as type button and with caption "hey", then coordinator 40 may seek an optional property such as the color red in order to identify the pertinent button to be used. It will be further appreciated that controller 8 may retrieve this information from database 30 via coordinator 40 together with the required action.

[0075] It will also be appreciated that the main purpose of configurer 50 is to enable user 5 to personalize the natural user interface of his applications. It will be further appreciated that there may be different ways for a user 5 to configure system 100 through the use of an appropriate user interface. It will also be appreciated that event values as stored in database 30, may be configured by user 5 by recording his events, i.e. user 5 instructs the pertinent device to start recording and then performs the events (voice, motion etc.) in the desired order with the desired timings between them.. It will be also appreciated that events may also be configured manually by selecting from a list of possible values and by manually typing the text of the required value.

[0076] In order to enable such configuration as described hereinabove, PNUI 100 may provide a further interface, to aid user 5 configure his events. One such interface may be through the use of an interface wizard, in which user 5 may be presented with different dialog boxes in sequence and may move between the dialog boxes usually by clicking on a "next" button after he has entered data or configuring information into the current dialog box. If user 5 decides to go back and change any previously entered information, he may do so by clicking on a "previous" button. It will also be appreciated that any standard wizard may be used and may be adapted accordingly for the application in use.

[0077] Reference is now to Fig. 10 which illustrates the stages of using an interface wizard to configure system 100 on a mobile communication device. First the wizard is activated, usually manually by selecting the pertinent wizard application. In an alternative embodiment, the wizard may also be activated using a pre-configured PNUI 100 command. It will be appreciated that in order for the wizard to complete its task, pertinent devices 10 and recognizers 20 must also be activated. User 5 may then be presented with a list of applications 15 for personalizing with a NUI interface (step 610). It will be appreciated only the applications that may be supported by PNUI 100 may be presented. Once a pertinent application 15 has been selected, the wizard may then present to user 5 a pre-defined list of functionalities available for the selected application (step 620). It will be appreciated that each functionality may be an action or a chain of actions and may combine actions on UI elements, APIs etc. It will also be appreciated that functionalities may be nested within functionalities represented as menus and sub menus. Once user 5 has selected his desired functionality, he needs to define the event that will be the trigger for the functionality selected (630). This may be executed either by recording in conjunction with recording module 55, selecting or defining the event value. For example, if user 5 desires to start application ABC on his mobile phone using the voice input of the word "ON", he may first select application ABC from the list of available applications, he may then select the start functionality that appears in the list of available functionality for application ABC and then he may hit the record button and record his voice saying the word "ON". When he has finished speaking, user 5 may stop the recorder and the resultant information may be stored in database 30 for later use.

[0078] In an alternative embodiment, a seamless UI may be used instead of a wizard to configure PNUI 100. It will be appreciated that the seamless UI may allow user 5 to configure an application NUI without leaving the application context and screen, thus eliminating the need for a wizard. It will be further appreciated that the seamless UI may be integrated into the application 15. In order for the seamless UI to function properly, pertinent devices 10 and recognizers 20 must also be activated. It will also be appreciated that the seamless UI may be used in conjunction with the mouse over functionality or touch provided by the host operating system and that agent 85 must be registered to receive 3rd party UI events. It will also be appreciated that the seamless UI may be created by agent 85 during the identification and hooking process of the UI elements of application 15 as described herein above. When user 5 hovers over a UI element belonging to application 15, the element may be identified by agent 85 as configurable and be highlighted. The highlight may be in the form of a pop-up screen or in the form of a change of color, shading etc. of the pertinent element as is illustrated in Fig. 11 to which reference is now made. The 3rd party application user interface (710) may provide various UI elements (710A, 710B, 710C etc.). User 5 may be presented with the same UI screen (720) with the configurable elements highlighted and then may proceed to configure the properties of any pertinent element via a configuration dialogue that may be triggered. It will be appreciated that agent 85 may also highlight UI elements that have an existing configuration. Configurer 50 may then be activated to configure PNUI 100 according to the configuration settings of user 5 It will be appreciated a seamless UI may be generic to a particular UI technology (such as Win32, .Net, Java , Web etc.) and does not have to be configured to a particular application. It will be further appreciated that properties concerning the UI elements that have been selected and configured may be stored in database 30 for subsequent configuration at runtime.

[0079] It will be appreciated that the properties stored in database 30 for a pertinent UI element of application 15 must provide a unique identification for the element. For example as illustrated in Fig. 1A, the properties for button 3 once configured, may provide enough identification that button 3 is the jump button for application 15.

[0080] It will be appreciated that database 30 may be stored locally on the pertinent computing device or may be stored on a remote account server 70 as illustrated in Fig. 12 to which reference is now made.. It will also be appreciated that once configured for a particular application 15, database 30 may be used with any other computerized device to control the same application 15 such as on cars, tablets, personal computers and smart-TVs. For example, user 5 may configure a command for Facebook which may be also used on a mobile communication device, a tablet, a smart television screen, a personal computer etc. and on different platforms such as Android, iPhone, iMac, Windows etc. It will be appreciated that if user 5 decides that the configuration is only for a specific device or platform, the configuration may be stored locally, else the configuration may be stored on account server 70. It will also be appreciated that the ability to access database 30 remotely may allow for configuration to be performed by other users and may allow for other users to share the associations and configurations that are held. It will also be appreciated that database 30 may be shared amongst other users by uploading database 30 to a shared server 75 that may be accessed by all users in a particular social group using the standard group sharing rules. It will be further appreciated that synchronization with servers 70 and/or 75 may be done manually by user 5 whenever he has an update or automatically using standard timers.

[0081] As discussed hereinabove, recognizer 20 may be installed on a server 80 instead or in addition to the local device as is illustrated in Fig. 13 to which reference is now made. It will be appreciated that the recognizer installed on server 80 may be either 3rd party or propriety. It will also be appreciated that recognizer 20 installed locally on the pertinent device may also be 3rd party or proprietary. Each recognizer 20 may include a sub-module comprising an external support analyzer 22 to decide if analysis of the input is to be performed locally or externally on server 80. For example, speech recognizer 20A may only have the ability to recognize a limited list of words. External support analyzer 22A may decide to activate an external recognizer situated on server 80 in order recognize the other unsupported words. Alternatively both recognizers (speech recognizer 20A and the external recognizer on server 80) may work in parallel. It will be appreciated that performing recognition locally has the advantage of a fast real-time response, but may be limited to the CPU power of the local device and other resources such as battery life. It will be further appreciated that performing the recognition action on server 80 has the advantage of almost unlimited CPU and resources (cloud) but has the disadvantage of suffering slow reaction because of the round-trip delay

[0082] Thus it will be appreciated that a user 5 with no technical skills or background may configure a personalized user interface to override or partially override control software such as operating systems and 3rd party applications. It will be further appreciated, that user 5 may have the ability to do this across different platforms and over different devices and with the ability to share his configurations and also have the ability to use other user configurations.

[0083] Unless specifically stated otherwise, as apparent from the preceding discussions, it is appreciated that, throughout the specification, discussions utilizing terms such as "processing," "computing," "calculating," "determining," or the like, refer to the action and/or processes of a computer, computing system, or similar electronic computing device that manipulates and/or transforms data represented as physical, such as electronic, quantities within the computing system's registers and/or memories into other data similarly represented as physical quantities within the computing system's memories, registers or other such information storage, transmission or display devices.

[0084] Embodiments of the present invention may include apparatus for performing the operations herein. This apparatus may be specially constructed for the desired purposes, or it may comprise a general-purpose computer selectively activated or reconfigured by a computer program stored in the computer. Such a computer program may be stored in a computer readable storage medium, such as, but not limited to, any type of disk, including floppy disks, optical disks, magnetic-optical disks, read-only memories (ROMs), compact disc read-only memories (CD-ROMs), random access memories (RAMs), electrically programmable read-only memories (EPROMs), electrically erasable and programmable read only memories (EEPROMs), magnetic or optical cards, Flash memory, or any other type of media suitable for storing electronic instructions and capable of being coupled to a computer system bus.

