



US 20160282988A1

(19) **United States**

(12) **Patent Application Publication**
Poupyrev

(10) **Pub. No.: US 2016/0282988 A1**

(43) **Pub. Date: Sep. 29, 2016**

(54) **TWO-LAYER INTERACTIVE TEXTILES**
(71) Applicant: **Google Inc.**, Mountain View, CA (US)
(72) Inventor: **Ivan Poupyrev**, Sunnyvale, CA (US)
(21) Appl. No.: **14/959,730**
(22) Filed: **Dec. 4, 2015**

(52) **U.S. Cl.**
CPC **G06F 3/044** (2013.01); **G06F 3/0416**
(2013.01); **D03D 25/005** (2013.01); **D03D**
1/0088 (2013.01); **G06F 2203/04102** (2013.01);
G06F 2203/04111 (2013.01); **D03D 2700/0166**
(2013.01); **D10B 2401/16** (2013.01); **D10B**
2401/18 (2013.01); **G06F 2203/04103**
(2013.01)

Related U.S. Application Data

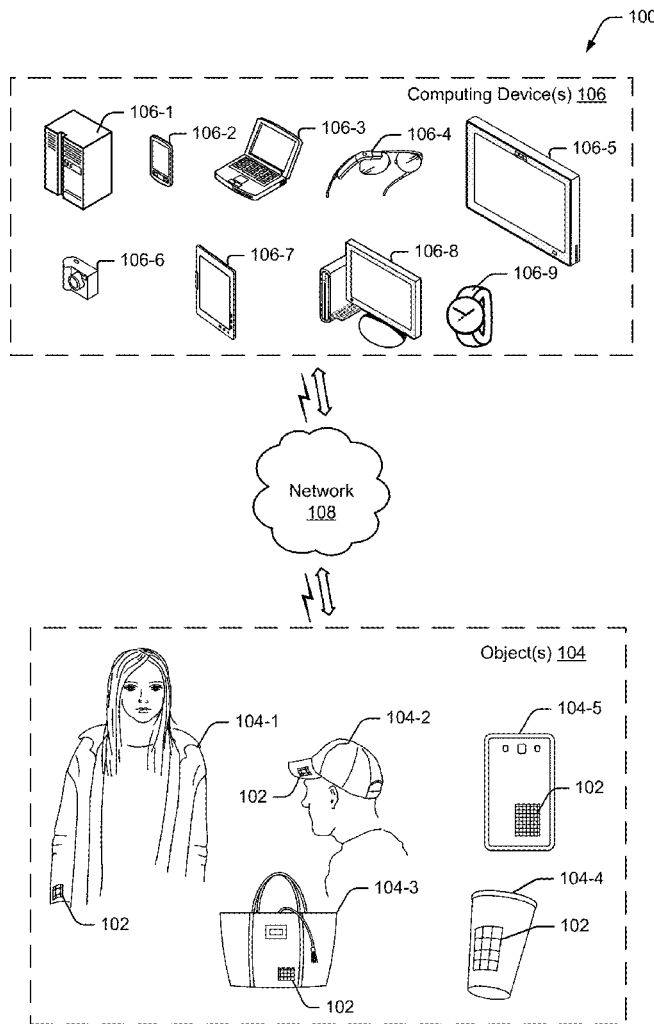
(60) Provisional application No. 62/138,831, filed on Mar. 26, 2015.

Publication Classification

(51) **Int. Cl.**
G06F 3/044 (2006.01)
D03D 25/00 (2006.01)
D03D 1/00 (2006.01)
G06F 3/041 (2006.01)

(57) **ABSTRACT**

This document describes two-layer interactive textiles. In one or more implementations, the interactive textile includes a top textile layer and a bottom textile layer. Conductive threads are woven into the top textile layer and the bottom textile layer. When the top textile layer is combined with the bottom textile layer, the conductive threads from each layer form a capacitive touch sensor that is configured to detect touch-input. The bottom textile layer is not visible and couples the capacitive through sensor to electronic components, such as a controller, a wireless interface, an output device (e.g., an LED, a display, or speaker), and so forth.



