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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
27 November 2014 (27.11.2014)

WIPO | PCT

(10) International Publication Number
WO 2014/189868 A1

(51) International Patent Classification:
G06F 3/16 (2006.01) *G06F 3/01* (2006.01)
G06F 21/32 (2013.01) *G07F 17/32* (2006.01)
A63F 13/20 (2014.01) *G06K 9/00* (2006.01)

(21) International Application Number:
PCT/US2014/038683

(22) International Filing Date:
20 May 2014 (20.05.2014)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
13/897,466 20 May 2013 (20.05.2013) US

(71) Applicant: **MICROSOFT CORPORATION** [US/US]; One Microsoft Way, Redmond, Washington 98052-6399 (US).

(72) Inventors: **SMITH, Robert**; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). **MILLER, Li-Chen**; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). **WYMAN, Joseph**; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). **GARCIA, Jonathan**; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). **HALVORSEN, Pat**; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US). **GILES, Jason**; c/o Microsoft Corporation, LCA - International Patents, One Microsoft Way, Redmond, Washington 98052-6399 (US).

(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, IR, IS, JP, KE, KG, 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, SA, 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 of regional protection available)*: ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ,

[Continued on next page]

(54) Title: ATTRIBUTING USER ACTION BASED ON BIOMETRIC IDENTITY

(57) Abstract: Example apparatus and methods concern attributing user actions based on biometric identifications. One example method includes detecting an action intended to control an application associated with a shared launch surface. The method includes associating the action with a body located in a field of view of a biometric sensor associated with a shared launch surface. Data from the biometric sensor is used to determine a biometric identity for the body and then the action is attributed to a user as a function of the biometric identity. A context associated with the user is accessed and an operation associated with the action is selectively controlled based on the context.

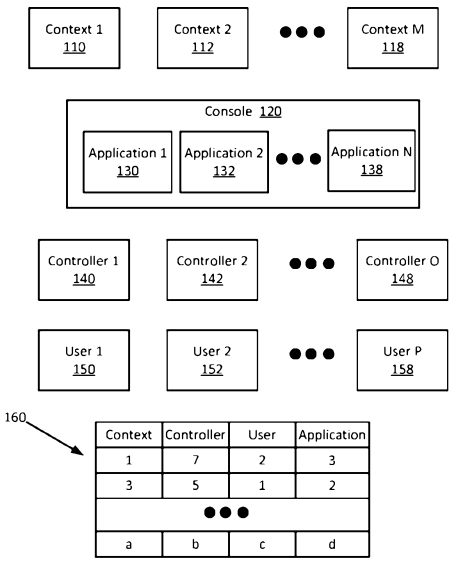

FIG. 1

WO 2014/189868 A1

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, KM, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii))*

— *as to the applicant's entitlement to claim the priority of the earlier application (Rule 4.17(iii))*

Published:

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

ATTRIBUTING USER ACTION BASED ON BIOMETRIC IDENTITY

BACKGROUND

[0001] Conventionally, game systems attribute user actions based on a one-to-one mapping between a signed-in user and a controller. This may limit the ability of a gamer to interact with a game console simultaneously in a variety of ways, to switch controllers, to move about in a game space, and in other ways. This conventional attribution approach arose because game consoles typically had a concept of a controller and a concept of a user and the only way to connect the user to the controller was through a sign-in procedure. The console may have established and maintained a fixed, one-to-one relationship between a controller and a player. All actions performed from a controller were associated with the user who entered their information from that controller and the only actions the user could perform were those associated with the controller on which the user signed in. In this one-to-one user-to-controller configuration, one user was identified as controlling the console. This control was manifested at the system level by having system level actions (e.g., application launches) associated with the user who controlled the console.

[0002] Computer systems, including game systems, may allow multiple users to access, sign in, or otherwise interact with the system at the same time. For example, a single game console may have dedicated controllers and may have the ability to allow additional transient controllers to interact with the game. Thus, multiple players may simultaneously be able to sign into and interact with a multi-user game. Additionally, the game console may allow different players to have different games active at different times and even to come and go from a game. The traditional, fixed, one-to-one relationship between a user and a controller may limit the game experience. This limiting experience may extend to other non-gaming systems.

[0003] Different users may have different contexts. A user context may describe, for example, user attributes and user state. User attributes may include, for example, name, language preferences, login credentials for applications, login credentials for websites, saved content including documents or game saves, or other data. User state may include, for example, location, or other data. When there are multiple users signed into a system, there are multiple contexts available to the system. Conventionally, when a user launched an application, the context associated with the application may have controlled, at least in part, the operation of the application. For example, a first user context may have caused an

application to produce an English language presentation for a male who is an intermediate level swordsman while a second user context may have caused an application to produce a French language presentation for a female who is an expert archer. The context may have controlled in-game attributes (e.g., point of view, character) but may also have controlled attributes of other applications launched from the launch surface. A launch surface refers to an interface with which a user may interact to launch an application. A launch surface may be, for example, a desktop, a start menu, a dashboard, or other interactive item from which applications or processes can be initiated (e.g., launched). Conventionally, given the one-to-one user-to-controller relationship and the single launch surface owner approach, it may have been difficult, if even possible at all, to change the user context associated with an action.

SUMMARY

[0004] This Summary is provided to introduce, in a simplified form, a selection of concepts that are further described below in the Detailed Description. This Summary is not intended to identify key features or essential features of the claimed subject matter, nor is it intended to be used to limit the scope of the claimed subject matter.

[0005] Example apparatus and methods detect actions intended to control an application. The application may have been launched from a shared launch surface. The shared launch surface may be provided by a game console, a computer, an operating system, or other device or system. Example apparatus and methods associate the action with a body located in a field of detection of a biometric sensor associated with the shared launch surface. For example, a game console may have an infrared (IR) system that produces a depth map for a game space, a visible light camera system that acquires images from the game space, a sound and voice recognition system for acquiring voice data, or haptic interfaces for capturing fine movements. Example apparatus and methods determine, using data provided by the biometric sensor(s), a biometric identity for the body and then attribute (e.g., associate, couple, map) the action to a user as a function of the biometric identity. Once the biometric identity is available, a context associated with the user can be accessed and the application or shared launch surface can be controlled or manipulated as a function of the action and the context. The context may include, for example, information that describes the user in general (e.g., user attributes) and dynamic information that describes the user at a particular point in time (e.g., user state).

[0006] Example apparatus may be configured with hardware including a processor and a biometric sensor(s). The example apparatus may include a memory configured to store

information concerning ownership of a shared launch surface provided by the apparatus or an operating system running on the apparatus. The apparatus may include logics that are configured to attribute a user action as a function of a biometric identity determined by data provided by the biometric sensor. The apparatus may track bodies located in a field of detection (e.g., field of view) of the biometric sensor. The apparatus may also identify actions performed in the field of detection. An action can be mapped to a body, and a body can be mapped to a user through a biometric identity. Once the action and the biometric identification of a user are identified, the apparatus may selectively control an operation of the apparatus as a function of the biometric identification.

BRIEF DESCRIPTION OF THE DRAWINGS

[0007] The accompanying drawings illustrate various example apparatus, methods, and other embodiments described herein. It will be appreciated that the illustrated element boundaries (e.g., boxes, groups of boxes, or other shapes) in the figures represent one example of the boundaries. In some examples, one element may be designed as multiple elements or multiple elements may be designed as one element. In some examples, an element shown as an internal component of another element may be implemented as an external component and vice versa. Furthermore, elements may not be drawn to scale.

[0008] Figure 1 illustrates an example game environment with example mappings.

[0009] Figure 2 illustrates an example method associated with attributing a user action based on a biometric identity.

[0010] Figure 3 illustrates a portion of an example method associated with attributing a user action based on a biometric identity.

[0011] Figure 4 illustrates an example game environment and different user actions.

[0012] Figure 5 illustrates an example apparatus configured to facilitate attributing a user action based on a biometric identity.

[0013] Figure 6 illustrates an example apparatus configured to facilitate attributing a user action based on a biometric identity.

[0014] Figure 7 illustrates an example cloud operating environment.

[0015] Figure 8 is a system diagram depicting an exemplary mobile communication device configured to facilitate attributing a user action based on a biometric identity.

DETAILED DESCRIPTION

[0016] Example apparatus and methods decouple the fixed 1:1 relationship between a launch surface, a context, a user, and a controller. Rather than fix the user/controller relationship and the controller/context relationship, example apparatus and methods detect

actions, detect and identify bodies, flexibly and dynamically determine an action/user relationship and a user/context relationship, and then associate the action with the appropriate context. Since users may employ different controllers at different times during a game session, identifying a user currently associated with a controller provides new flexibility. A controller may include, for example, a gamepad, a remote, a game guitar, a voice control apparatus, a gesture capture apparatus, haptic interfaces, natural user interfaces, or other apparatus. Additionally, since users may interact (e.g., initiate operations) with an application, launch surface, or system using different actions (e.g., button press, virtual button press, user interface interaction, voice command, gesture) at different times, identifying an action, and then identifying a user associated with the action, and then identifying a context to associate with the action, also provides new flexibility. The flexibility is extended even further by mapping the action to user to context at the time of the action.