[0085] The processes and displays presented herein are not inherently related to any particular computer or other apparatus. Various general-purpose systems may be used with programs in accordance with the teachings herein, or it may prove convenient to construct a more specialized apparatus to perform the desired method. The desired structure for a variety of these systems will appear from the description below. In addition, embodiments of the present invention are not described with reference to any particular programming language. It will be appreciated that a variety of programming languages may be used to implement the teachings of the invention as described herein.

[0086] While certain features of the invention have been illustrated and described herein, many modifications, substitutions, changes, and equivalents will now occur to those of ordinary skill in the art. It is, therefore, to be understood that the appended claims are intended to cover all such modifications and changes as fall within the true spirit of the invention.

CLAIMS

What is claimed is:

1. A system for providing a natural user interface to an application, the system comprising:

a personalized natural user interface to at least partially override a user interface of said application; and

a control element to receive an action from said personalized natural user interface and to provide a control instruction to said application.

2. The system according to claim 1 and comprising an input device to receive natural input and wherein said input device is at least one of a microphone, a motion sensor, an accelerometer, a gyroscope, a magnetometer, a camera, a location sensor and a temperature sensor.

3. The system according to claim 1 and wherein said personalized natural user interface comprises at least one recognizer to recognize at least one natural input as a formatted output.

4. The system according to claim 3 wherein said recognizer is at least one of a speech recognizer, a voice recognizer, a motion recognizer, a location recognizer, a temperature recognizer and a facial expression recognizer.

5. The system according to claim 3 and comprising an external support analyzer to analyze if said recognizing is to be performed on at least one of a local server and a remote server.

6. The system according to claim 1 and wherein said personalized natural user interface comprises a database to store associations of formatted output and associated actions.

7. The system according to claim 6 wherein said associations are one of pre-determined and user generated.

8. The system according to claim 6 and also storing application context and properties of elements of said user interface of said application.

9. The system according to claim 1 and wherein said personalized natural user interface comprises a coordinator to receive said formatted output, to generate at least one action associated with said formatted output from said database and to provide said at least one action to said control element.

10. The system according to claim 1 and wherein said personalized natural user interface comprises a configurer to receive said natural input from a user and associate said natural input with at least one action.

11. The system according to claim 1 and wherein said control element comprises an agent to communicate with said application.

12. The method according to claim 11 and wherein said agent is one of an emulator, a native user interface agent and a web browser add-on.

13. The system according to claim 1 and wherein said control element is a software development kit (SDK).

14. The system according to claim 1 and wherein said application is one of a web-based application and a non-web-based application.

15. A method for providing a natural user interface to an application, the method comprising:

converting natural input into an associated action or chain of actions; and

providing said associated action or chain of actions as input to said application.

14. The method according to claim 15 and wherein said converting comprises receiving said natural input via an input device and wherein said input device is at least one of a microphone, a motion sensor, an accelerometer, a gyroscope, a magnetometer, a camera, a location sensor and a temperature sensor.

15. The method according to claim 13 and wherein said converting comprises recognizing via a recognizer said natural input as a formatted output.

16. The method according to claim 15 and wherein said recognizer is at least one of a speech recognizer, a voice recognizer, a motion recognizer, a location recognizer, a temperature recognizer and a facial expression recognizer.

17. The method according to claim 15 and wherein said converting comprises saving said associated action or chains of actions in a database.

18. The method according to claim 17 wherein said associations are one of: pre-determined and user generated.

19. The method according to claim 15 and wherein said converting comprises generating said action or chain of actions associated with said formatted output.

20. The method according to claim 13 and wherein said converting comprises configuring said natural input to be associated with said action or chain of actions.

21. The method according to claim 13 and wherein said providing comprises communicating with said application via an agent.

22. The method according to claim 13 and wherein said providing comprises communicating with said application via an SDK.

23. The method according to claim 2 1 and wherein said agent is one of: an emulator, a native user interface agent and a web browser add-on.

24. The method according to claim 15 and wherein said application is one of a web-based application and a non-web-based application.

Fig. 1A

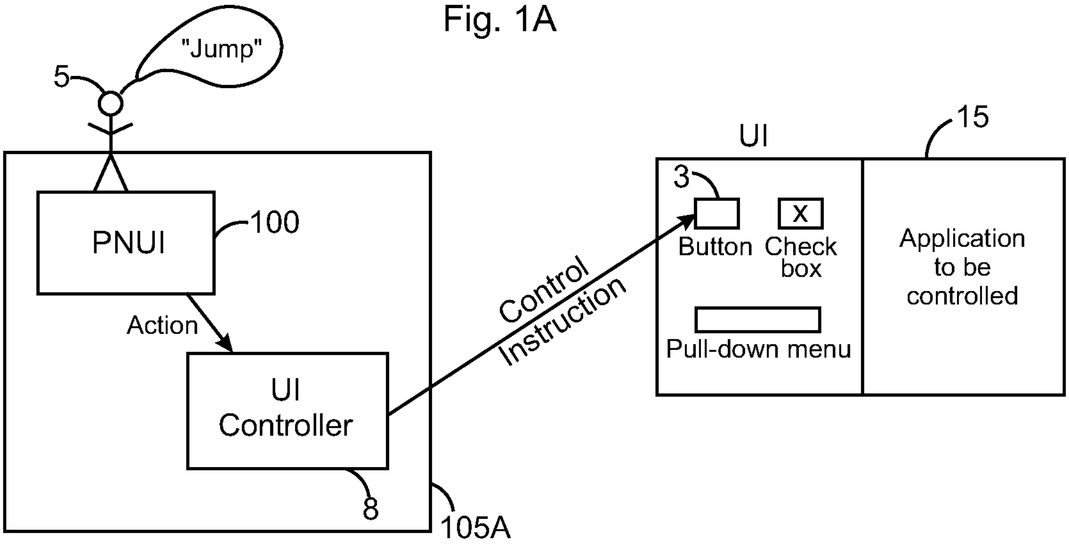
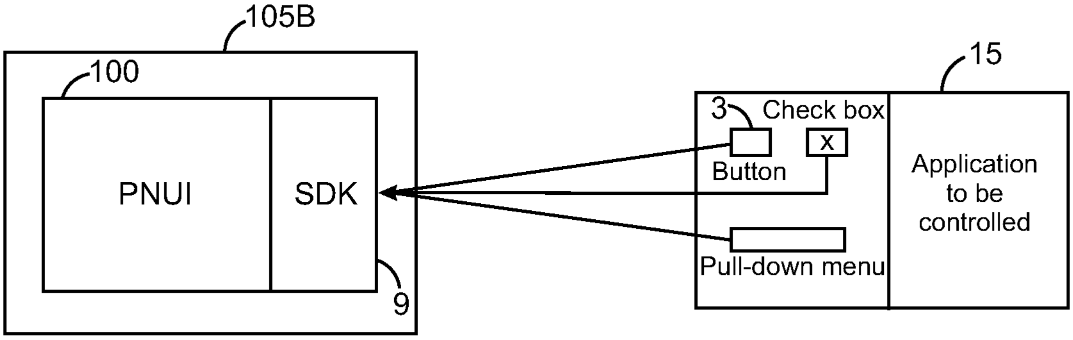

"Jump"
5
PNUI
100
Action
UI
Controller
8
105A
Control
Instruction
UI
15
3
Button
X
Check
box
Pull-down menu
Application
to be
controlled

Fig. 1B


105B
100
PNUI
SDK
9
15
3
Check box
X
Button
Pull-down menu
Application
to be
controlled