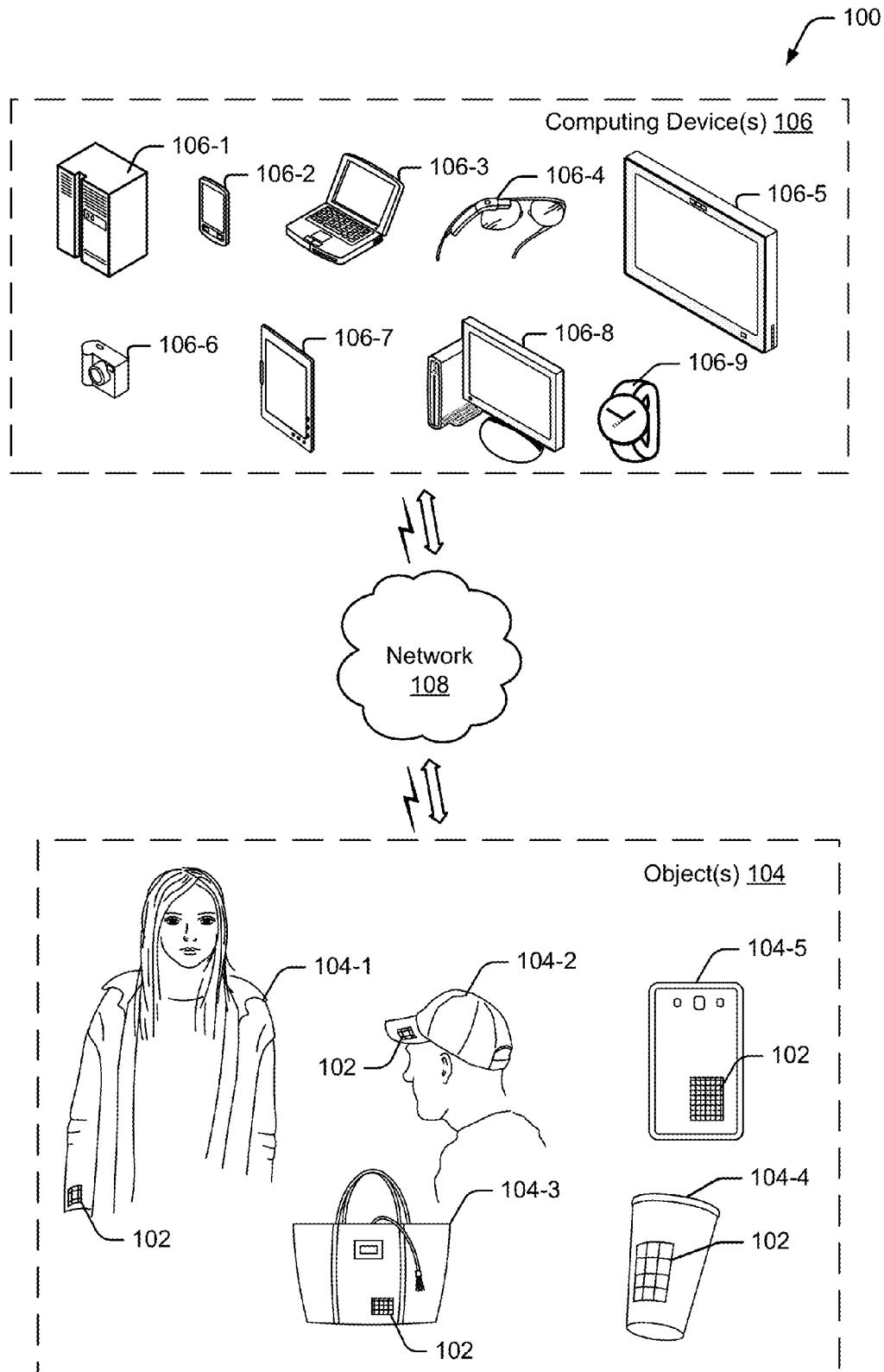


Fig. 1

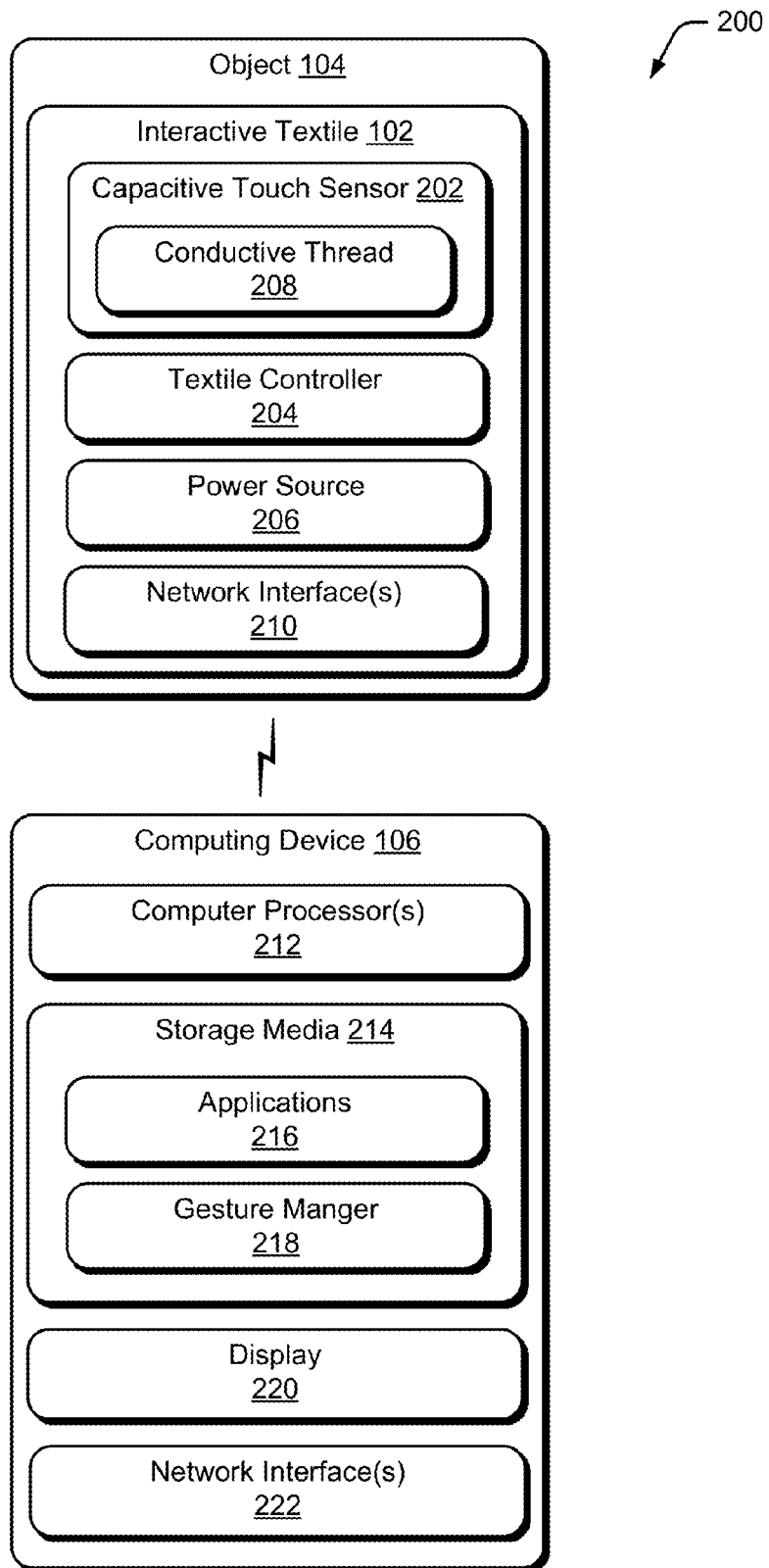


Fig. 2

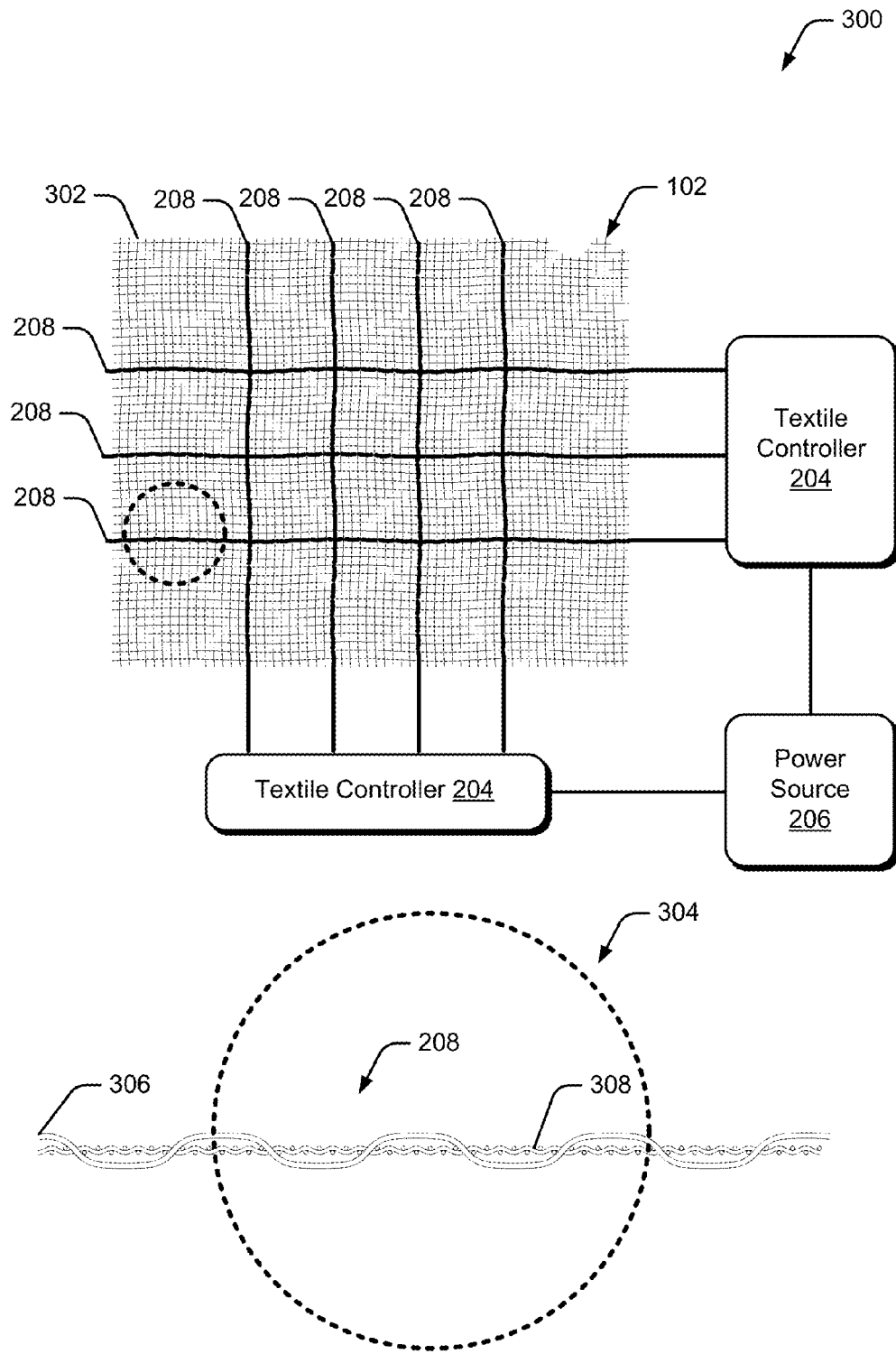


Fig. 3

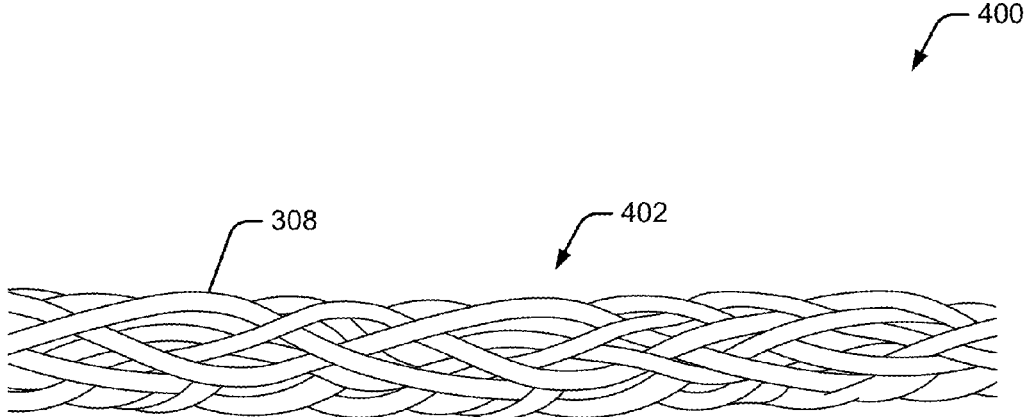


Fig. 4a

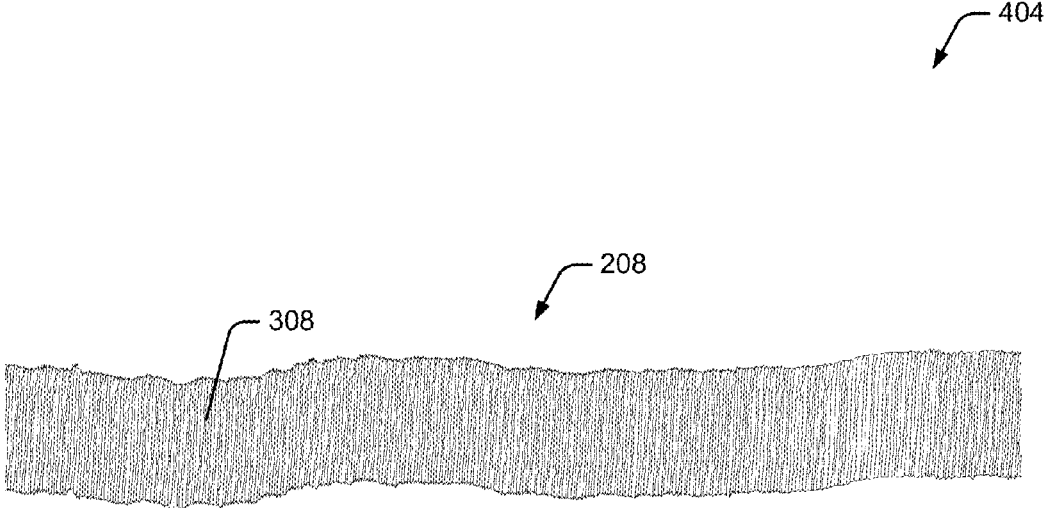