[0017] Consider the following scenario: three gamers are signed in to a video game and are playing a collaborative search, reward, and story line game. A first user may have launched the game and may currently be in control of or otherwise associated with the launch surface. As the game progresses, one player leaves and another player joins, one player changes from a wired controller to a wireless controller, and another player changes from one wired controller to another wired controller. Conventionally, these controller changes may have caused the game to be halted or even exited so that the conventional one-to-one mappings could be brought up-to-date. Over time, a player other than the first player may acquire enough points to be eligible for a new game piece that may be available in an online auction. Conventionally, if the player wanted to launch the online auction application so they could acquire the item, the game would, once again, have to be suspended, perhaps even exited, while the player navigated to the launch surface, acquired the launch surface, and then launched the application. Once an item was acquired, the first user might then have to navigate back to the launch surface, re-acquire the launch surface, and re-launch the game. These are sub-optimal experiences for gamers. Similar sub-optimal experiences may be encountered by users of other non-gaming applications.

[0018] Conventionally, game systems have allowed multiple users but have had a fixed 1:1 relationship during a session between a controller and a player. This frustrates gamers because picking up the wrong controller, losing controller battery power during a game, or changing controllers during a game may produce undesirable results. For example, a gamer might have to end the game, reconfigure the game, and start over. Gamers really

don't like doing this. Gamers want a more flexible experience, especially in games that may last a long time and where players can come and go during the game. Additionally, as more sophisticated wireless controllers become available, the ability to switch between controllers during a game may provide the gamer with a richer experience.

[0019] Example apparatus and methods break the fixed one-to-one relationship by making biometric identifications of users. Instead of using the biometrics for conventional actions (e.g., authentication, security), a biometric identity is used to map the person associated with an action to a context. As used herein, a biometric identity may include a fingerprint, a retinal pattern, or other unique physical attribute. A biometric identity may also include, for example, a user identified attribute (e.g., I am wearing the red shirt, I am wearing the white hat, I am the tallest person). A biometric identity may also include, for example, a system identified attribute (e.g., the tallest player, the roundest player, the player with the deepest voice, the player with the darkest colored shirt/hat, the player with the lightest hat/hair). Thus, biometric identity is intended to be used in a broad manner. Rather than performing a single biometric identification for security purposes, on-going real time or near real time biometric identifications may be made for bodies visible in the field of view of a game system for action attribution reasons. A gamer may move around during a game session. As the person moves around they might use different controllers or might initiate actions using different approaches. For example, at a first time a user might be on the left side of the game space and might use controller A to initiate an action (e.g., swing virtual golf club). At a second time, the user might be in the middle of the game space and might initiate an action (e.g., move to next hole) using a gesture (e.g., flick). At a third time, the user might be in the left of the game space and might initiate an action (e.g., check current stock prices) using a voice command. Example apparatus and methods may track the position of the user in real-time or near real-time and then match an action location to the user location to facilitate attributing the action.

[0020] Checking stock prices may require launching a separate application in, for example, a small popup that might be displayed just to the requester and that would not interrupt the game, at least for other players. The console may first detect the action and then identify the user associated with the action through biometrics. Example apparatus and methods may then consult, create, update or otherwise manipulate a current mapping to associate a context with the user that made the action. Once the context associated with the user has been identified, the action can be taken in light of the proper context. This can

be done without navigating to the launch surface or reconfiguring the launch surface. This provides a seamless experience that facilitates an improved game experience.

[0021] Example apparatus and methods use the identity of the biometrically identified user and, in some cases, the location of the action, to map a context to an operation (e.g., launch application, manipulate application, use application), without disrupting the game in progress. For example, while the game is in progress the gamer might receive a text that a new video by their favorite artist is available. The gamer may want to download, acquire, or view the new video without having to exit the game. Conventionally, the user that initiated the game owns the launch surface and any application launched during the session will be associated with the launch surface owner. To switch the launch surface ownership to another person typically requires suspending/ending the game, which is unacceptable to many gamers. Conventionally, the gamer might not be able to simply pause the game, or even keep the game in progress, and launch a video acquisition application at the same time. With a flexible, dynamic biometric identity approach like that described herein, the gamer may make an application initiating action and acquire or view the video without ever leaving the game. Additionally, the video would be acquired by the user who initiated the action, not by the user who "owns" the launch surface. Thus, billing updates, data plan updates, or other user specific results would be associated with the gamer who actually accessed the video, not with the gamer who initially started the game. While games are described, other non-gaming applications may also use the flexible, dynamic, biometric identity based action attribution described herein.

[0022] Example apparatus and methods recognize that a multi-user system may have multiple users that may want to launch or otherwise interact with applications. Example apparatus and methods also recognize that users may have contexts that they want associated with an application or an action taken in an application. Thus, rather than tightly coupling a launch surface to a single user and to a single context that depends solely on a one-to-one relationship established between a user and a controller, example apparatus and methods may map users to contexts and contexts to actions with a flexible decoupled approach that attributes user actions based on biometric identities made at the time an action is taken.

[0023] Since users may come and go or may enter and leave a game space, example apparatus and methods may track bodies in the game space in real time or near real time, may biometrically identify a tracked body as a known user, and may associate a context with an action taken by that body based on the biometric identity. A biometric identity can

be determined in different ways. Biometric identification may include, for example, facial recognition, voice recognition, fingerprint recognition, gesture recognition, haptic recognition, or other approaches.

[0024] Figure 1 illustrates an example game environment with example mappings. A game console 120 may be able to run a number of applications (e.g., application1 130, application2 132, ... applicationN 138). Console 120 may be able to support or interact with a number of controllers (e.g., controller1 140, controller2 142, ... controllerO 148). Since console 120 is a multi-user system, console 120 may be able to support a number of users (e.g., user1 150, user2, 152, ... userP 158). Since the console 120 may support multiple users, console 120 may have access to multiple contexts (e.g., context1 110, context2 112, ... contextM 118). Conventionally, for any instance of any application, a specific controller would have a fixed one-to-one relationship with a specific user. The fixed one-to-one relationship would determine the context. Unfortunately, this fixed one-to-one relationship produced burdensome limitations for game play. For example, if user1 150 signed in with controller1 140, then user1 150 could only use controller1 140 for the duration of the game.

[0025] Example apparatus and methods break this fixed one-to-one relationship by tracking users during a game session and then mapping a user to an action at the time of the action. Thus, user1 150 might start playing a game while seated using controller1 140, then after a while user1 150 might put down controller1 140, move from the comfortable chair to a standing position and start interacting with the game using gestures rather than the controller. At a later time, user1 150 might lie down and start controlling the game using voice commands. Near the end of the game, user1 150 might pick up controller 148 and control the game with button presses from controller 148. This scenario would be impossible in traditional games. However, by using mappings 160, example apparatus and methods may provide this level of flexibility. Mappings 160 show a four way mapping that relates a context to a controller and a user and an application. Different mappings may involve a greater or lesser number of attributes. In one embodiment, mappings 160 may be updated in real time or near real time to reflect the current reality in the game space.

[0026] Some portions of the detailed descriptions that follow are presented in terms of algorithms and symbolic representations of operations on data bits within a memory. These algorithmic descriptions and representations are used by those skilled in the art to convey the substance of their work to others. An algorithm is considered to be a sequence of operations that produce a result. The operations may include creating and manipulating

physical quantities that may take the form of electronic values. Creating or manipulating a physical quantity in the form of an electronic value produces a concrete, tangible, useful, real-world result.

[0027] It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, and other terms. It should be borne in mind, however, that these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise, it is appreciated that throughout the description, terms including processing, computing, and determining, refer to actions and processes of a computer system, logic, processor, system-on-a-chip (SoC), or similar electronic device that manipulates and transforms data represented as physical quantities (e.g., electronic values).

[0028] Example methods may be better appreciated with reference to flow diagrams. For simplicity, the illustrated methodologies are shown and described as a series of blocks. However, the methodologies may not be limited by the order of the blocks because, in some embodiments, the blocks may occur in different orders than shown and described. Moreover, fewer than all the illustrated blocks may be required to implement an example methodology. Blocks may be combined or separated into multiple components. Furthermore, additional or alternative methodologies can employ additional, not illustrated blocks.

[0029] Figure 2 illustrates an example method 200 associated with attributing a user action based on a biometric identity. In different examples, method 200 may be performed on a single device, may be performed partially or completely in the cloud, may be performed on distributed co-operating devices, or may be performed other ways. In different examples, method 200 may be performed on devices including, but not limited to, a game console, a computer, a laptop computer, a tablet computer, a phone, and a smart phone.

[0030] Method 200 includes, at 210, detecting an action intended to control an application associated with a shared launch surface. In one embodiment, the application may be a video game and the shared launch surface may be associated with a video game console. While a video game and a video game console are described, other multi-user systems that employ a shared launch surface may be employed. For example, the shared launch surface may be associated with a multi-user operating system running on laptop computer, a tablet computer, or a phone.