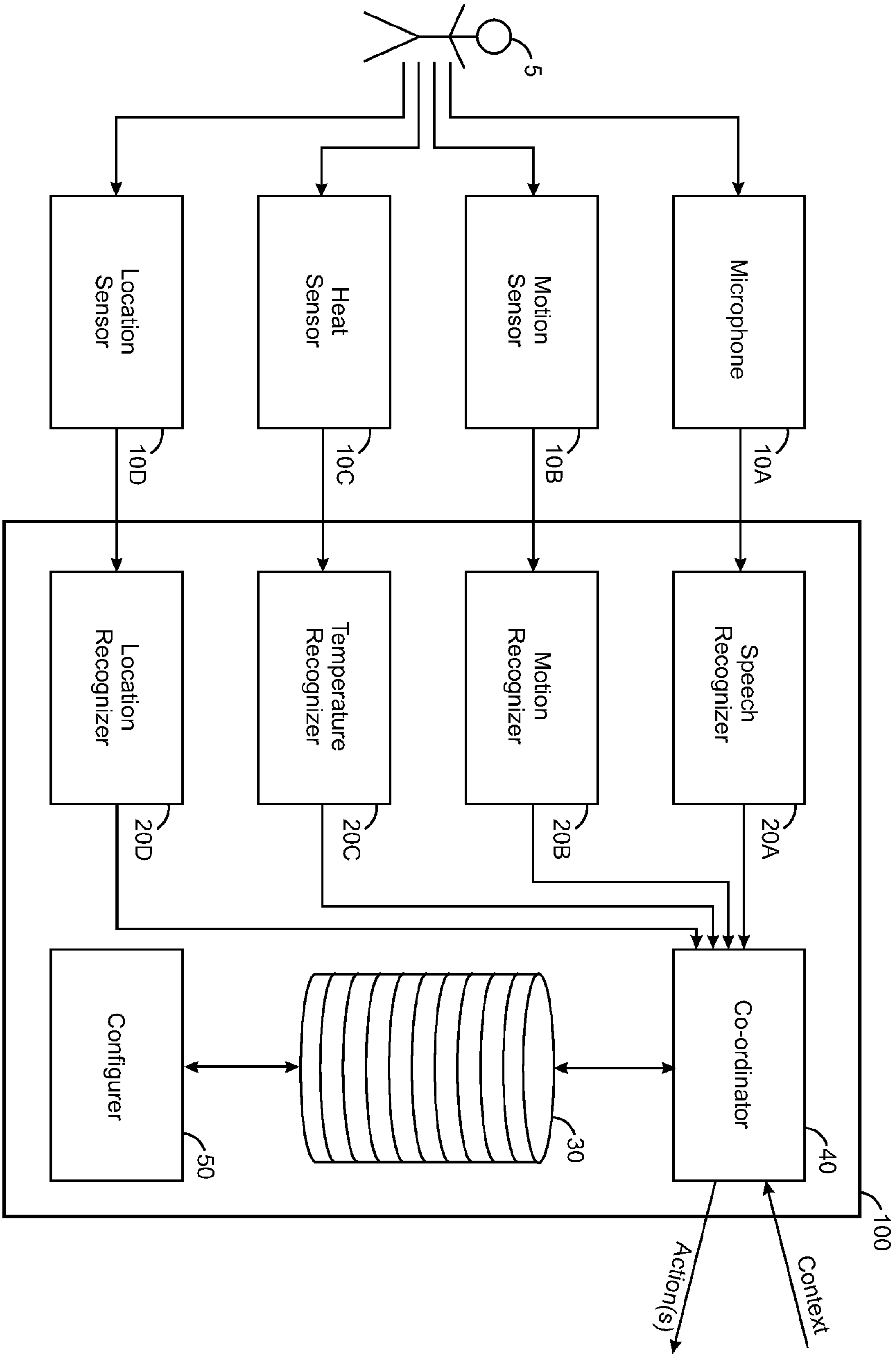

5
Location Sensor
Heat Sensor
Motion Sensor
Microphone
10D
10C
10B
10A
Location Recognizer
Temperature Recognizer
Motion Recognizer
Speech Recognizer
20D
20C
20B
20A
Configurer
50
30
Co-ordinator
40
100
Action(s)
Context

Fig. 2

Fig. 3

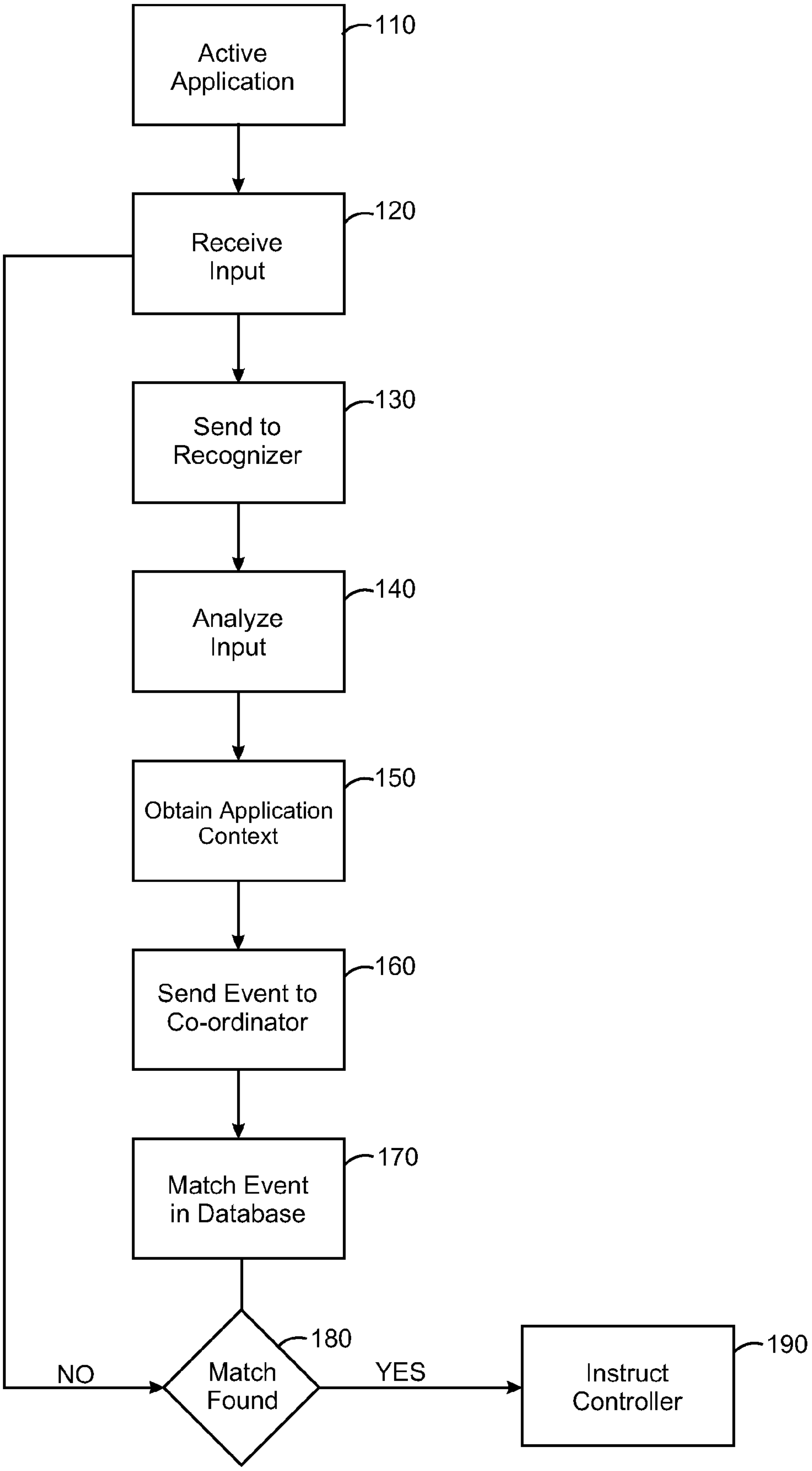

Active Application
110
Receive Input
120
Send to Recognizer
130
Analyze Input
140
Obtain Application Context
150
Send Event to Co-ordinator
160
Match Event in Database
170
Match Found
180
NO
YES
Instruct Controller
190