Fig. 4b

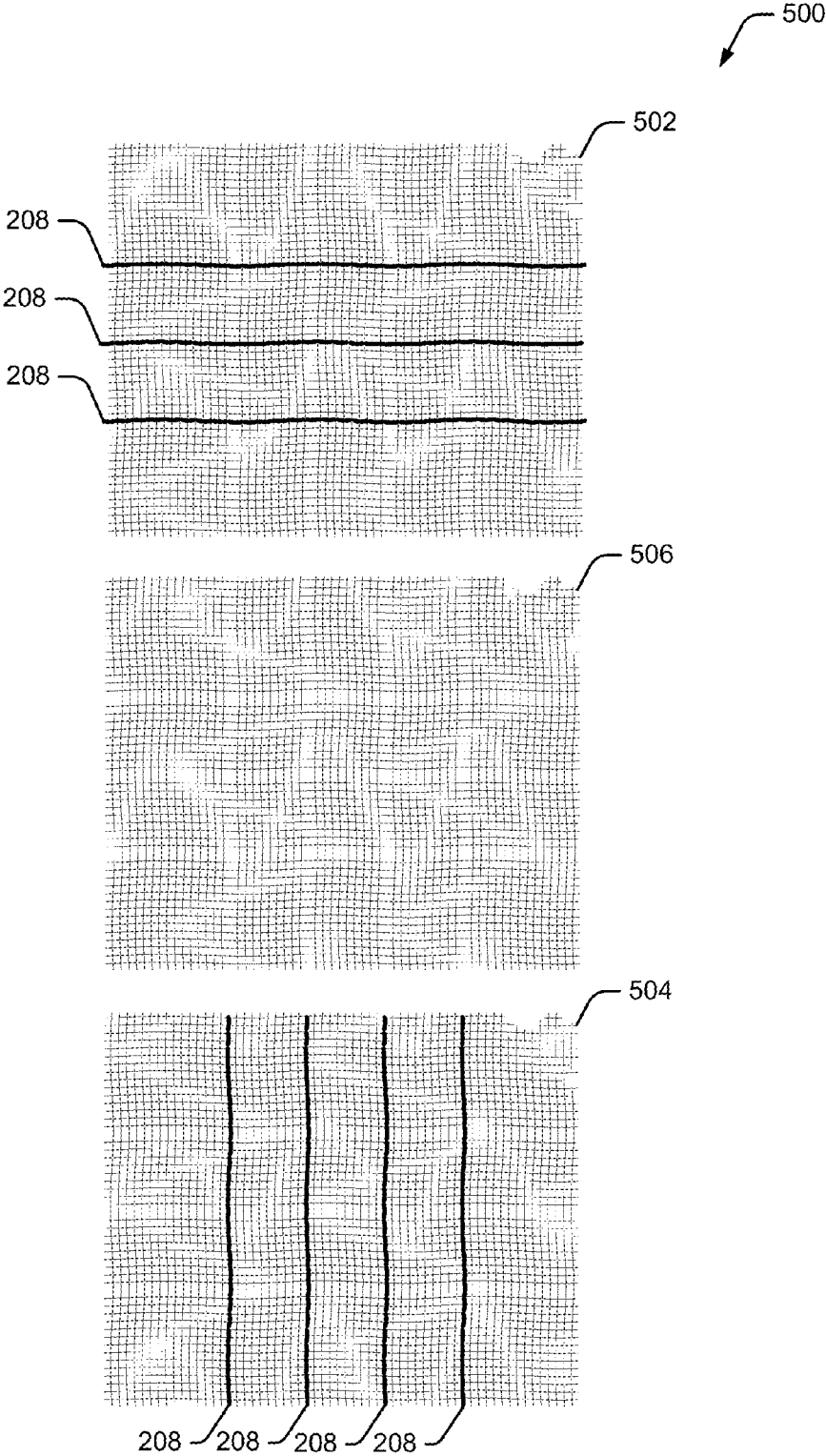


Fig. 5

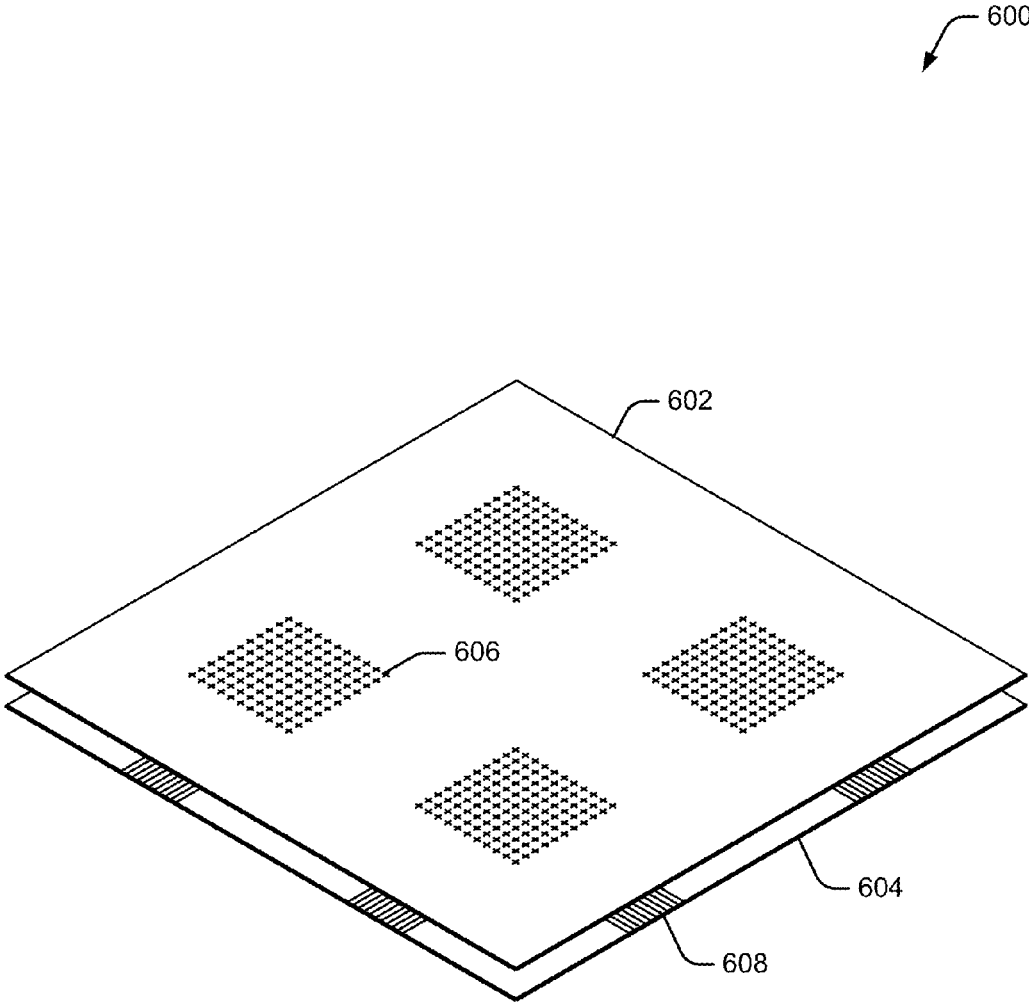


Fig. 6

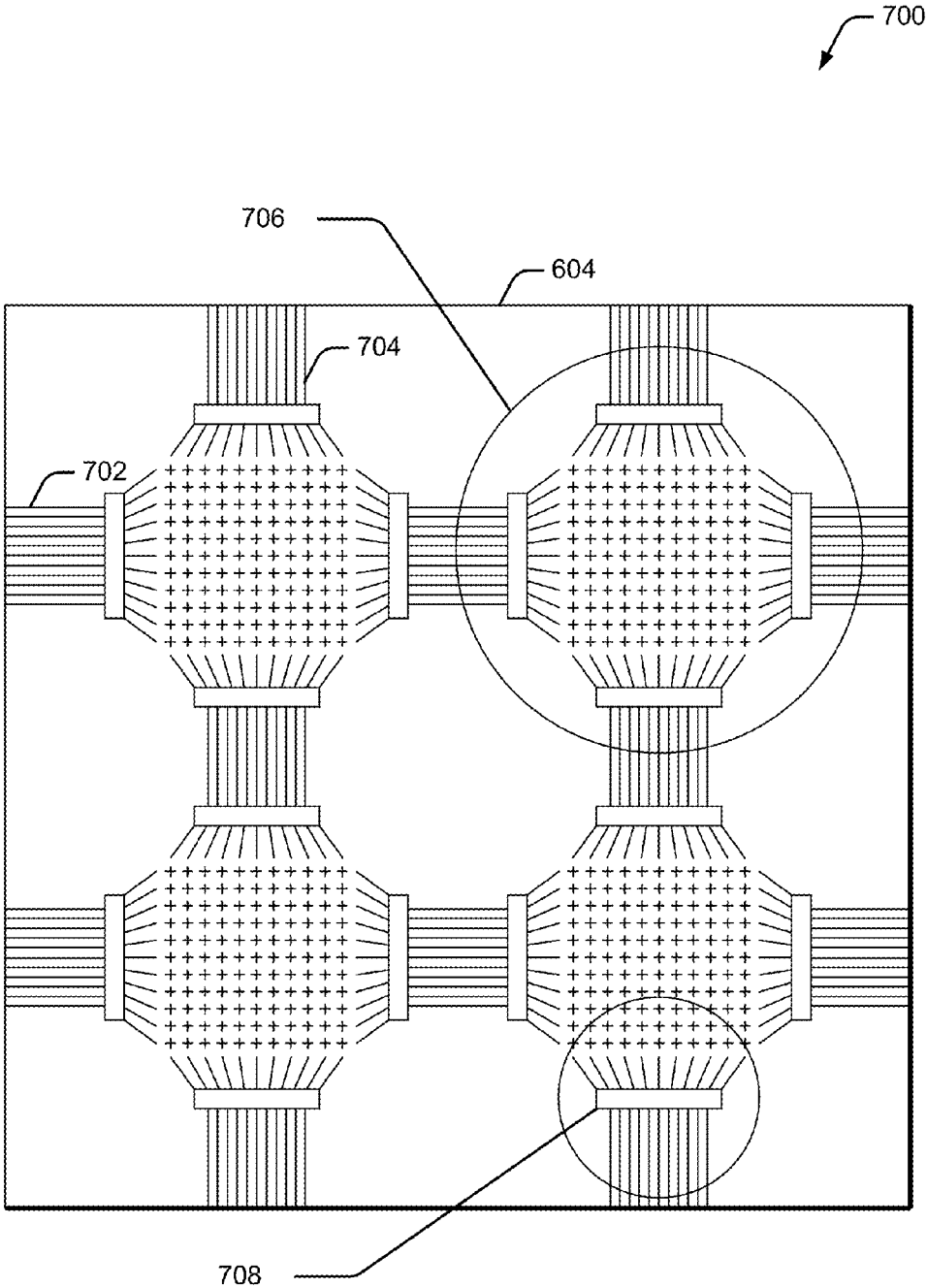


Fig. 7

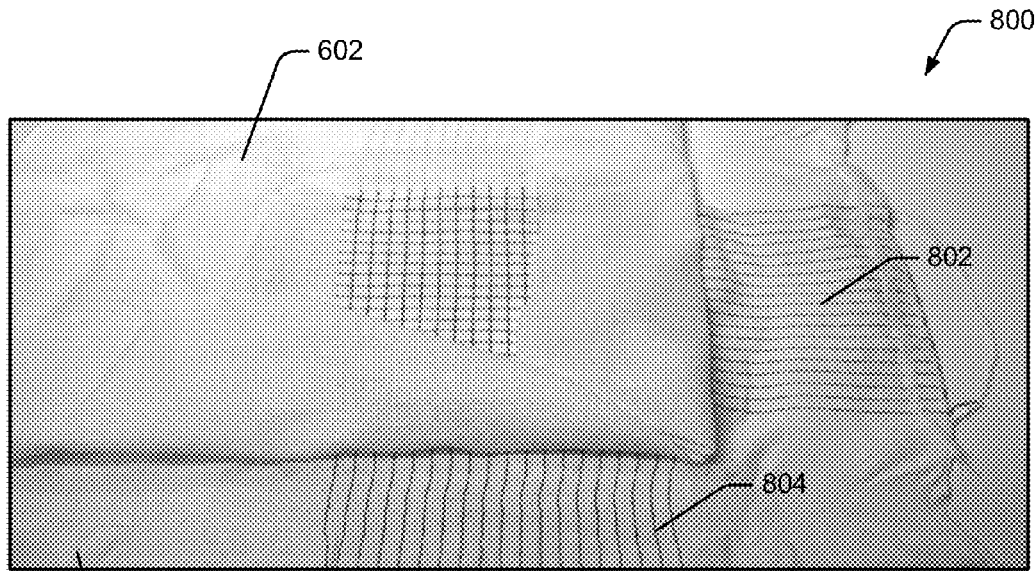


Fig. 8