[0031] In one example, detecting the action may include detecting actions including, but not limited to a button press, a virtual button press, an interaction with a user interface, an interaction with a game controller, a manipulation of a game controller, a voice command, or a gesture. The button press may occur on a game controller, on the console, or on another device configured to provide button press signals to the system or application. A virtual button press may be performed using a voice command, a gesture, or other action where a non-physical button is "pressed". An interaction with a user interface may include a physical interaction through, for example, a capacitive touch interface, a gesture interaction, or other interaction. A manipulation of a game controller may include, for example, pressing buttons on the controller, moving arrows on the controller, turning knobs or dials on the controller, accelerating the controller in a direction, holding the controller in an orientation, or other actions.

[0032] Method 200 also includes, at 220, associating the action with a body located in a field of view of a biometric sensor associated with a shared launch surface. In one example, the associating may be location based. For example, sounds from a location that is occupied by a body can be associated with that body. Similarly, controller actions from a location occupied by a body can be associated with the controller. When a gesture is made, the gesture may already be associated with a body that made the gesture.

[0033] Method 200 also includes, at 230, determining, using data provided by the biometric sensor, a biometric identity for the body. In one example, determining the biometric identity involves performing actions including, but not limited to, facial recognition, gesture recognition, voice recognition, fingerprint recognition, haptic recognition, or retinal recognition. The recognitions may be based on information received from a biometric sensor(s). For example, facial recognition may be performed from contours present in a depth map or visible features present in a visible light picture.

[0034] Method 200 also includes, at 240, attributing the action to a user as a function of the biometric identity. Attributing the action may include updating a mapping, updating a table that stores information relating actions to users, or otherwise associating an action and a user.

[0035] Method 200 also includes, at 250, identifying a context associated with the user. The context may include an attribute and a state. The attribute may describe data that is relatively unchanging for a user. The attribute may be, for example, a user name, a screen name, a data plan identifier, a billing identifier, an account identifier, a parental control setting, a display preference, or a social media data. Different contexts may include a

greater or smaller number of attributes. The state may describe data that changes more frequently for a user. The state may be, for example, a location, a data plan balance, a billing balance, an account balance, an experience level, an access time, an engaged time, a location, or a connectivity level. Different contexts may include a greater or smaller number of state variables.

[0036] Method 200 also includes, at 260, selectively controlling the application or the shared launch surface as a function of the action and the context. In one example, the action may be intended to launch an application from the shared launch surface. In this example, selectively controlling the application or the shared launch surface may include launching the application from the shared launch surface using the context. In another example, the action may be intended to cause an operation in an application available through the shared launch surface. In this example, selectively controlling the application or the shared launch surface includes associating the context with the application and then causing the operation to occur in the application.

[0037] In one embodiment, selectively controlling the application or the shared launch surface may include launching a second application using the context while preserving a first context associated with a first application running on the shared launch surface. In this way, different applications may be associated with different contexts. For example, the game context may be associated with the person who launched the game while a stock checking popup may be associated with the person who wanted to check their stocks and a video application could be associated with the gamer who got the text and wanted to view the video immediately.

[0038] In one embodiment, selectively controlling the application or the shared launch surface may include selectively denying control of the application in response to the action based on the context or selectively denying launching an application based on the context. The denial may be based, for example, on a parental control, on a data plan limitation, or other information about the user.

[0039] Figure 3 illustrates a portion of an example method associated with attributing a user action based on a biometric identity. One part of attributing the user action may include identifying the action as a voice command, identifying the action as a gesture, or identifying a controller associated with the action. If the action is identified as a voice command at 270, then processing may include, at 272, identifying a location from which the voice command originated. The location may be determined by a directional microphone(s), by triangulating the sound, or by other location methods. Processing may

include, at 274, identifying a body corresponding to the location. The body may have been tracked by biometric sensors, IR sensors, visible light camera systems, or other apparatus in, for example, a game system. In one embodiment, the biometric identity is not being used to authenticate a user for security purposes (although it could), but rather is being used to identify who is currently at a location from which a voice command originated. Once the location of the voice and the location of the body have been identified, processing may include, at 276, mapping the body corresponding to the location to the voice command.

[0040] Attributing the user action may include identifying the action as a gesture at 280. If the action is identified as a gesture, then there will likely already be a body associated with the location at which the gesture was made, and processing may include, at 284, identifying the body that made the gesture.

[0041] Attributing the user action may include, at 290, identifying a game controller on which the action was performed. The game controller may be a wired controller, a wireless controller, a controller provided with the console, a controller provided by the gamer, or other controller. When a game controller is used, processing may include, at 292, identifying a location at which the game controller was located. Once the location has been identified, processing may include identifying a body corresponding to the location. The body may be identified using biometric identification. The biometrics are not being used to authenticate a user for security purposes, but rather are used to identify who is currently using a controller. Once the identification is made, processing may include, at 286, mapping the body corresponding to the location to the action.

[0042] While Figures 2 and 3 illustrates various actions occurring in serial, it is to be appreciated that various actions illustrated in Figures 2 and 3 could occur substantially in parallel. By way of illustration, a first process could detect actions, a second process could associate actions with bodies, a third process could produce a biometric identification of a body, and a fourth process could selectively control an operation as a function the action and a context associated with the biometric identification. While four processes are described, it is to be appreciated that a greater or lesser number of processes could be employed and that lightweight processes, regular processes, threads, and other approaches could be employed.

[0043] In one example, a method may be implemented as computer executable instructions. Thus, in one example, a computer-readable storage medium may store computer executable instructions that if executed by a machine (e.g., computer) cause the

machine to perform methods described or claimed herein including method 200. While executable instructions associated with the above method are described as being stored on a computer-readable storage medium, it is to be appreciated that executable instructions associated with other example methods described or claimed herein may also be stored on a computer-readable storage medium. In different embodiments the example methods described herein may be triggered in different ways. In one embodiment, a method may be triggered manually by a user. In another example, a method may be triggered automatically.

[0044] "Computer-readable storage medium", as used herein, refers to a medium that stores instructions or data. "Computer-readable storage medium" does not refer to propagated signals, *per se*. A computer-readable storage medium may take forms, including, but not limited to, non-volatile media, and volatile media. Non-volatile media may include, for example, optical disks, magnetic disks, tapes, flash memory, ROM, and other media. Volatile media may include, for example, semiconductor memories, dynamic memory (e.g., dynamic random access memory (DRAM), synchronous dynamic random access memory (SDRAM), double data rate synchronous dynamic random-access memory (DDR SDRAM), etc.), and other media. Common forms of a computer-readable storage medium may include, but are not limited to, a floppy disk, a flexible disk, a hard disk, a magnetic tape, other magnetic medium, an application specific integrated circuit (ASIC), a compact disk (CD), other optical medium, a random access memory (RAM), a read only memory (ROM), a memory chip or card, a memory stick, and other media from which a computer, a processor or other electronic device can read.

[0045] Figure 4 illustrates an example game environment and different user actions. A console 400 may detect actions including a controller action 410, a voice command 420, or a gesture 430. Detecting a controller action 410 may include receiving an electrical signal through a wire that connects a controller to console 410. Detecting the controller action 410 may also include receiving a data signal from a wireless controller through a wireless communication medium (e.g., radio frequency signal). Detecting the controller action 410 may not involve any biometric sensor.

[0046] Detecting a voice command 420 may include receiving sounds at a microphone(s). The microphone(s) may be directional. Voices may be distinguishable between users. Therefore, detecting a voice command 420 may involve a sensor(s) that is also used for a biometric identity. Like the controller action 410, detecting the voice

command 420 may be a passive or unidirectional operation where signals flow from the device or person to the console 400.

[0047] Detecting a gesture 430 may involve an active operation that includes sending signals out into a game space and receiving reflected signals. Detecting a gesture may include identifying where the gesture occurred, what the gesture was (e.g., flick, pinch, spread, wave), the speed at which the gesture was performed, and other attributes of a gesture.

[0048] Figure 5 illustrates an apparatus 500 that includes a processor 510, a memory 520, a set 530 of logics, a biometric sensor 599, and an interface 540 that connects the processor 510, the memory 520, the biometric sensor 599, and the set 530 of logics. The memory 520 may be configured to store information concerning ownership of a shared launch surface. The set 530 of logics may be configured to attribute a user action as a function of a biometric identity performed using data provided by the biometric sensor 599. Apparatus 500 may be, for example, a game console, a device acting as a game console, a computer, a laptop computer, a tablet computer, a personal electronic device, a smart phone, system-on-a-chip (SoC), or other device that can access and process data.

[0049] In one embodiment, the apparatus 500 may be a general purpose computer that has been transformed into a special purpose computer through the inclusion of the set 530 of logics. Apparatus 500 may interact with other apparatus, processes, and services through, for example, a computer network.