Fig.4

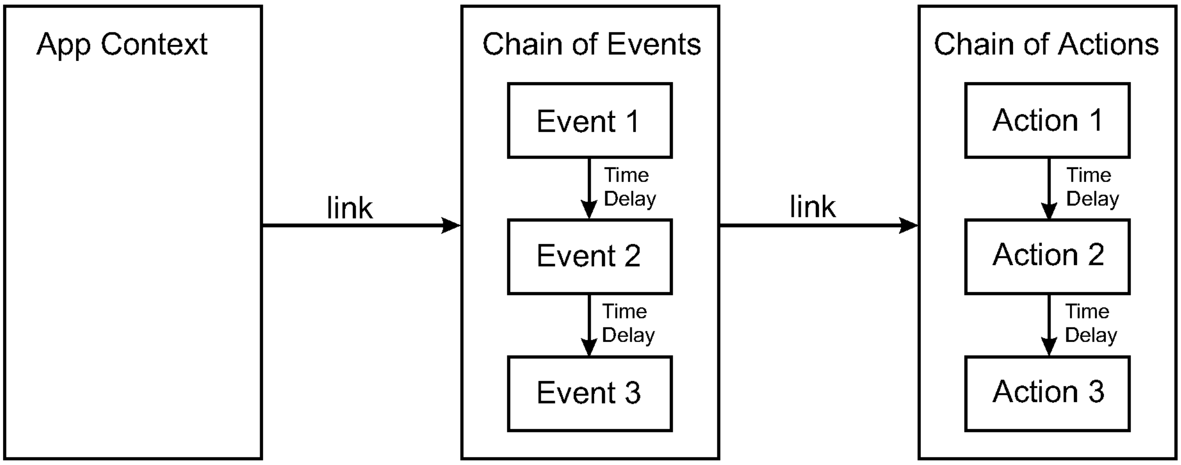

App Context
link
Chain of Events
Event 1
Time Delay
Event 2
Time Delay
Event 3
link
Chain of Actions
Action 1
Time Delay
Action 2
Time Delay
Action 3

Fig. 5

Event	Action	Context
speech "ON"	turn on application	global
arm movement to left	move character left	screen 5 of game
speech "BYE", wave motion	turn off app.	global
thumbs up motion	like	appropriate screen of facebook
speech "call 12345"	call no.12345	global
speech "call Brian"	call no.12345	global

Fig. 6

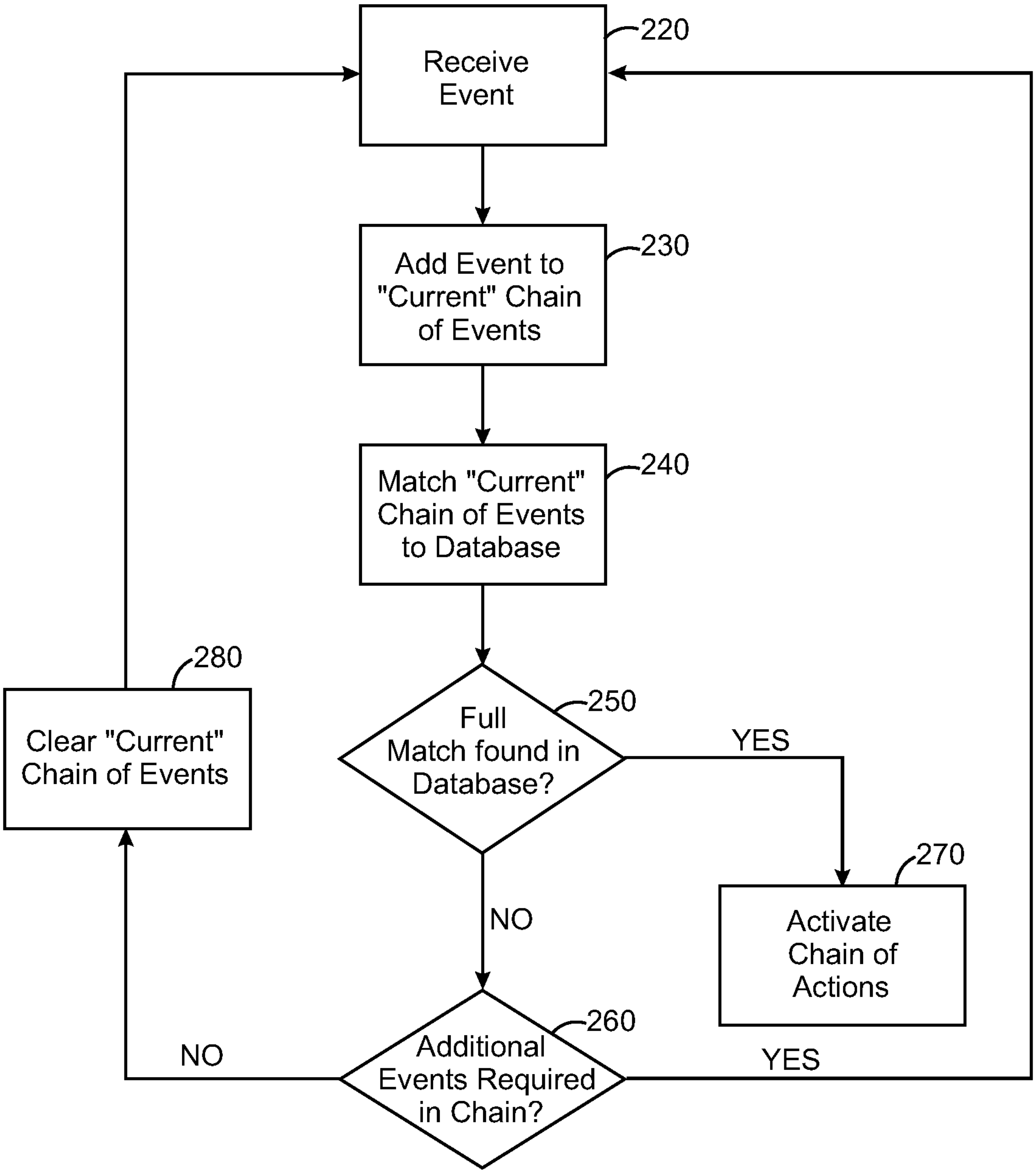

Receive Event
220
Add Event to "Current" Chain of Events
230
Match "Current" Chain of Events to Database
240
Full Match found in Database?
250
YES
Activate Chain of Actions
270
NO
Additional Events Required in Chain?
260
YES
NO
Clear "Current" Chain of Events
280