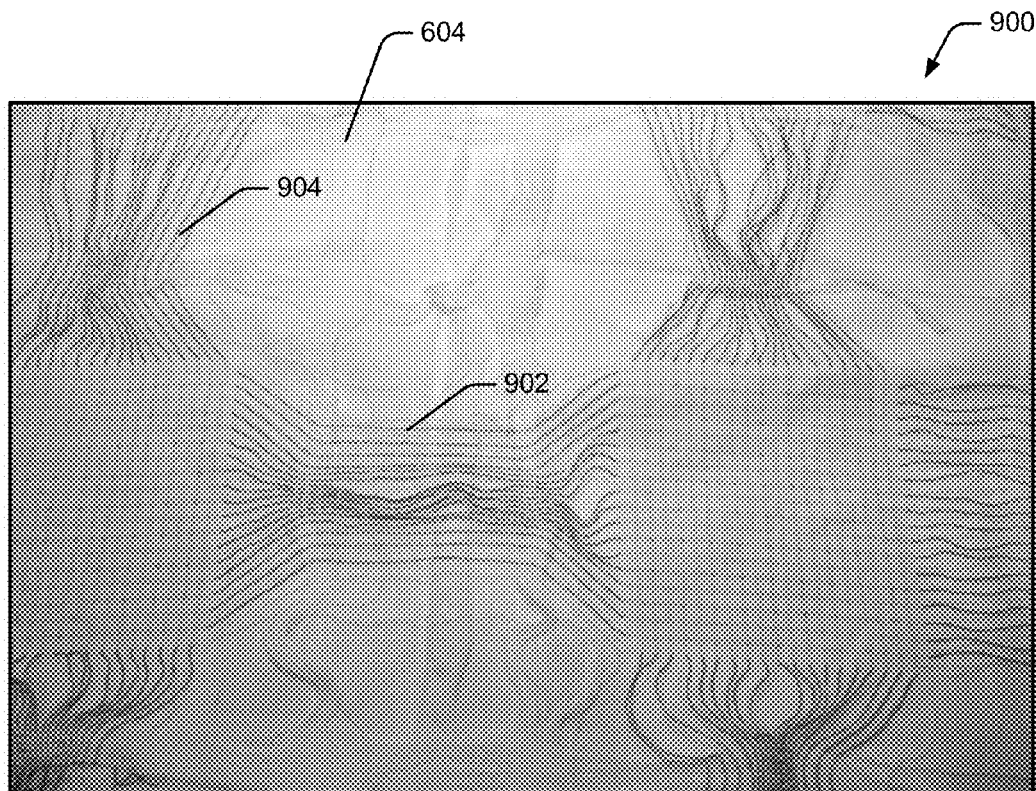


Fig. 9

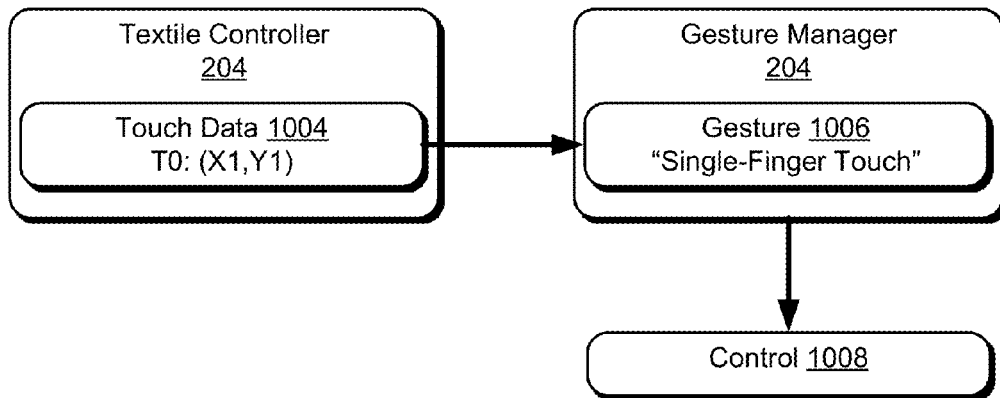
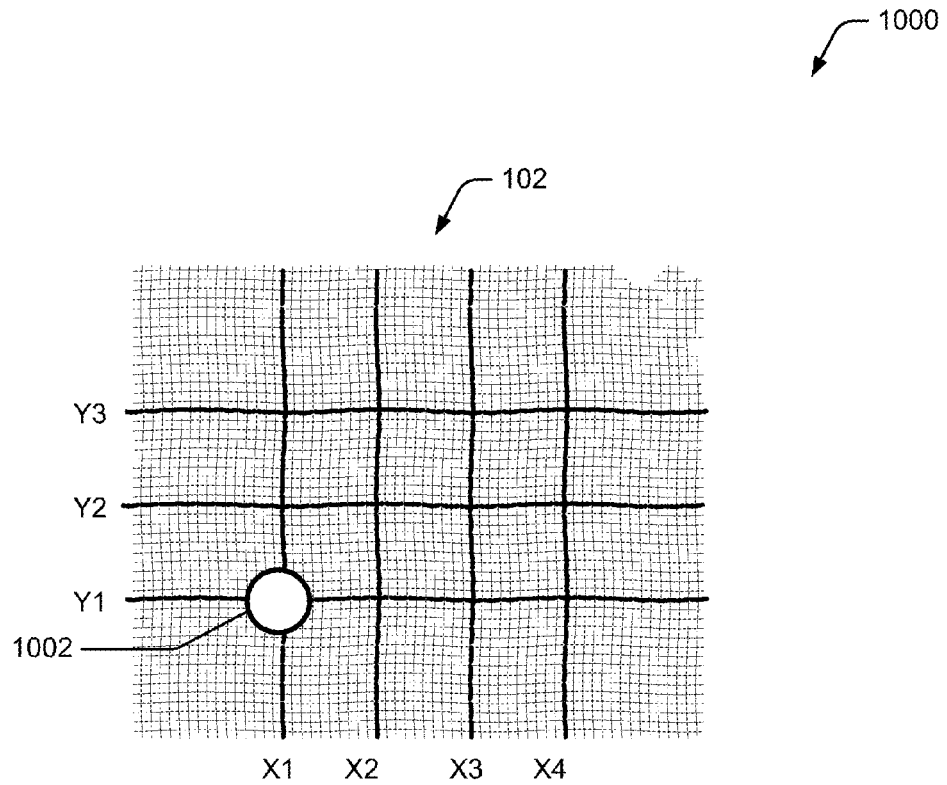


Fig. 10A

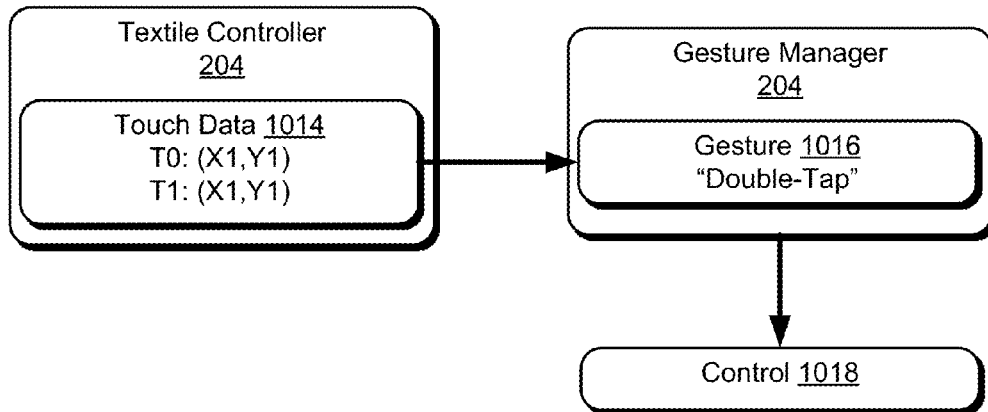
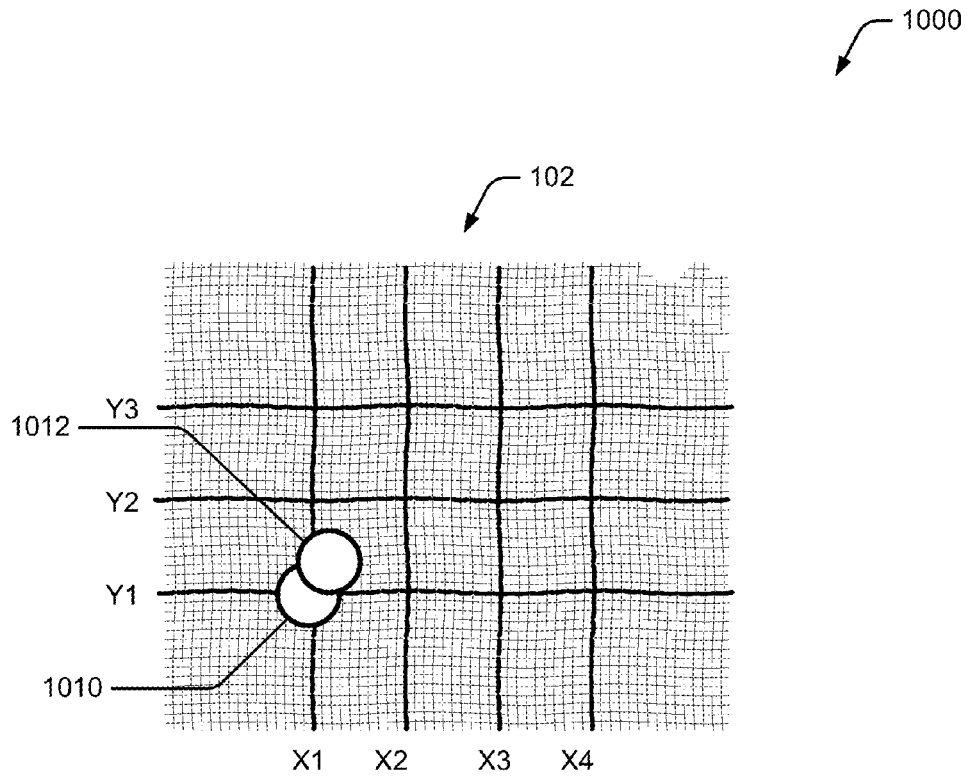