[0050] The set 530 of logics may include a first logic 532 that is configured to track one or more bodies located in a field of detection of the biometric sensor. In one embodiment, the first logic 532 may be configured to track the one or more bodies in real time using data provided by the biometric sensor 599. In this embodiment, the third logic 536 may be configured to maintain, in real time, a mapping between a member of the one or more bodies and a biometric identity.

[0051] The set 530 of logics may also include a second logic 534 that is configured to identify an action performed by a member of the one or more bodies. In different embodiments, the second logic 534 may be configured to identify a controller action, a voice command, a gesture, or other operation initiating action.

[0052] The set 530 of logics may also include a third logic 536 that is configured to produce a biometric identification of the member using data provided by the biometric sensor 599. The biometric sensor 599 may be, for example an infrared (IR) sensor, a vision system, a haptic system, a sound system, or a voice system. In one embodiment, the

third logic 536 may be configured to acquire a set of user specific data associated with the biometric identification. The user specific data may include attribute data and state data.

[0053] The set 530 of logics may also include a fourth logic 538 that is configured to selectively control an operation of the apparatus as a function of the biometric identification. In one embodiment, the operation may be intended to launch an application from the shared launch surface using the user specific data. For example, while a game is in progress, a gamer may want to view a recently released video. The gamer may start a video watching app in a small window on their display without disrupting other gamers. The video watching app might cost the gamer some money or might count against the gamer's data plan. Thus, launching the video watching app using the gamer's context may allow this direct billing. In another embodiment, the operation may be intended to be performed in an application launched from the shared launch surface. In this case, the operation may be controlled, at least in part, by the user specific data.

[0054] In different embodiments, some processing may be performed on the apparatus 500 and some processing may be performed by an external service or apparatus. Thus, in one embodiment, apparatus 500 may also include a communication circuit that is configured to communicate with an external source to facilitate accessing or processing action data, user data, biometric identity data, or other data associated with attributing a user action. In one embodiment, the set 530 of logics may interact with a presentation service 560 to facilitate displaying data using different presentations for different devices.

[0055] Figure 6 illustrates an apparatus 600 that is similar to apparatus 500 (Figure 5). For example, apparatus 600 includes a processor 610, a memory 620, a set of logics 630 (e.g., 632, 634, 636, 638) that correspond to the set of logics 530 (Figure 5), a biometric sensor 699, and an interface 640. However, apparatus 600 includes an additional fifth logic 639. The fifth logic 639 may be configured to control ownership of the shared launch surface. The ownership may be controlled as a function of the action and the set of user specific data. For example, a first action may not require surface ownership to be changed while a second action may require surface ownership to change. When surface ownership needs to be changed, the change may be from a current owner to a subsequent owner that is identified by the context. Additionally, attributes of the launch surface may depend on the context. For example, the set of operations that can be performed by the launch surface may be limited or expanded based on the context. In one embodiment, controlling ownership involves manipulating (e.g., reading, writing, updating) the information stored in the memory 620.

[0056] Figure 7 illustrates an example cloud operating environment 700. A cloud operating environment 700 supports delivering computing, processing, storage, data management, applications, and other functionality as an abstract service rather than as a standalone product. Services may be provided by virtual servers that may be implemented as one or more processes on one or more computing devices. In some embodiments, processes may migrate between servers without disrupting the cloud service. In the cloud, shared resources (e.g., computing, storage) may be provided to computers including servers, clients, and mobile devices over a network. Different networks (e.g., Ethernet, Wi-Fi, 802.x, cellular) may be used to access cloud services. Users interacting with the cloud may not need to know the particulars (e.g., location, name, server, database) of a device that is actually providing the service (e.g., computing, storage). Users may access cloud services via, for example, a web browser, a thin client, a mobile application, or in other ways.

[0057] Figure 7 illustrates an example action attribution service 760 residing in the cloud. The action attribution service 760 may rely on a server 702 or service 704 to perform processing and may rely on a data store 706 or database 708 to store data. While a single server 702, a single service 704, a single data store 706, and a single database 708 are illustrated, multiple instances of servers, services, data stores, and databases may reside in the cloud and may, therefore, be used by the action attribution service 760.

[0058] Figure 7 illustrates various devices accessing the action attribution service 760 in the cloud. The devices include a computer 710, a tablet 720, a laptop computer 730, a personal digital assistant 740, and a mobile device (e.g., cellular phone, satellite phone, wearable computing device) 750. The action attribution service 760 may store, access, or process action data, user data, biometric data, mapping data, or other data associated with connecting a user action to a context and controlling an operation (e.g., application launch, application operation) based on the action, context, and mapping.

[0059] It is possible that different users at different locations using different devices may access the action attribution service 760 through different networks or interfaces. In one example, the action attribution service 760 may be accessed by a mobile device 750. In another example, portions of action attribution service 760 may reside on a mobile device 750. In one example, action attribution service 760 may dynamically, on a per-action basis, map a real-time biometric identity with a detected action in a video game environment. This may allow flexibility (e.g., switching controllers, switching interaction approach) that is not present in conventional systems.

[0060] Figure 8 is a system diagram depicting an exemplary mobile device 800 that includes a variety of optional hardware and software components, shown generally at 802. Components 802 in the mobile device 800 can communicate with other components, although not all connections are shown for ease of illustration. The mobile device 800 may be a variety of computing devices (e.g., cell phone, smartphone, handheld computer, Personal Digital Assistant (PDA), wearable computing device, etc.) and may allow wireless two-way communications with one or more mobile communications networks 804, such as a cellular or satellite networks.

[0061] Mobile device 800 can include a controller or processor 810 (e.g., signal processor, microprocessor, ASIC, or other control and processing logic circuitry) for performing tasks including signal coding, data processing, input/output processing, power control, or other functions. An operating system 812 can control the allocation and usage of the components 802 and support application programs 814. The application programs 814 can include gaming applications, mobile computing applications (e.g., email applications, calendars, contact managers, web browsers, messaging applications), or other computing applications. In different embodiments, mobile device 800 may function as a game console or game controller.

[0062] Mobile device 800 can include memory 820. Memory 820 can include non-removable memory 822 or removable memory 824. The non-removable memory 822 can include random access memory (RAM), read only memory (ROM), flash memory, a hard disk, or other memory storage technologies. The removable memory 824 can include flash memory or a Subscriber Identity Module (SIM) card, which is well known in GSM communication systems, or other memory storage technologies, such as "smart cards." The memory 820 can be used for storing data or code for running the operating system 812 and the applications 814. Example data can include web pages, text, images, sound files, video data, or other data sets to be sent to or received from one or more network servers or other devices *via* one or more wired or wireless networks. The memory 820 can be used to store a subscriber identifier, such as an International Mobile Subscriber Identity (IMSI), and an equipment identifier, such as an International Mobile Equipment Identifier (IMEI). The identifiers can be transmitted to a network server to identify users or equipment.

[0063] The mobile device 800 can support one or more input devices 830 including, but not limited to, a touchscreen 832, a microphone 834, a camera 836, a physical keyboard 838, or trackball 840. The mobile device 800 may also support output devices 850

including, but not limited to, a speaker 852 and a display 854. Other possible output devices (not shown) can include piezoelectric or other haptic output devices. Some devices can serve more than one input/output function. For example, touchscreen 832 and display 854 can be combined in a single input/output device. The input devices 830 can include a Natural User Interface (NUI). An NUI is an interface technology that enables a user to interact with a device in a "natural" manner, free from artificial constraints imposed by input devices such as mice, keyboards, remote controls, and others. Examples of NUI methods include those relying on speech recognition, touch and stylus recognition, gesture recognition (both on screen and adjacent to the screen), air gestures, head and eye tracking, voice and speech, vision, touch, gestures, and machine intelligence. Other examples of a NUI include motion gesture detection using accelerometers/gyroscopes, facial recognition, three dimensional (3D) displays, head, eye, and gaze tracking, immersive augmented reality and virtual reality systems, all of which provide a more natural interface, as well as technologies for sensing brain activity using electric field sensing electrodes (EEG and related methods). Thus, in one specific example, the operating system 812 or applications 814 can comprise speech-recognition software as part of a voice user interface that allows a user to operate the device 800 via voice commands. Further, the device 800 can include input devices and software that allow for user interaction via a user's spatial gestures, such as detecting and interpreting gestures to provide input to a gaming application.

[0064] A wireless modem 860 can be coupled to an antenna 891. In some examples, radio frequency (RF) filters are used and the processor 810 need not select an antenna configuration for a selected frequency band. The wireless modem 860 can support two-way communications between the processor 810 and external devices. The modem 860 is shown generically and can include a cellular modem for communicating with the mobile communication network 804 and/or other radio-based modems (e.g., Bluetooth 864 or Wi-Fi 862). The wireless modem 860 may be configured for communication with one or more cellular networks, such as a Global system for mobile communications (GSM) network for data and voice communications within a single cellular network, between cellular networks, or between the mobile device and a public switched telephone network (PSTN). Near field communication (NFC) element 892 facilitates having near field communications.