Fig. 7

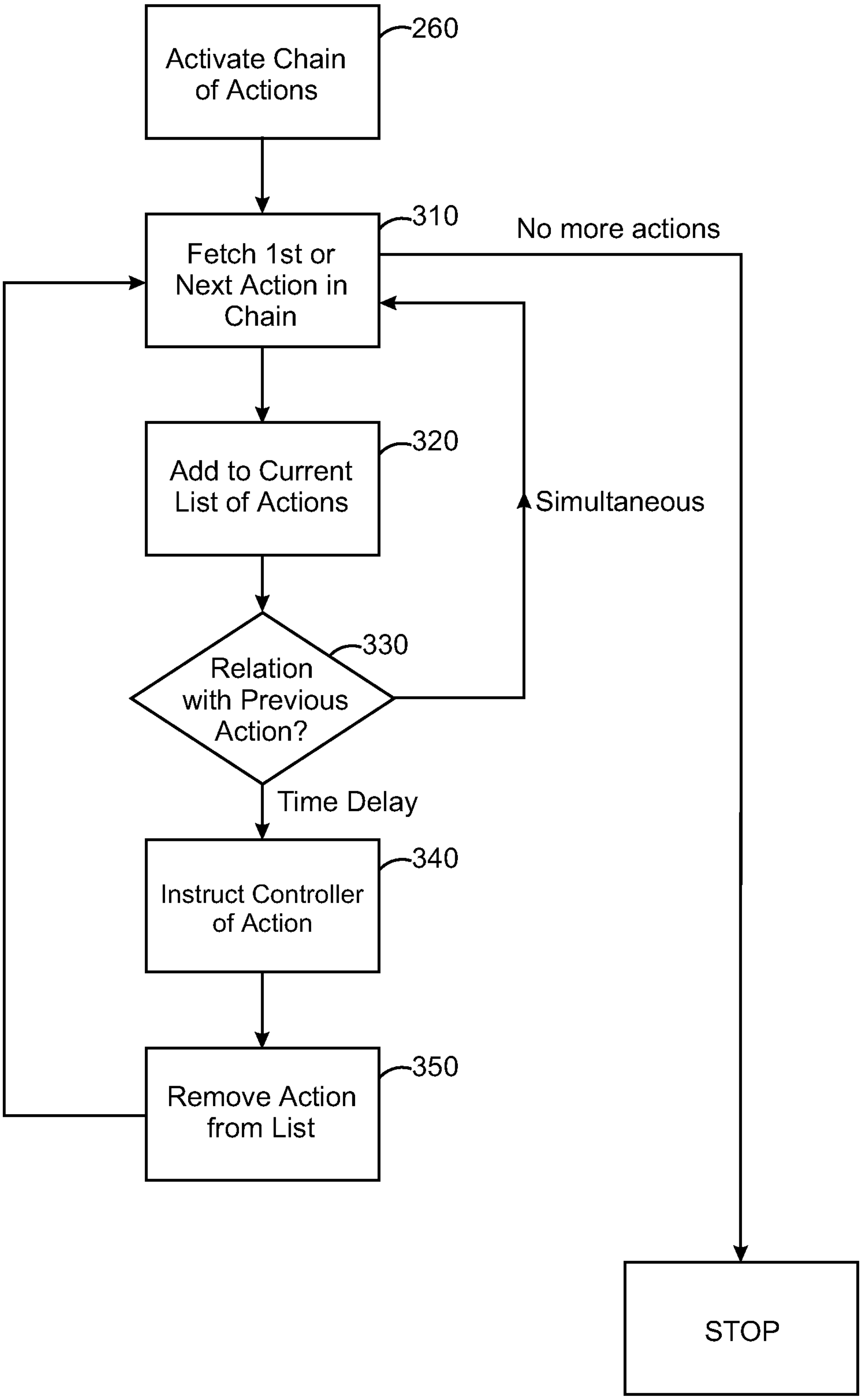

Activate Chain of Actions
260
Fetch 1st or Next Action in Chain
310
No more actions
Add to Current List of Actions
320
Simultaneous
Relation with Previous Action?
330
Time Delay
Instruct Controller of Action
340
Remove Action from List
350
STOP

Fig. 8

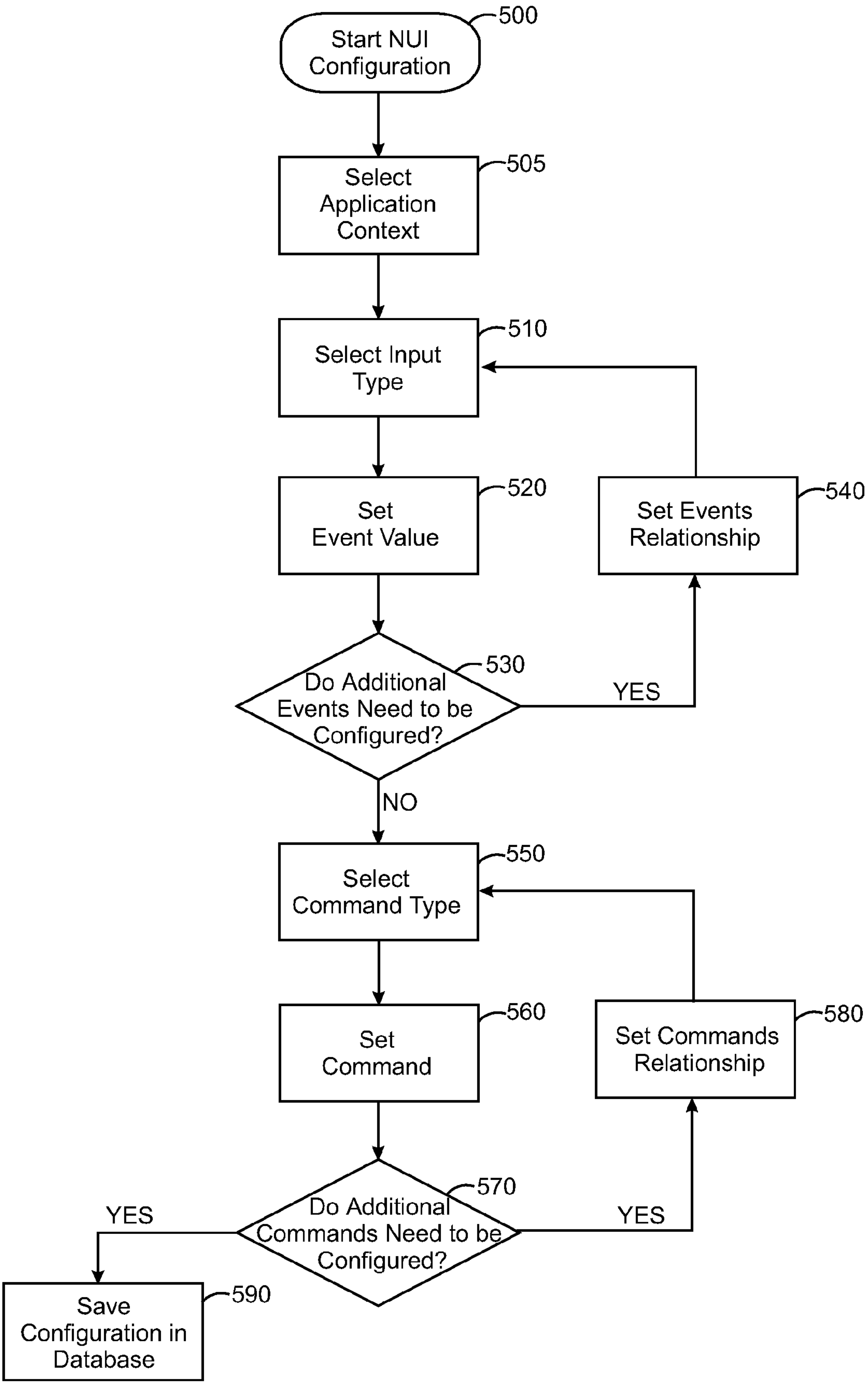

Start NUI Configuration
500
Select Application Context
505
Select Input Type
510
Set Event Value
520
Set Events Relationship
540
Do Additional Events Need to be Configured?
530
YES
NO
Select Command Type
550
Set Command
560
Set Commands Relationship
580
Do Additional Commands Need to be Configured?
570
YES
YES
Save Configuration in Database
590