Fig. 10B

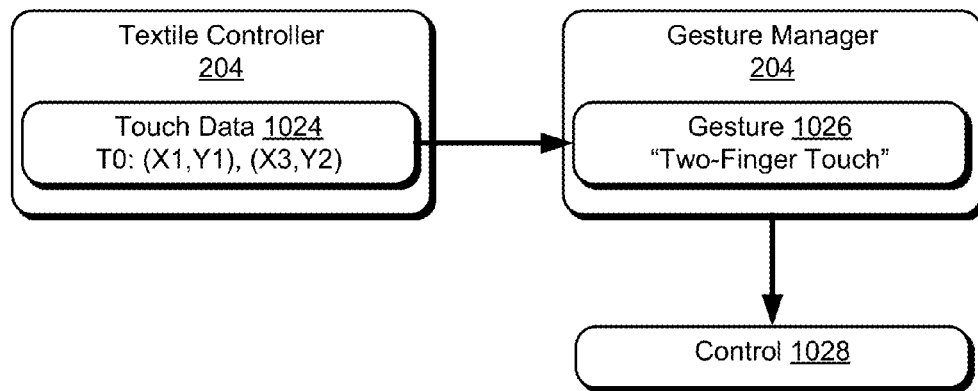
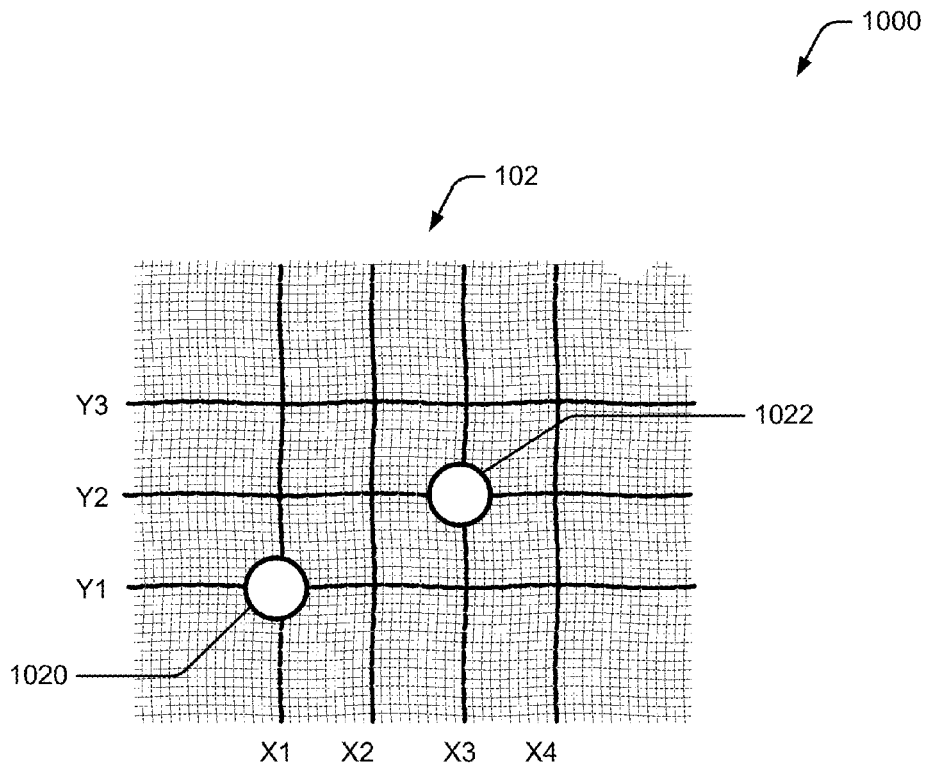


Fig. 10C

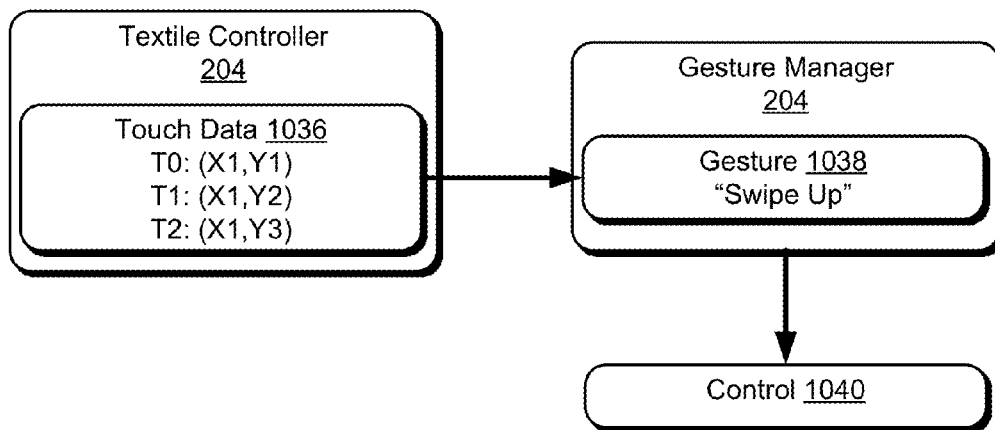
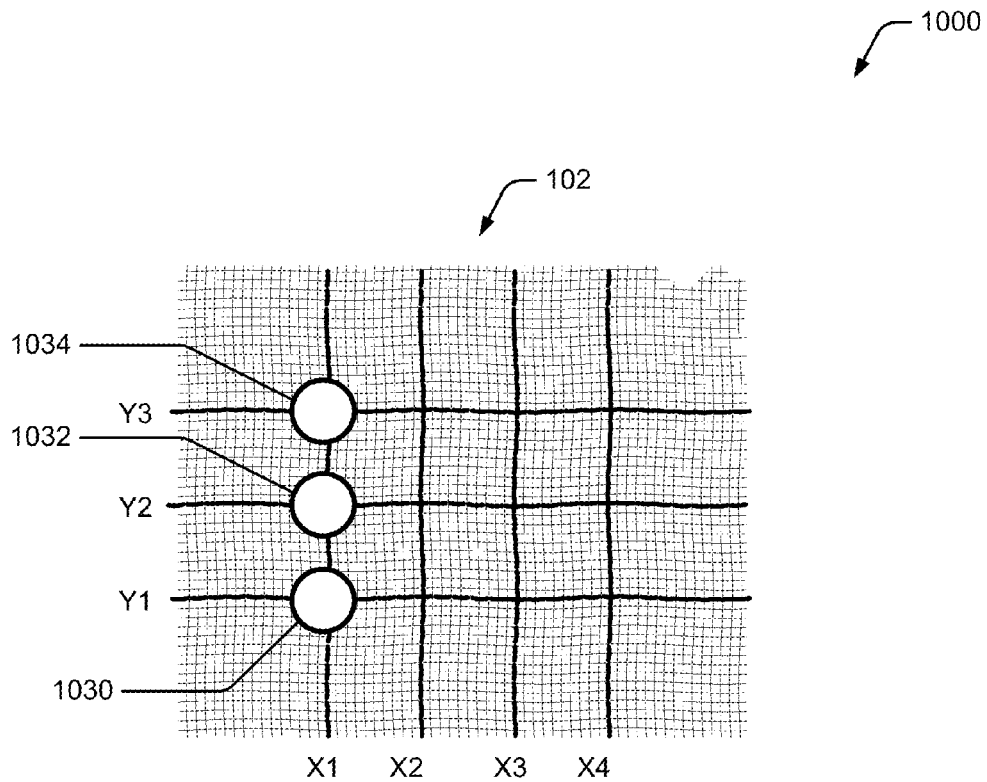