[0065] The mobile device 800 may include at least one input/output port 880, a power supply 882, a satellite navigation system receiver 884, such as a Global Positioning

System (GPS) receiver, an accelerometer 886, or a physical connector 890, which can be a Universal Serial Bus (USB) port, IEEE 1394 (FireWire) port, RS-232 port, or other port. The illustrated components 802 are not required or all-inclusive, as other components can be deleted or added.

[0066] Mobile device 800 may include an attribution logic 899 that is configured to provide a functionality for the mobile device 800. For example, attribution logic 899 may provide a client for interacting with a service (e.g., service 760, figure 7). Portions of the example methods described herein may be performed by attribution logic 899. Similarly, attribution logic 899 may implement portions of apparatus described herein.

[0067] The following includes definitions of selected terms employed herein. The definitions include various examples or forms of components that fall within the scope of a term and that may be used for implementation. The examples are not intended to be limiting. Both singular and plural forms of terms may be within the definitions.

[0068] References to "one embodiment", "an embodiment", "one example", and "an example" indicate that the embodiment(s) or example(s) so described may include a particular feature, structure, characteristic, property, element, or limitation, but that not every embodiment or example necessarily includes that particular feature, structure, characteristic, property, element or limitation. Furthermore, repeated use of the phrase "in one embodiment" does not necessarily refer to the same embodiment, though it may.

[0069] "Data store", as used herein, refers to a physical or logical entity that can store data. A data store may be, for example, a database, a table, a file, a list, a queue, a heap, a memory, a register, and other physical repository. In different examples, a data store may reside in one logical or physical entity or may be distributed between two or more logical or physical entities.

[0070] "Logic", as used herein, includes but is not limited to hardware, firmware, software in execution on a machine, or combinations of each to perform a function(s) or an action(s), or to cause a function or action from another logic, method, or system. Logic may include a software controlled microprocessor, a discrete logic (e.g., ASIC), an analog circuit, a digital circuit, a programmed logic device, a memory device containing instructions, and other physical devices. Logic may include one or more gates, combinations of gates, or other circuit components. Where multiple logical logics are described, it may be possible to incorporate the multiple logical logics into one physical logic. Similarly, where a single logical logic is described, it may be possible to distribute that single logical logic between multiple physical logics.

[0071] To the extent that the term "includes" or "including" is employed in the detailed description or the claims, it is intended to be inclusive in a manner similar to the term "comprising" as that term is interpreted when employed as a transitional word in a claim.

[0072] To the extent that the term "or" is employed in the detailed description or claims (e.g., A or B) it is intended to mean "A or B or both". When the Applicant intends to indicate "only A or B but not both" then the term "only A or B but not both" will be employed. Thus, use of the term "or" herein is the inclusive, and not the exclusive use. See, Bryan A. Garner, A Dictionary of Modern Legal Usage 624 (2d. Ed. 1995).

[0073] Although the subject matter has been described in language specific to structural features or methodological acts, it is to be understood that the subject matter defined in the appended claims is not necessarily limited to the specific features or acts described above. Rather, the specific features and acts described above are disclosed as example forms of implementing the claims.

CLAIMS

1. A method, comprising:

detecting an action intended to control an application associated with a shared launch surface;

associating the action with a body located in a field of view of a biometric sensor associated with a shared launch surface;

determining, using data provided by the biometric sensor, a biometric identity for the body;

attributing the action to a user as a function of the biometric identity;

identifying a context associated with the user, and

selectively controlling the application or the shared launch surface as a function of the action and the context.

2. The method of claim 1, the application being a video game and the shared launch surface being associated with a video game console.

3. The method of claim 1, where detecting the action comprises detecting a button press, detecting a virtual button press, detecting an interaction with a user interface, detecting an interaction with a game controller, detecting a manipulation of a game controller, detecting a voice command, detecting an eye movement, detecting a brain wave, or detecting a gesture.

4. The method of claim 3, the action being intended to launch an application from the shared launch surface and where selectively controlling the application or the shared launch surface comprises launching the application from the shared launch surface using the context.

5. The method of claim 3, the action being intended to cause an operation in an application available through the shared launch surface and where selectively controlling the application or the shared launch surface comprises associating the context with the application and then selectively causing or preventing the operation to occur.

6. The method of claim 1, the shared launch surface being associated with a multi-user operating system running on laptop computer, a tablet computer, or a phone.

7. The method of claim 1, where associating the action with the body comprises:
identifying a game controller on which the action was performed;
identifying a location at which the game controller was located;
identifying a body corresponding to the location, and
mapping the body corresponding to the location to the action.

8. The method of claim 1, where determining the biometric identity includes performing, in real-time, on a per-act basis, facial recognition, gesture recognition, voice recognition, fingerprint recognition, haptic recognition, breathing pattern recognition, apparel recognition, height recognition, ear shape recognition, or retinal recognition.

9. The method of claim 1, the context comprising an attribute and a state, the attribute being a user name, a screen name, a data plan identifier, a billing identifier, an account identifier, a parental control setting, a display preference, or a social media data, and the state being a location, a data plan balance, a billing balance, an account balance, an experience level, an access time, a engaged time, a location, a user language, a saved game, a saved file, a purchased content, or a connectivity level.

10. The method of claim 1, where selectively controlling the application or the shared launch surface comprises launching a second application using the context while preserving a first context associated with a first application running on the shared launch surface, or selectively denying control of the application in response to the action based on the context or selectively denying launching an application based on the context.

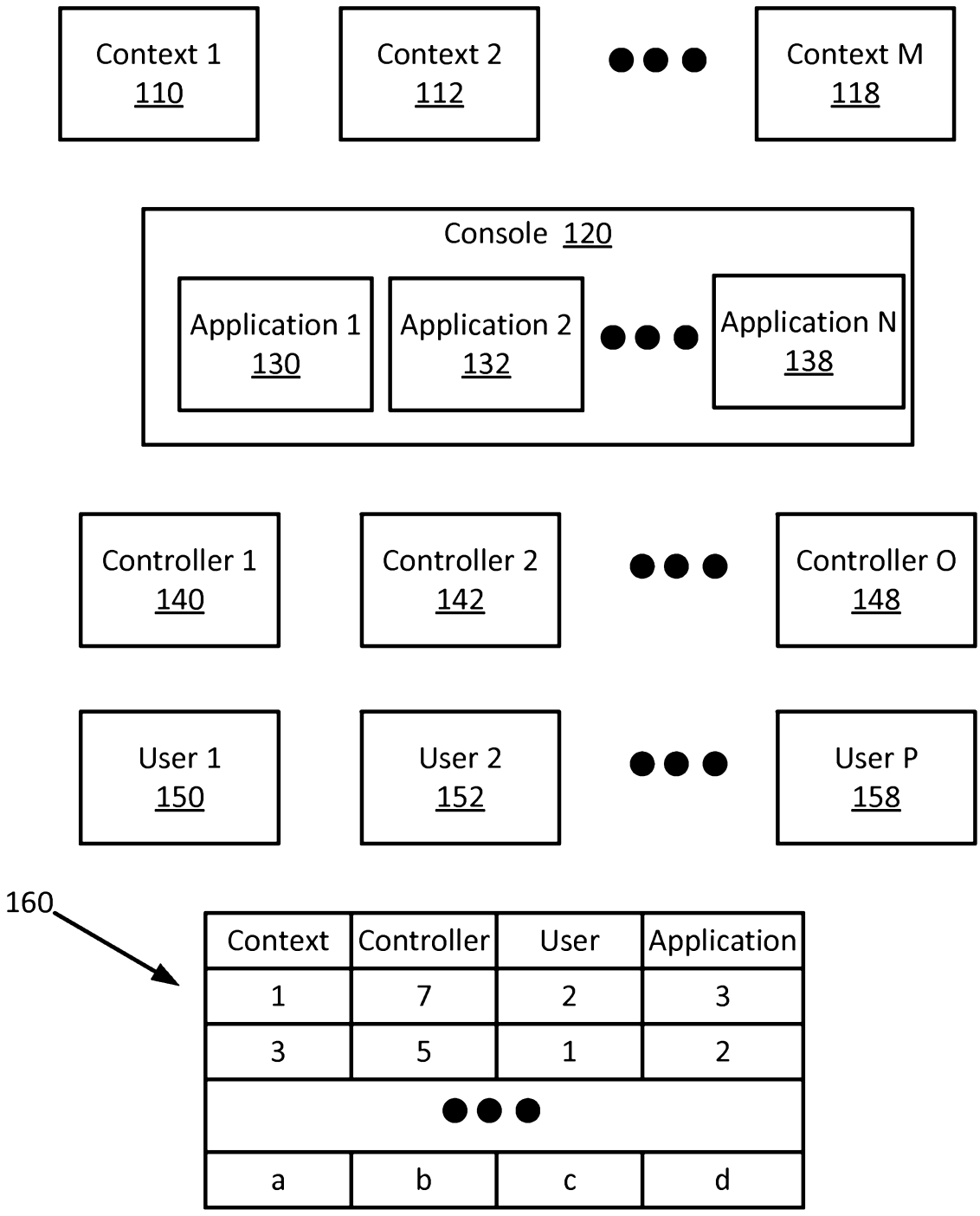
Context 1
110
Context 2
112
Context M
118
Console 120
Application 1
130
Application 2
132
Application N
138
Controller 1
140
Controller 2
142
Controller O
148
User 1
150
User 2
152
User P
158
160
Context
Controller
User
Application
1
7
2
3
3
5
1
2
a
b
c
d