Fig. 9

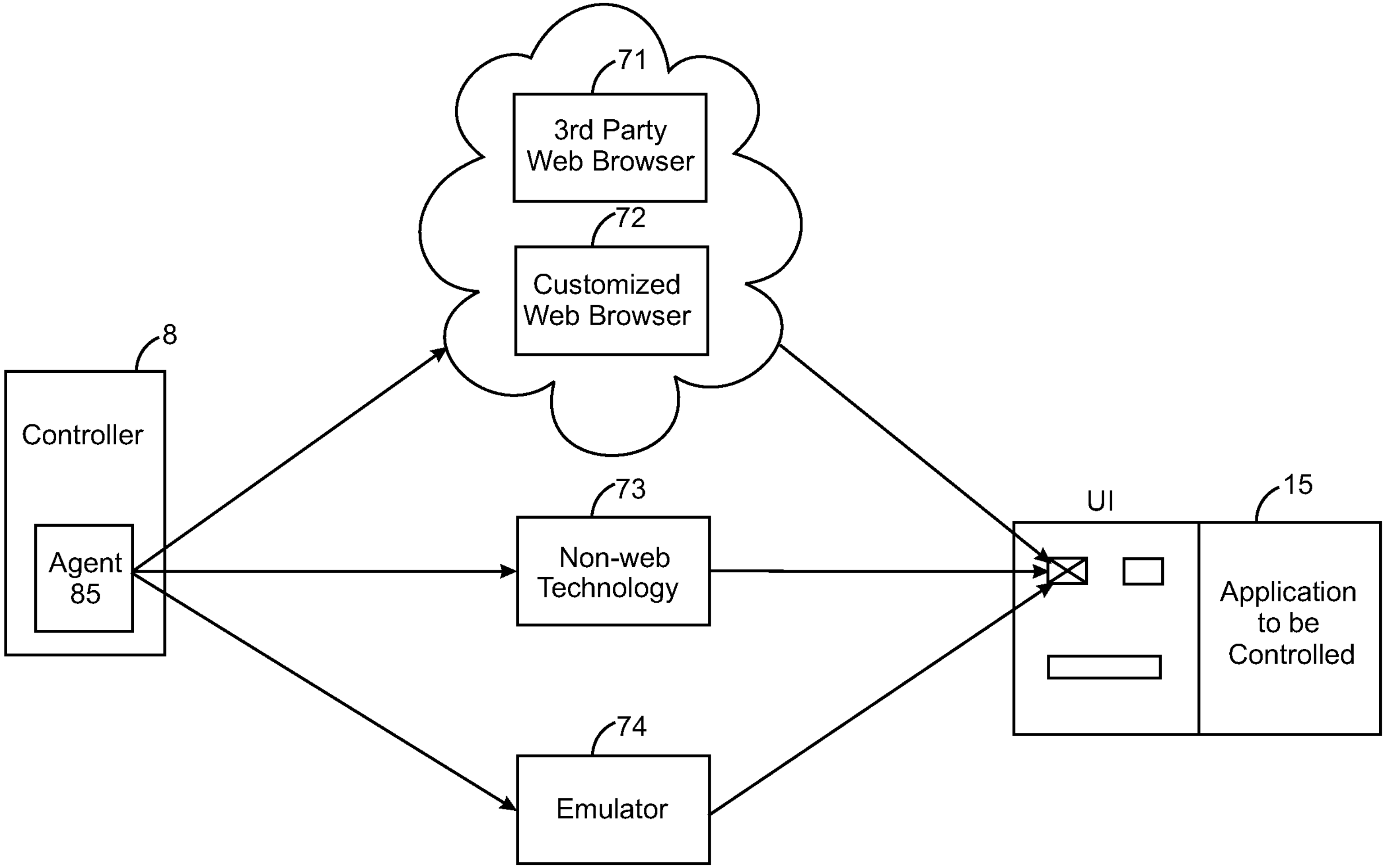

Controller
Agent
85
8
3rd Party
Web Browser
71
Customized
Web Browser
72
Non-web
Technology
73
Emulator
74
UI
Application
to be
Controlled
15

Fig. 10

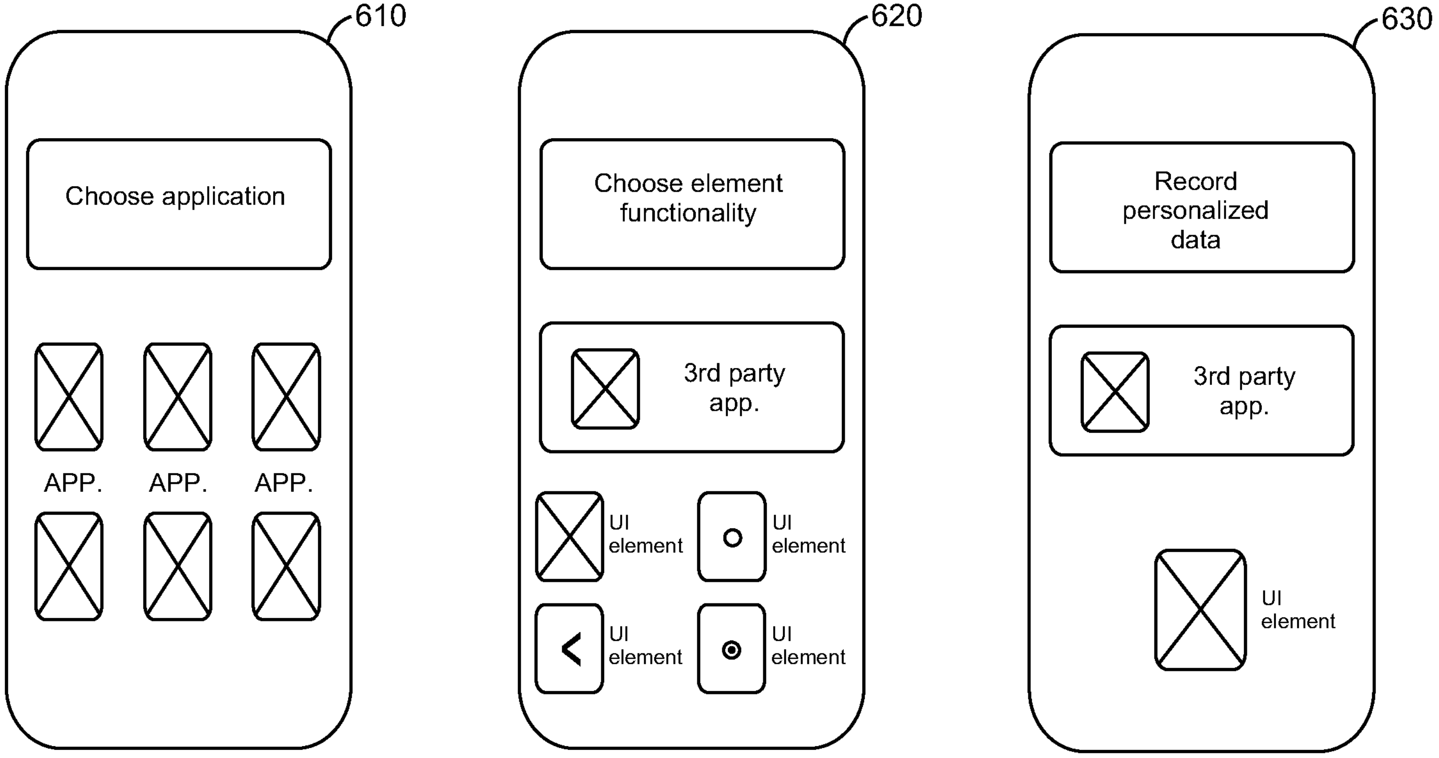

610
Choose application
APP.
APP.
APP.
620
Choose element functionality
3rd party app.
UI element
UI element
UI element
UI element
630
Record personalized data
3rd party app.
UI element