Fig. 10D

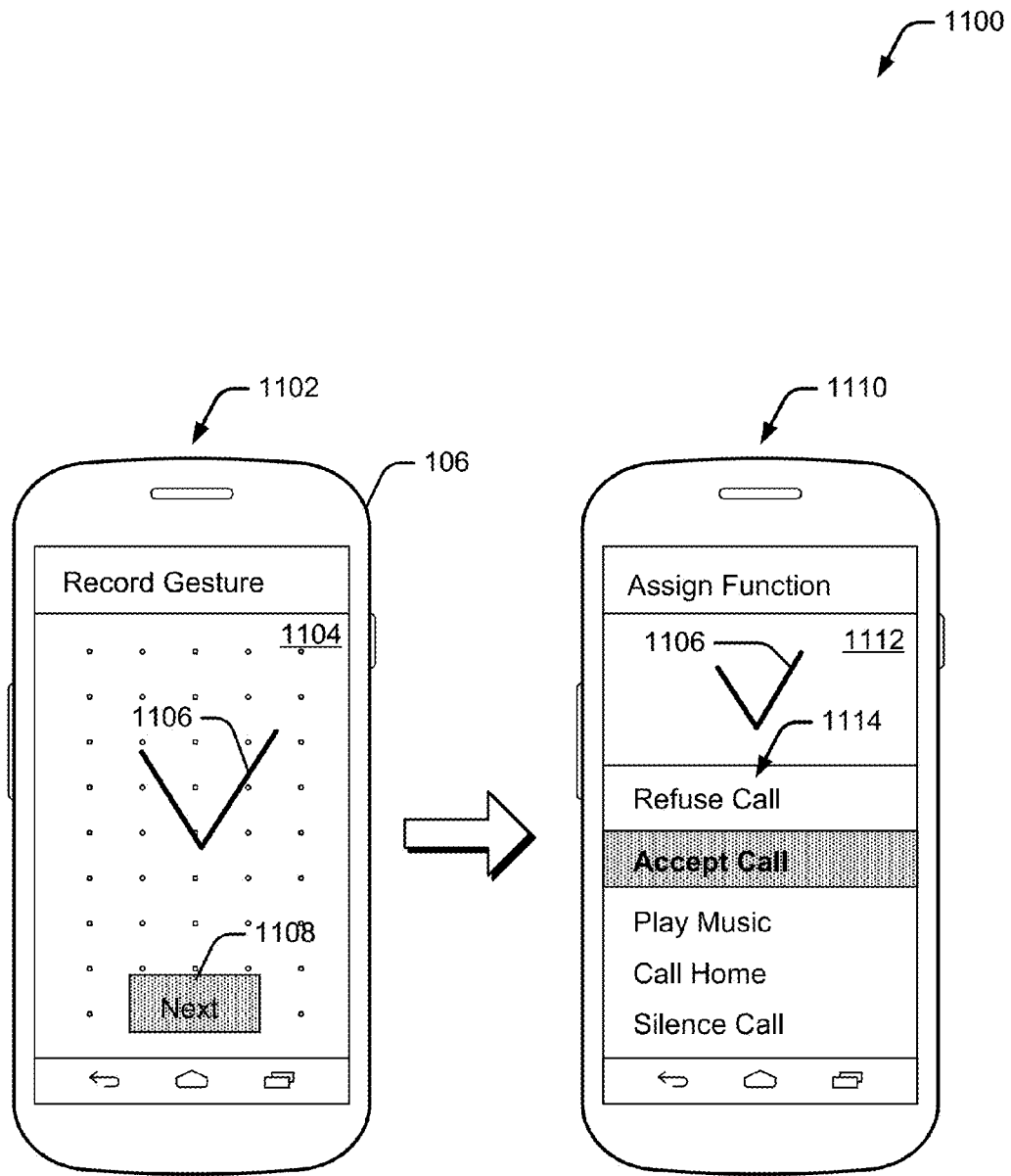


Fig. 11

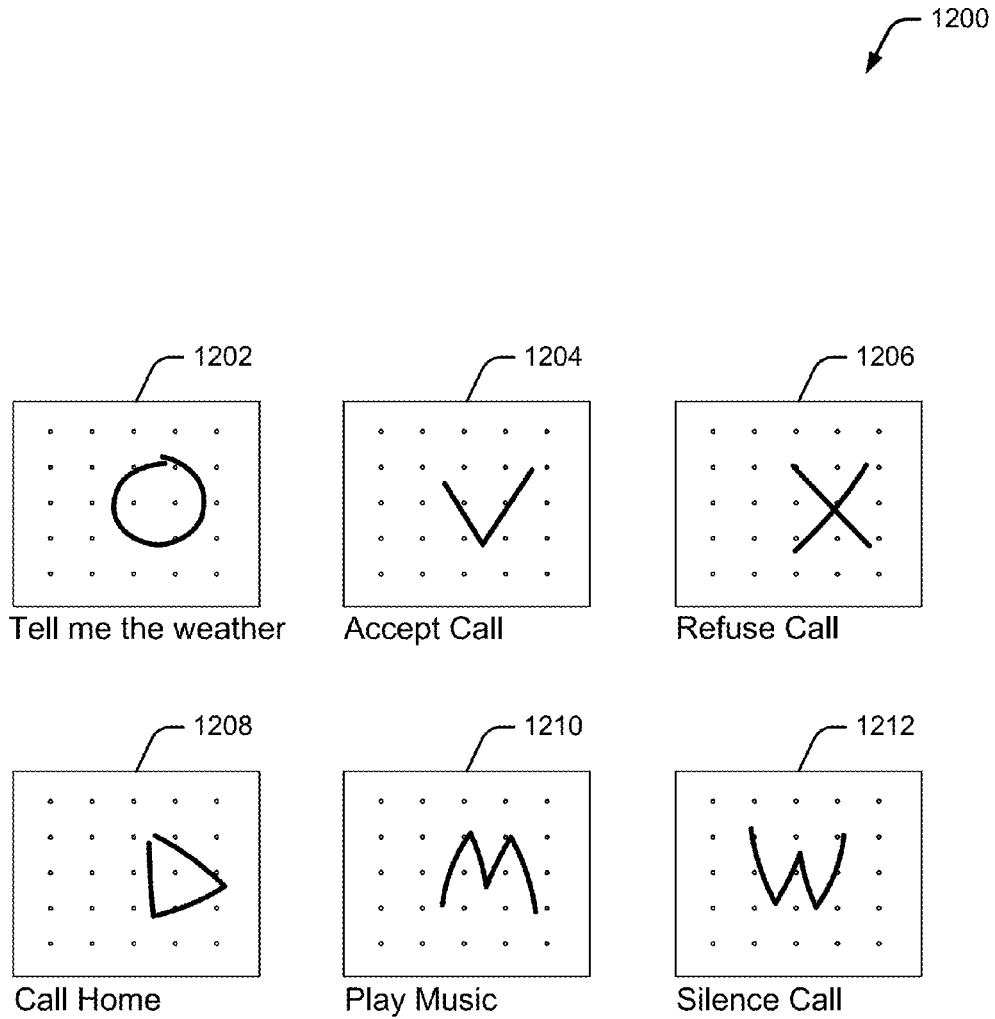


Fig. 12

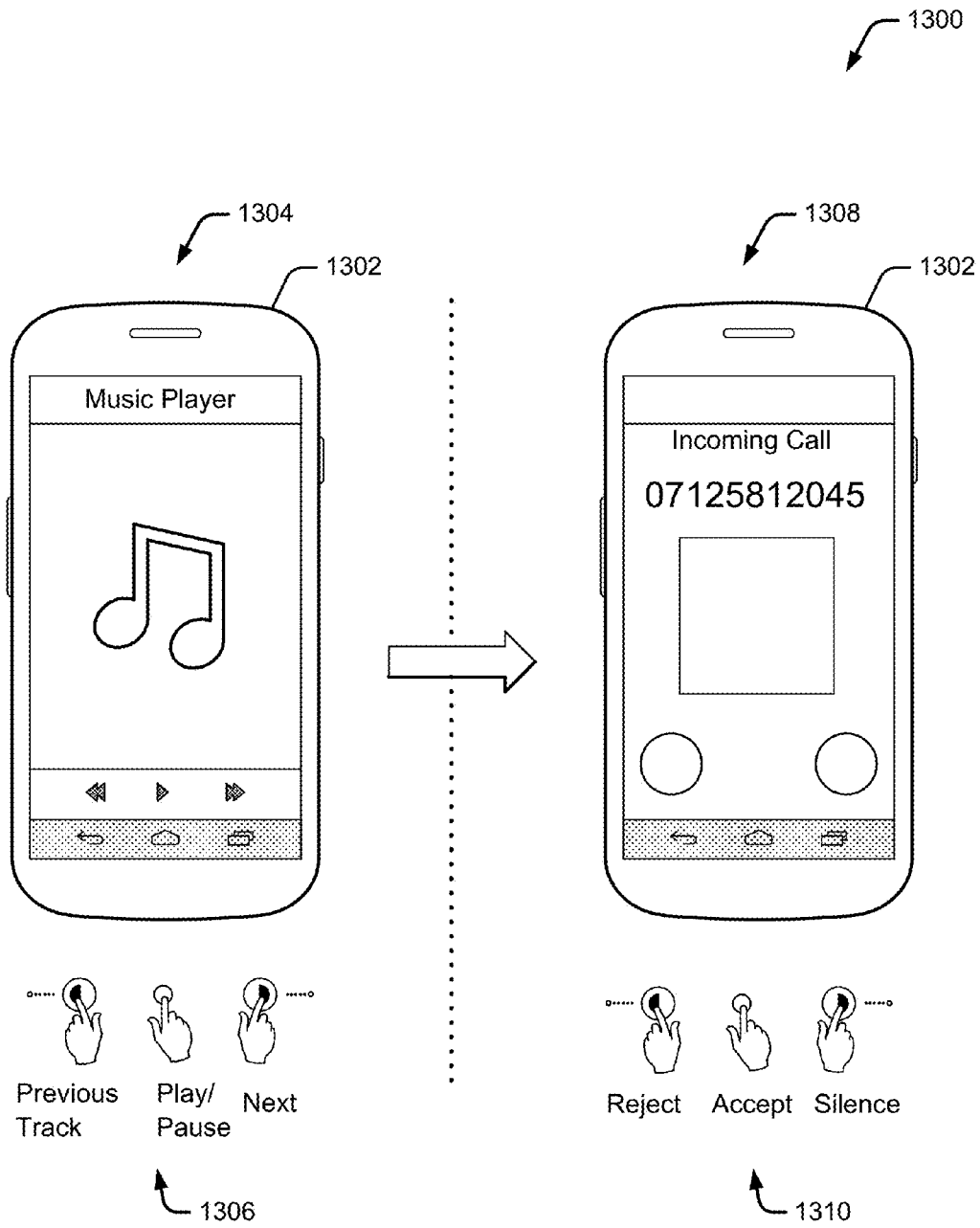


Fig. 13

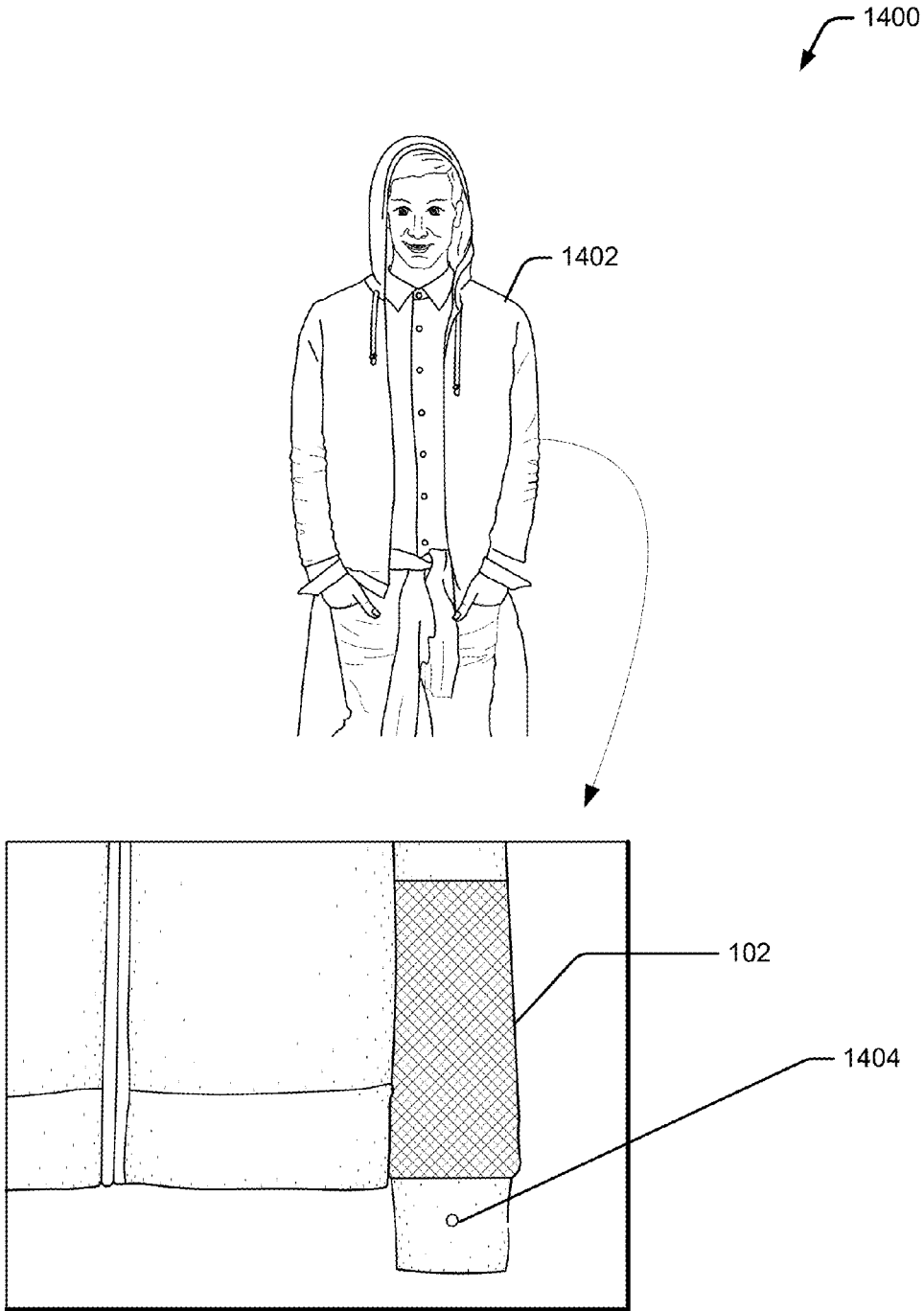