FIG. 1

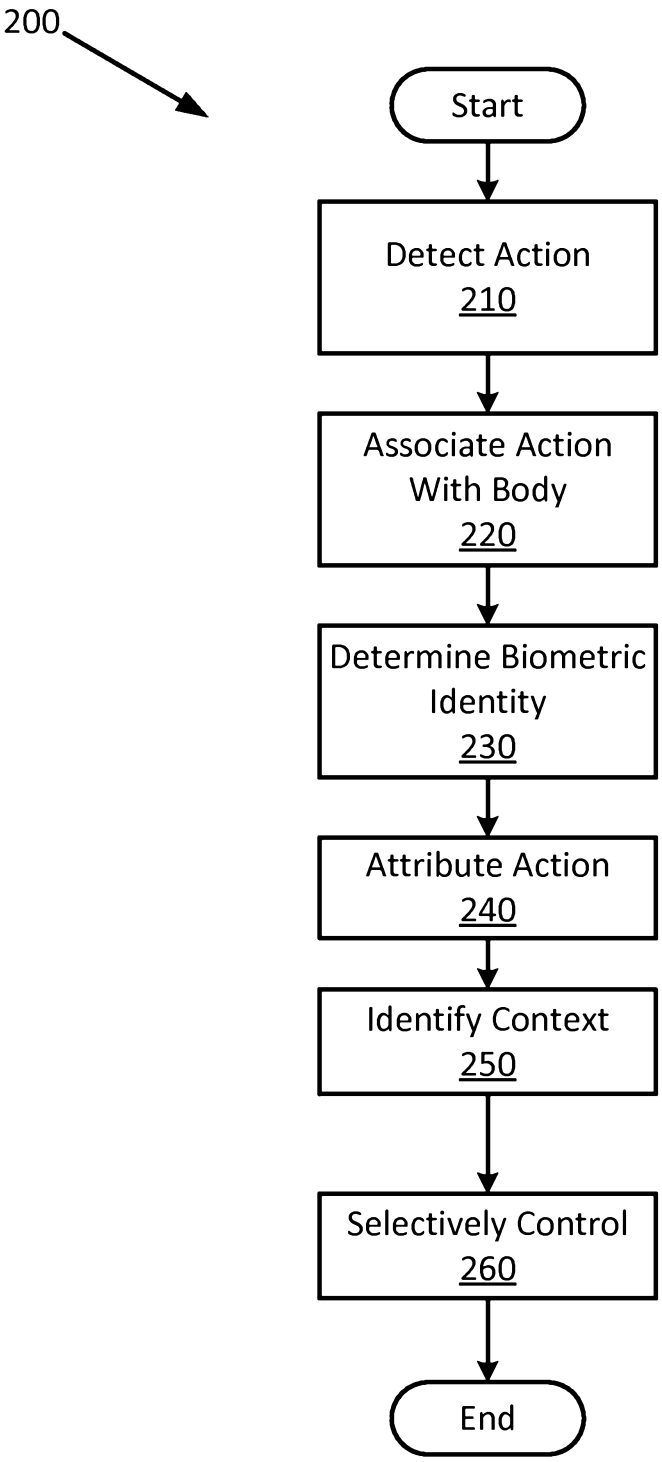
200
Start
Detect Action
210
Associate Action
With Body
220
Determine Biometric
Identity
230
Attribute Action
240
Identify Context
250
Selectively Control
260
End

FIG. 2

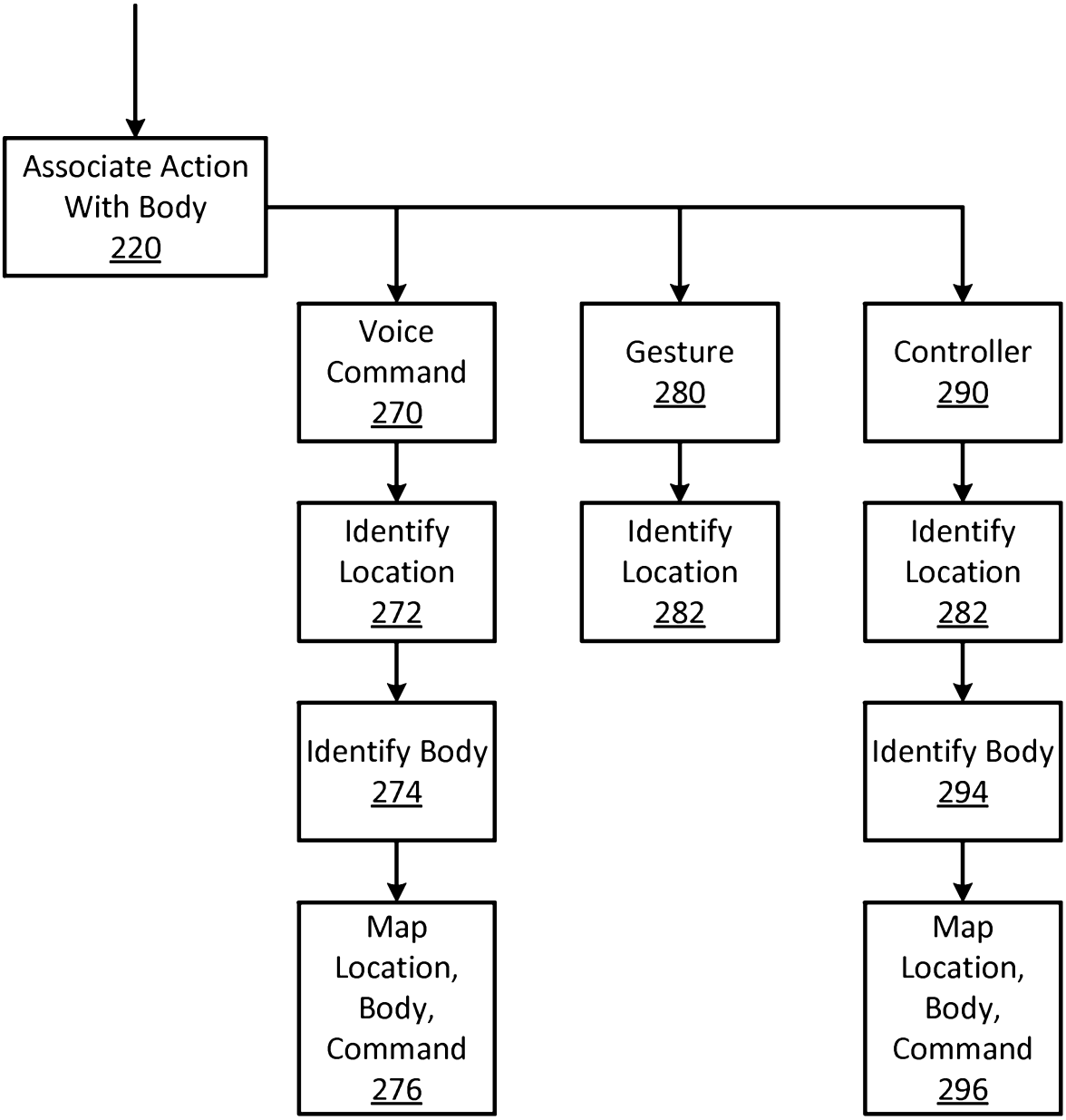
Associate Action With Body 220
Voice Command 270
Gesture 280
Controller 290
Identify Location 272
Identify Location 282
Identify Location 282
Identify Body 274
Identify Body 294
Map Location, Body, Command 276
Map Location, Body, Command 296