Fig. 11

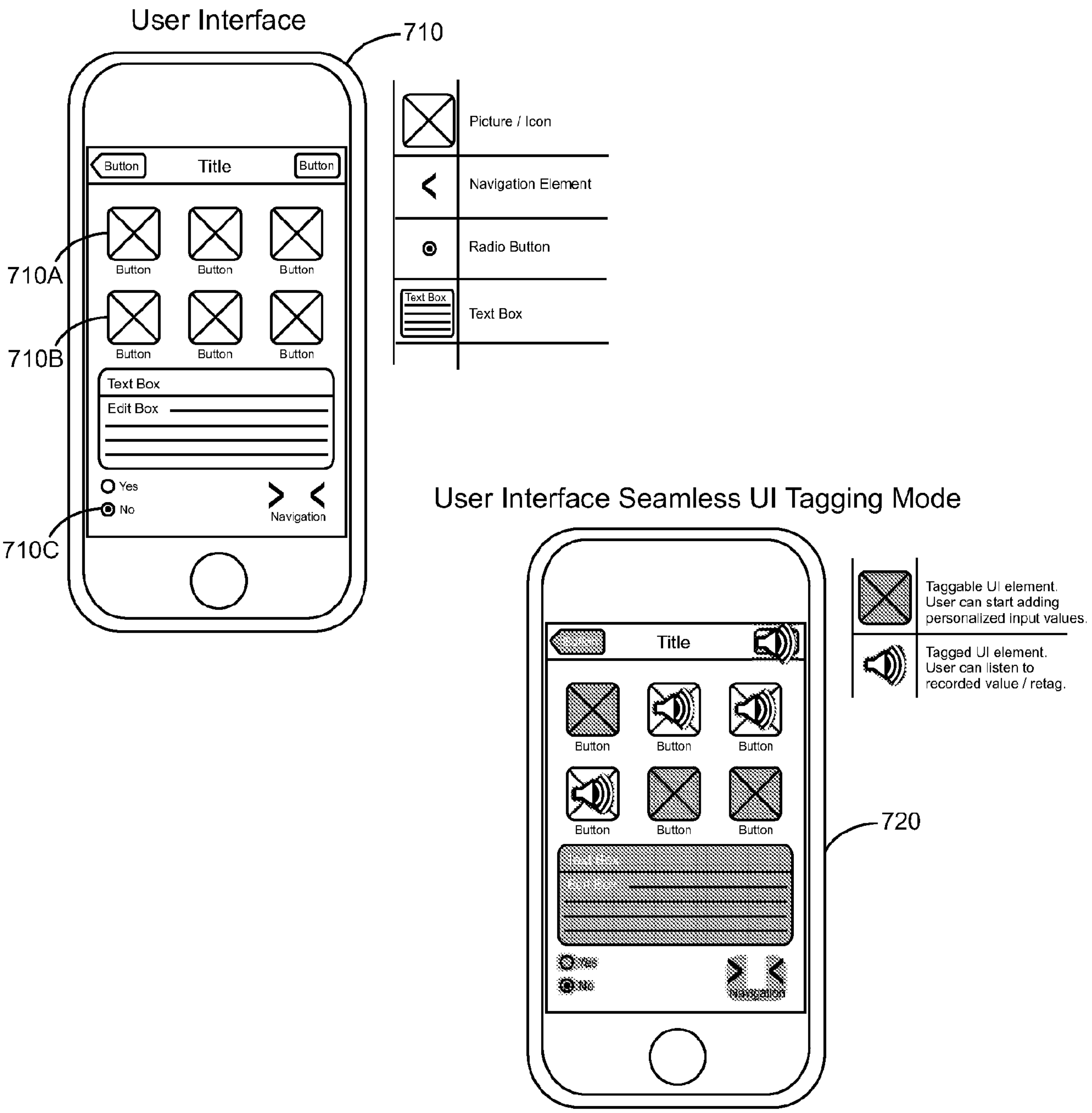

User Interface
710
Button
Title
Button
710A
Button
Button
Button
710B
Button
Button
Button
Text Box
Edit Box
Yes
No
Navigation
710C
Picture / Icon
Navigation Element
Radio Button
Text Box
Text Box
User Interface Seamless UI Tagging Mode
Title
Button
Button
Button
Button
Button
Button
720
Text Box
Edit Box
Yes
No
Navigation
Taggable UI element. User can start adding personalized input values.
Tagged UI element. User can listen to recorded value / retag.

Fig. 12

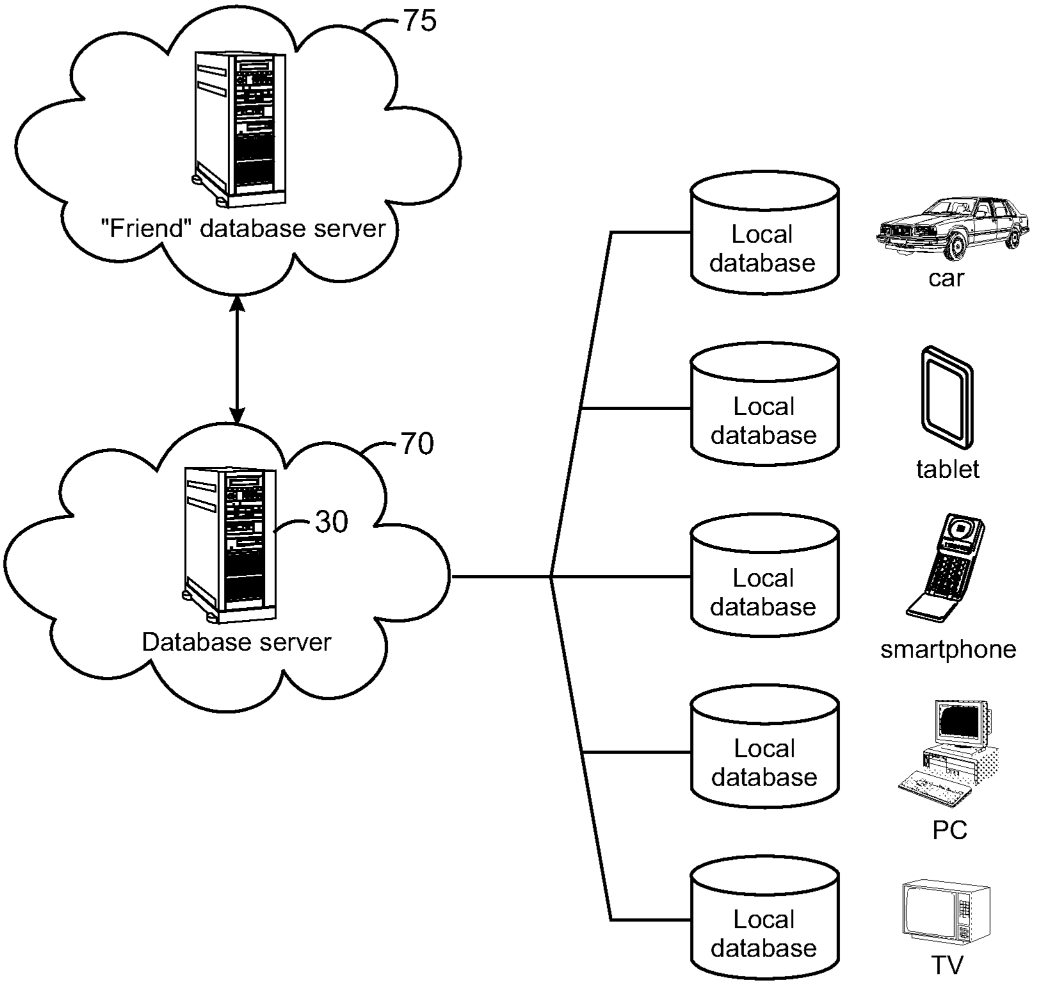
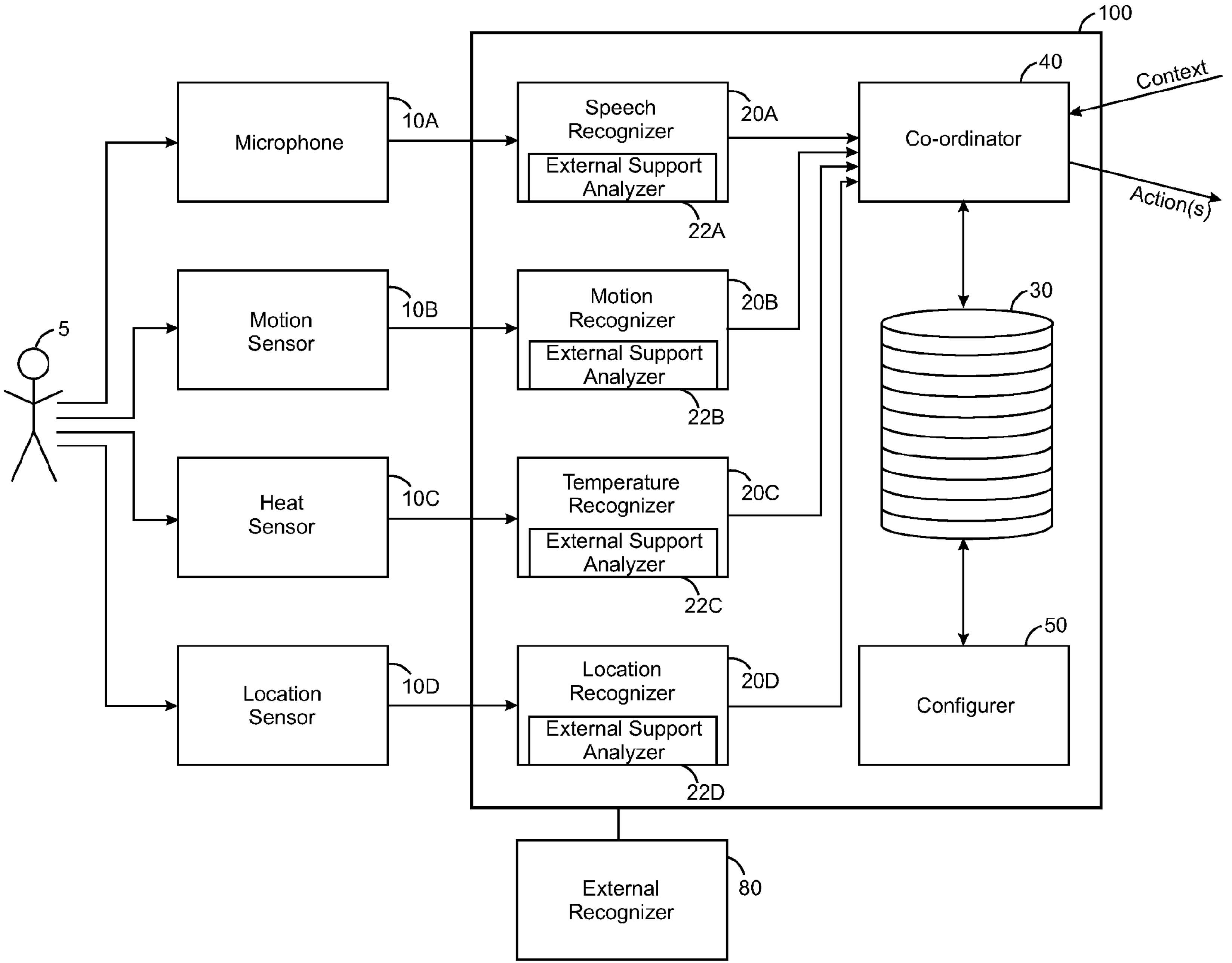