Fig. 14

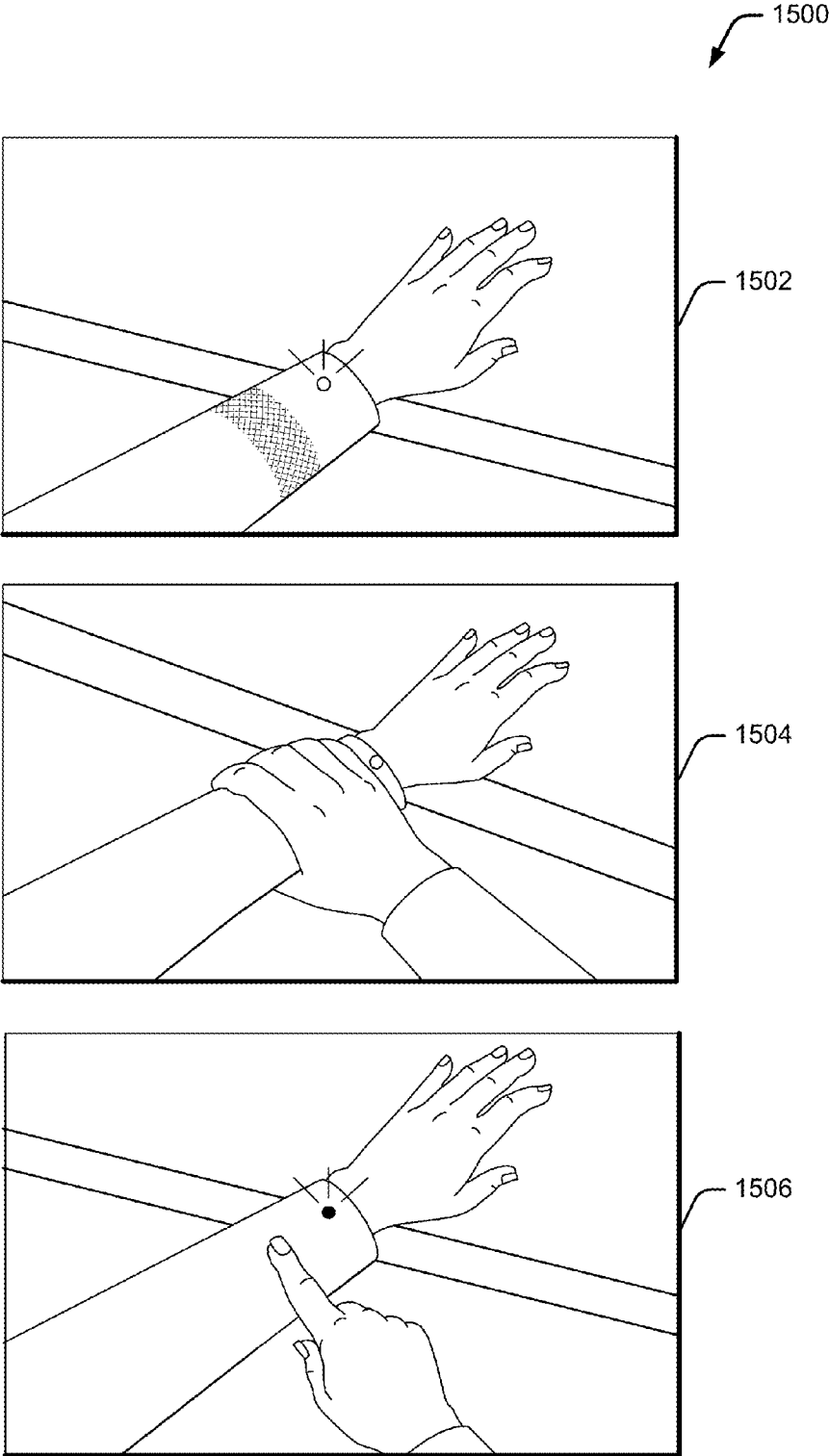


Fig. 15

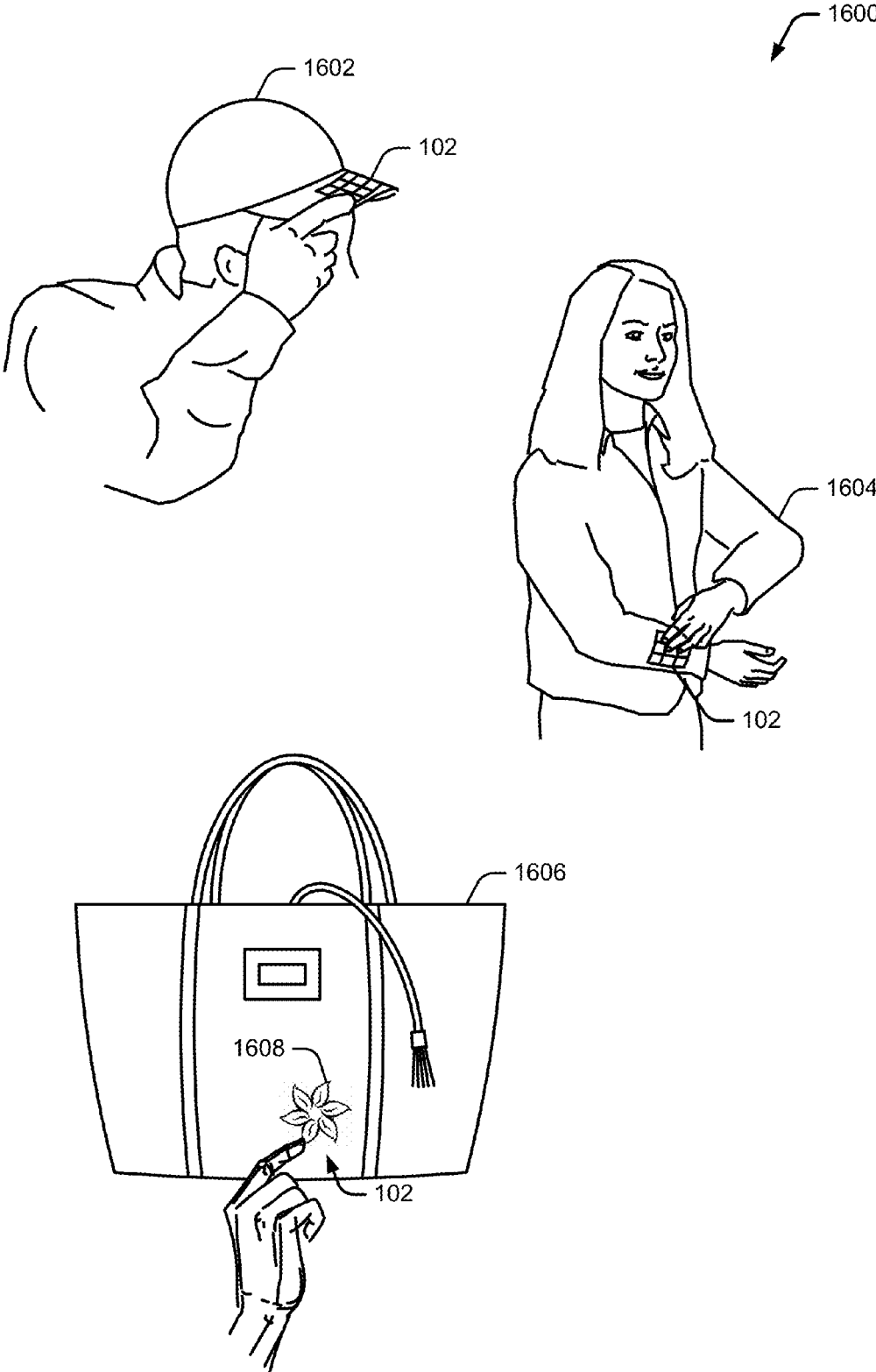


Fig. 16

1700

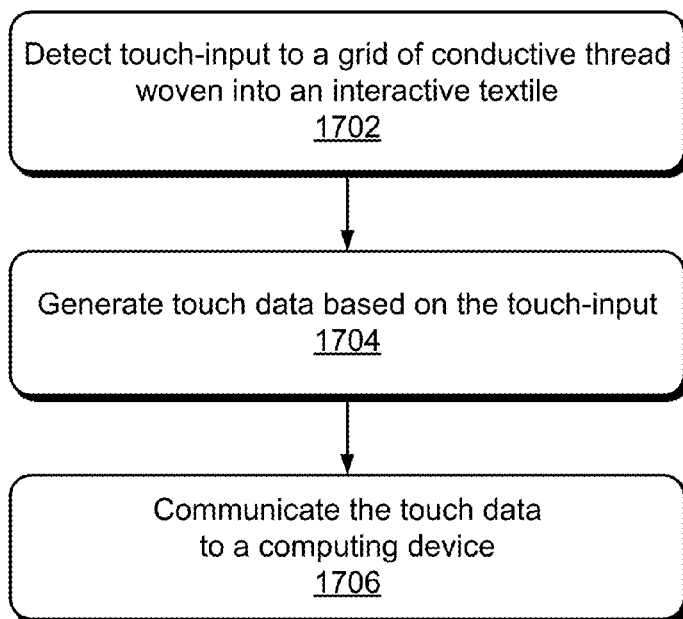
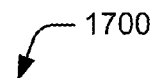