FIG. 3

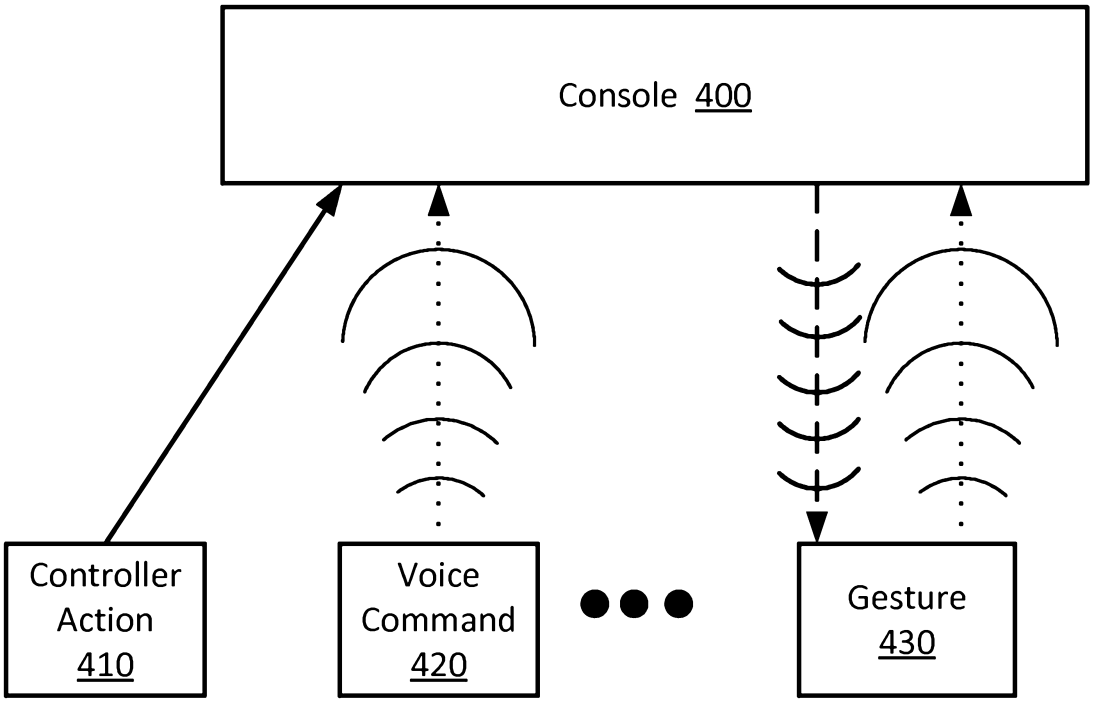
Console 400
Controller Action 410
Voice Command 420
Gesture 430

FIG. 4

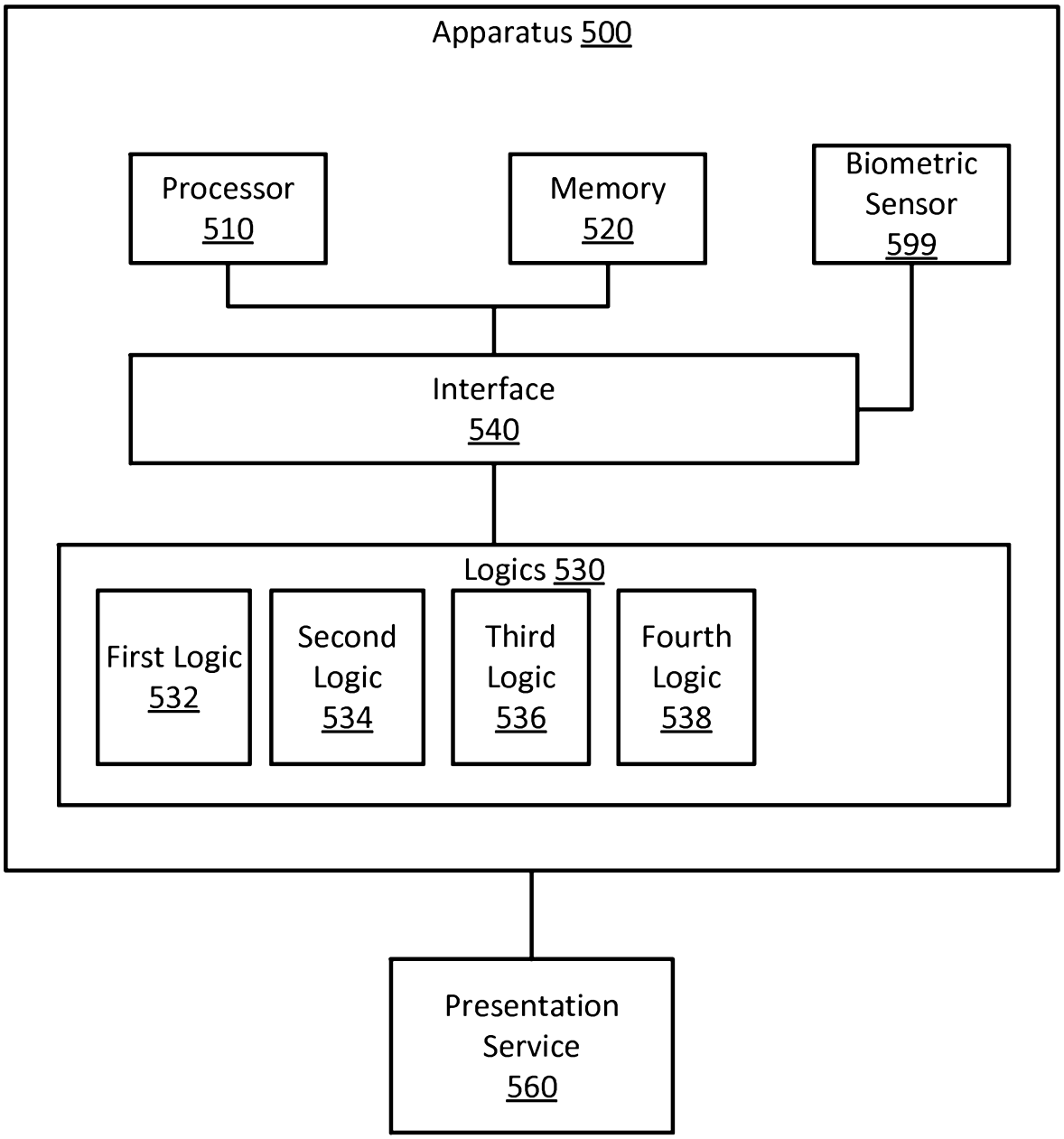
Apparatus 500
Processor 510
Memory 520
Biometric Sensor 599
Interface 540
Logics 530
First Logic 532
Second Logic 534
Third Logic 536
Fourth Logic 538
Presentation Service 560

FIG. 5

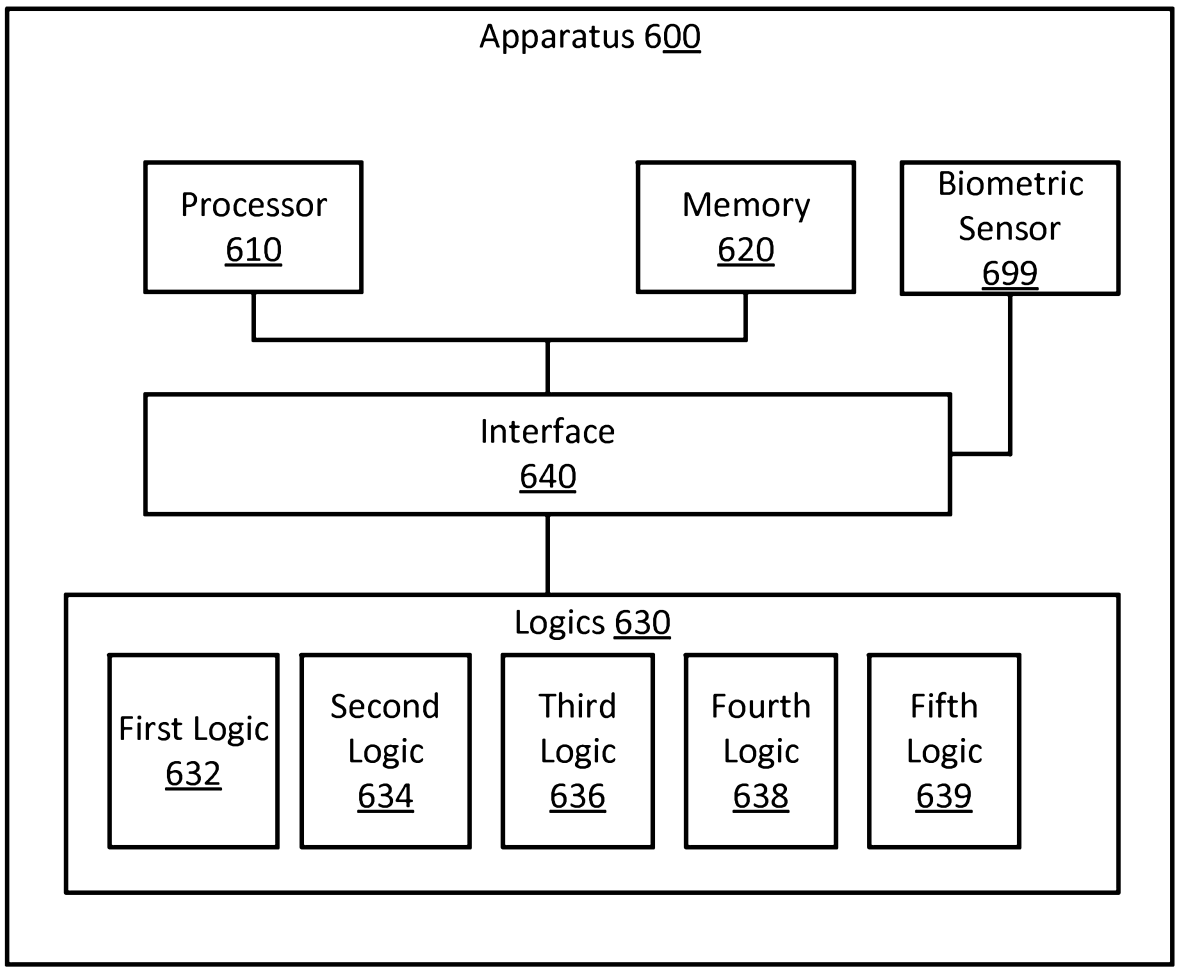
Apparatus 600
Processor 610
Memory 620
Biometric Sensor 699
Interface 640
Logics 630
First Logic 632
Second Logic 634
Third Logic 636
Fourth Logic 638
Fifth Logic 639