75
"Friend" database server
70
30
Database server
Local database
car
Local database
tablet
Local database
smartphone
Local database
PC
Local database
TV

Fig. 13


100
5
Microphone
10A
Motion Sensor
10B
Heat Sensor
10C
Location Sensor
10D
Speech Recognizer
External Support Analyzer
20A
22A
Motion Recognizer
External Support Analyzer
20B
22B
Temperature Recognizer
External Support Analyzer
20C
22C
Location Recognizer
External Support Analyzer
20D
22D
Co-ordinator
40
Context
Action(s)
30
Configurer
50
External Recognizer
80

INTERNATIONAL SEARCH REPORT

International application No.

PCT/IB20 13/050123

A. CLASSIFICATION OF SUBJECT MATTER

IPC (2013.01) GIOL 15/18, GIOL 15/28, G06F 3/048, G06F 3/01, G06F 3/00

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed **by** classification symbols)

IPC (2013.01) GIOL, G06F

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

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

Databases consulted: THOMSON INNOVATION, Esp@cenet, Google Patents

Search terms used: natural user interface personal input action instruct application voice movement eye override overwrite replace

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2006 13622 1 A1 JAMES et al. 22 Jun 2006 (2006/06/22) Paragraphs 0031-0059	1-12,14-21,23,24
X	WO 001 1571 Al BCL COMPUTERS INC [US] 02 Mar 2000 (2000/03/02) Abstract, page 4, line 9-page 5, line 24; page 6, lines 1-25; page 7, lines 1-9; page 9, lines 11-20; page 10, lines 3-12; page 11, lines 4-42; page 17, lines 8-14; page 24, lines 4-17; page 26, line 13-page 29, line 6; page 34, lines 10-20	1-9,1 1-19,21-24
X	US 2004249640 Al GRANT et al. 09 Dec 2004 (2004/12/09) Abstract, paragraphs 0020-0022,0036,0044-0052	1-9,1 1,14-19,21,24
A	WO 20041 14034 A2 SCHNEIDER AUTOMATION INC. 29 Dec 2004 (2004/12/29) The whole document	1-24

□ Further documents are listed in the continuation of Box C. ☒ See patent family annex.

* Special categories of cited documents;

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" **earlier application or patent but published on** or after the international tiling date

"L" **document which** may throw doubts on **priority** claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international tiling date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of die same patent family

Date of the actual completion of the international search 28 May 2013	Date of mailing of the international search report 02 Jun 2013
Name and mailing address of the ISA: Israel Patent Office Technology Park, Bldg.5, Malcha, Jerusalem, 9695101, Israel Facsimile No. 972-2-5651616	Authorized officer PLACHINTA Ekaterina Telephone No. 972-2-5651740

INTERNATIONAL SEARCH REPORT
Infortnatiori on patent family members

International application No.
PCT/IB2013/050123

Patent document cited search report	Publication date	Patent family member(s)	Publication Date
US 2006136221 Al	22 Jun 2006	US 2006136221 Al	22 Jun 2006
WO 001 1571 Al	02 Mar 2000	CN 1313972 A	19 Sep 2001
		EP 1116134 Al	18 Jul 2001
		JP 2002523828 A	30 Jul 2002
		WO 001 1571 Al	02 Mar 2000
US 2004249640 Al	09 Dec 2004	us 6208972 Bl	27 Mar 2001
		us 2002065654 Al	30 May 2002
		us 6606599 B2	12 Aug 2003
		us 2004249640 Al	09 Dec 2004
		us 7188067 B2	06 Mar 2007
		us 7426469 Bl	16 Sep 2008
		us 743051 1 Bl	30 Sep 2008
		us 7433823 Bl	07 Oct 2008
		us 7447637 Bl	04 Nov 2008
		us 7447638 Bl	04 Nov 2008
		us 2009024391 Al	22 Jan 2009
		us 7827035 B2	02 Nov 2010
		us 201 1029316 Al	03 Feb 201 1
		us 8000974 B2	16 Aug 201 1
		us 2012029921 Al	02 Feb 2012
		us 8175883 B2	08 May 2012
		us 2012259641 Al	11 Oct 2012
		us 8340970 B2	25 Dec 2012
		us 2013013319 Al	10 Jan 2013
		WO 02073603 Al	19 Sep 2002
WO 20041 14034 A2	29 Dec 2004	US 2005213036 Al	29 Sep 2005
		WO 20041 14034 A2	29 Dec 2004
		WO 20041 14034 A3	26 May 2005

INTERNATIONAL SEARCH REPORT
Infortnatiori on patent family members

International application No.

PCT/IB2013/050123

Patent document cited search report	Publication date	Patent family member(s)	Publication Date
------------------------------------	------------------------------------	------------------------------------	------------------------------------