FIG. 6

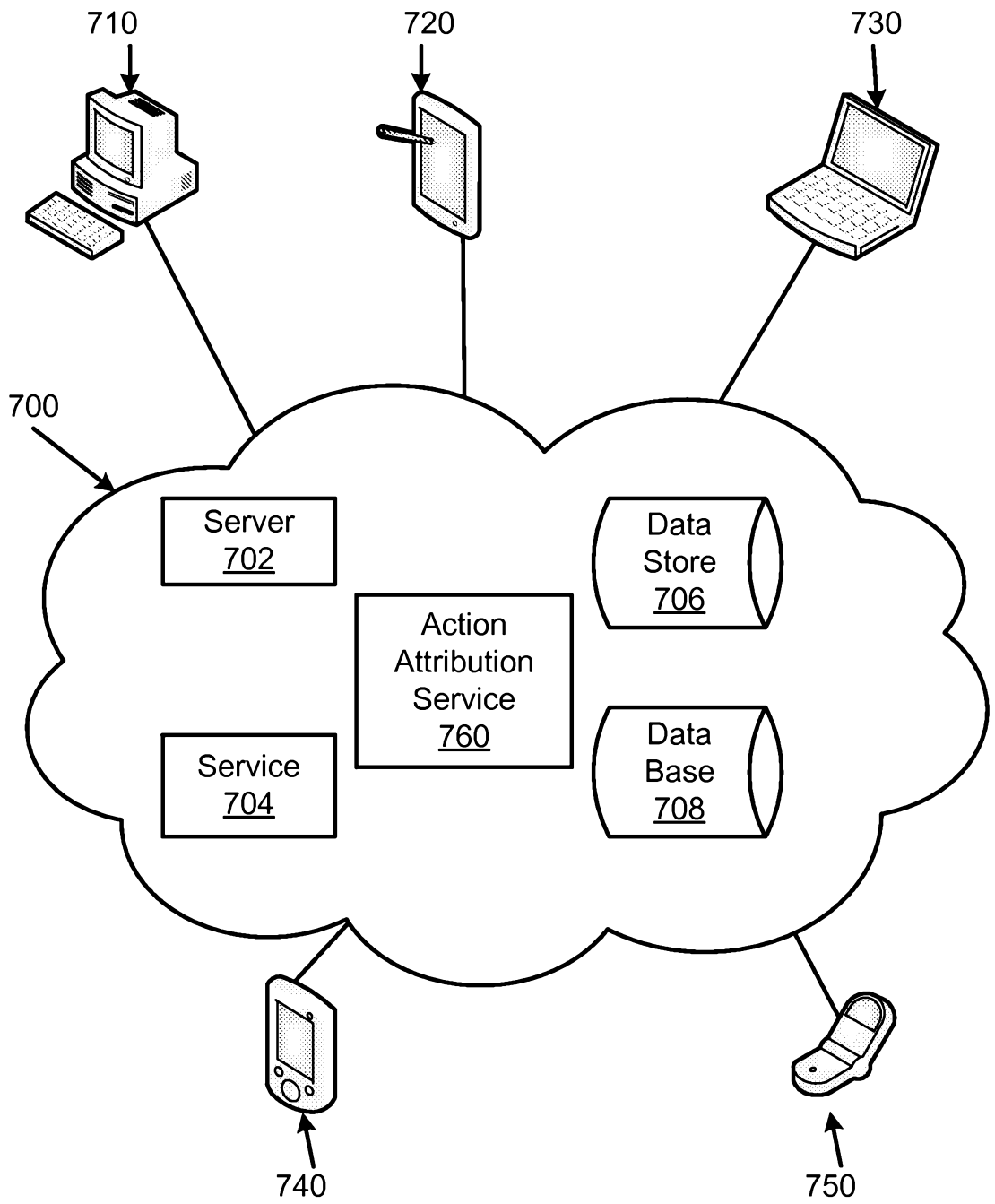
710
720
730
700
Server
702
Data
Store
706
Action
Attribution
Service
760
Service
704
Data
Base
708
740
750

FIG. 7

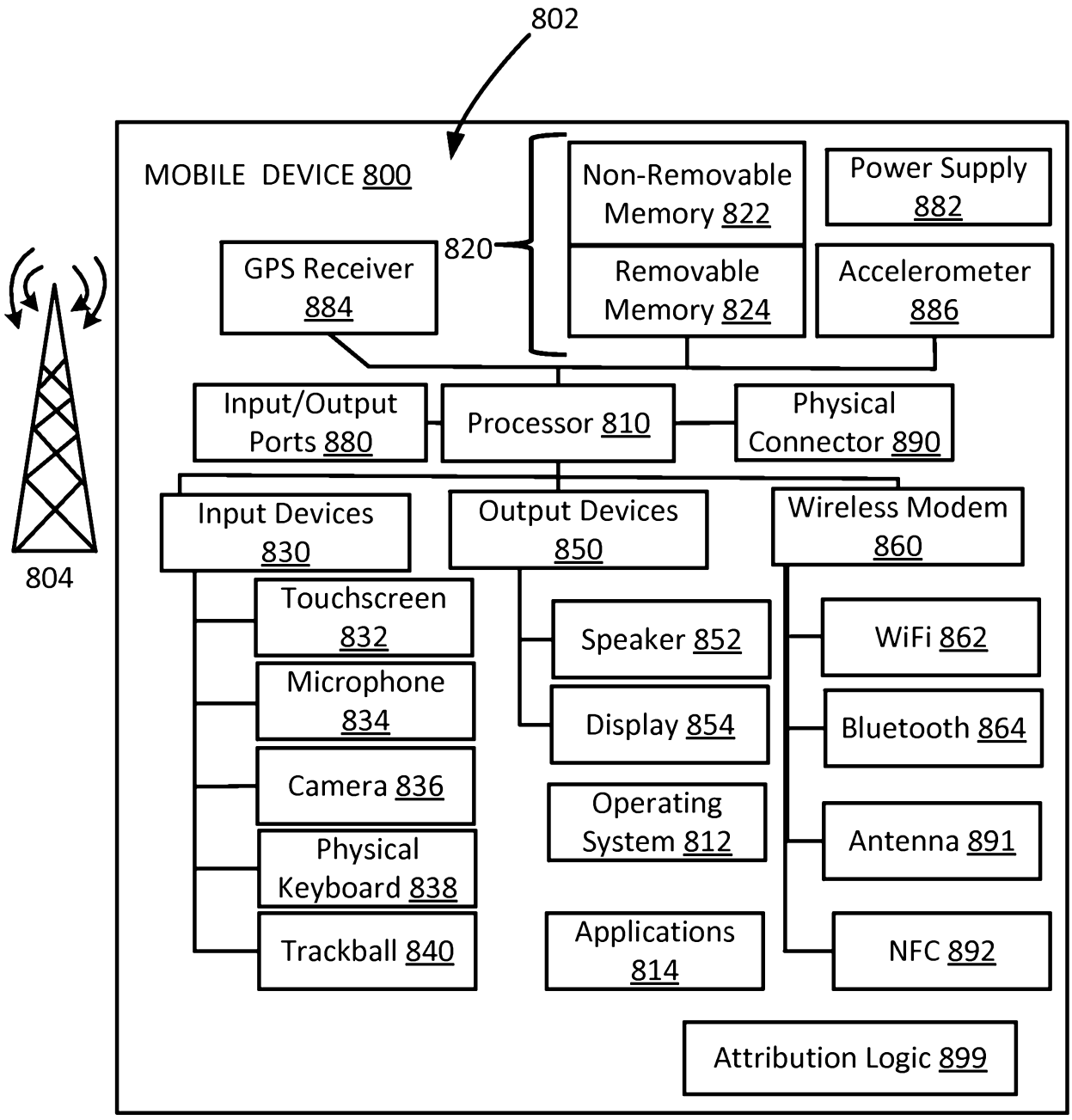
802
MOBILE DEVICE 800
Non-Removable Memory 822
Power Supply 882
820
GPS Receiver 884
Removable Memory 824
Accelerometer 886
Input/Output Ports 880
Processor 810
Physical Connector 890
Input Devices 830
Output Devices 850
Wireless Modem 860
804
Touchscreen 832
Speaker 852
WiFi 862
Microphone 834
Display 854
Bluetooth 864
Camera 836
Operating System 812
Antenna 891
Physical Keyboard 838
Trackball 840
Applications 814
NFC 892
Attribution Logic 899

FIG. 8