Fig. 17

1800
↙

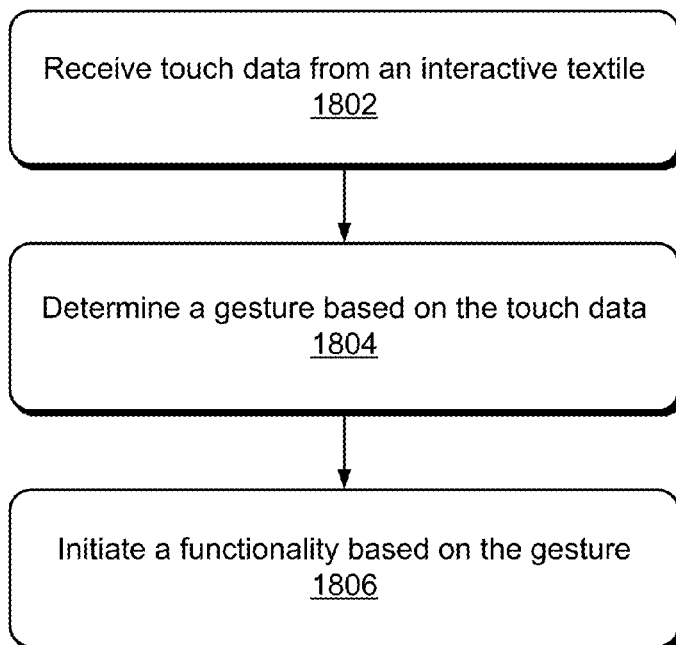


Fig. 18

1900

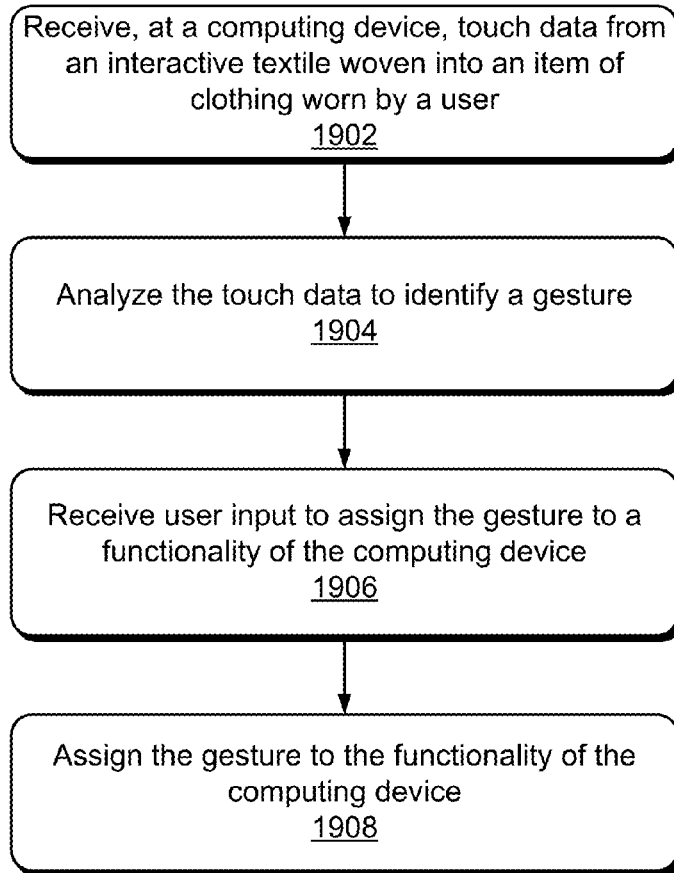



Fig. 19

2000

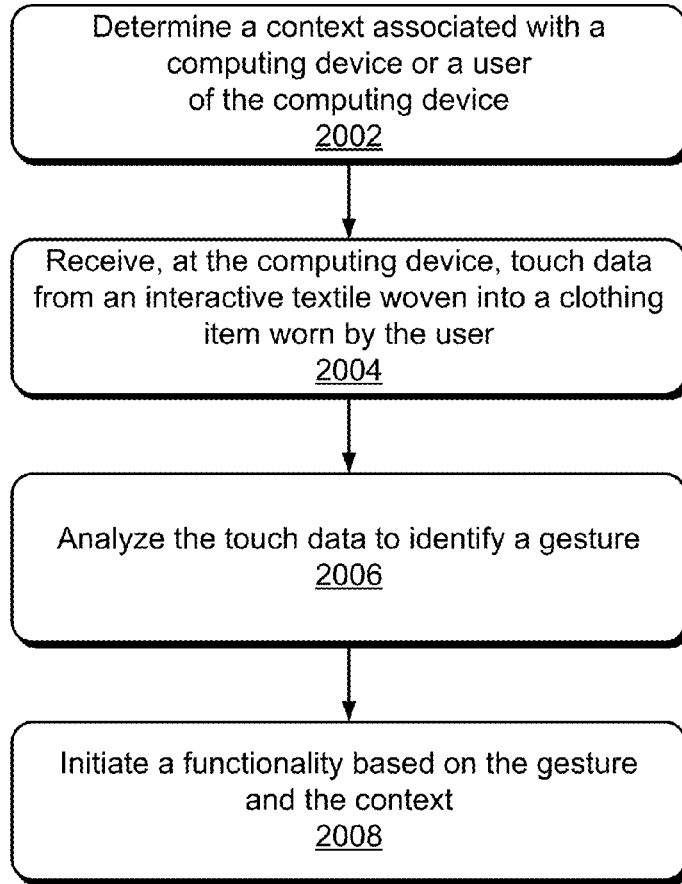



Fig. 20

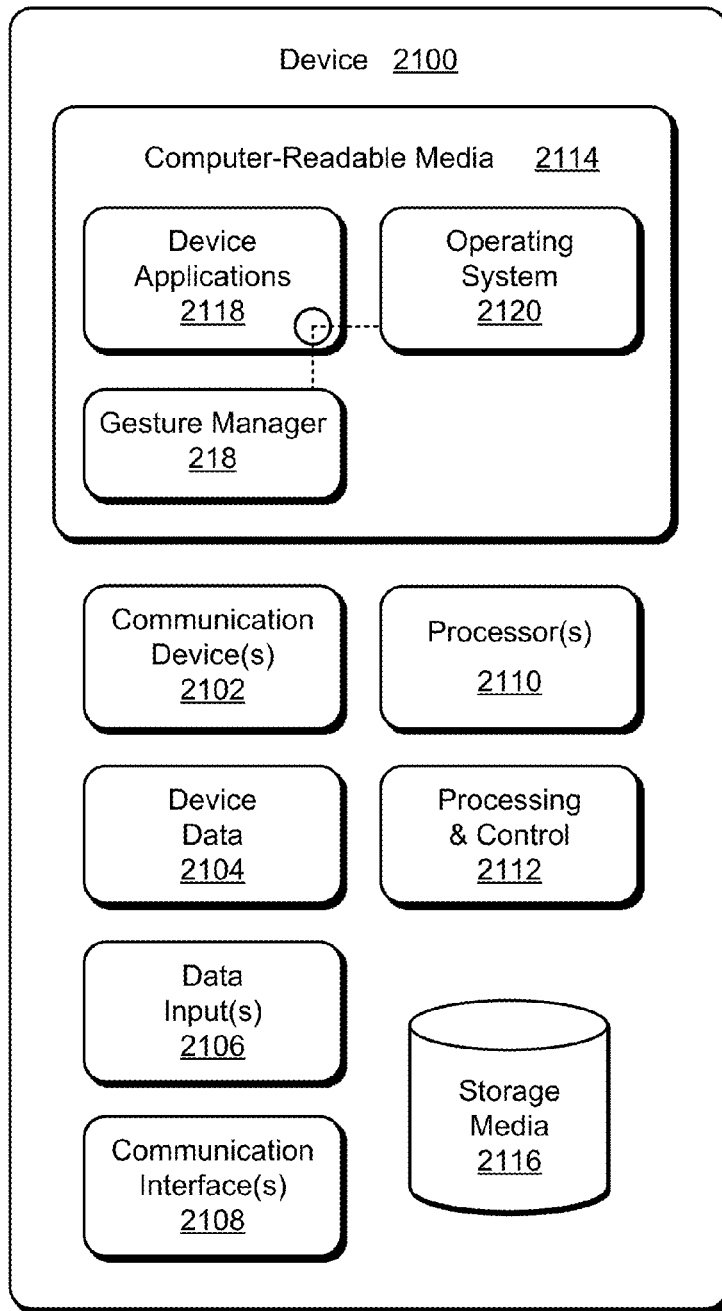


Fig. 21

TWO-LAYER INTERACTIVE TEXTILES

PRIORITY APPLICATION

[0001] This application is a non-provisional of and claims priority under 35 U.S.C. §119(e) to U.S. patent application Ser. No. 62/138,831 titled “Two-Layer Interactive Textiles,” filed Mar. 26, 2015, the disclosure of which is incorporated by reference herein in its entirety.

BACKGROUND

[0002] Currently, producing touch sensors can be complicated and expensive, especially if the touch sensor is intended to be light, flexible, or adaptive to various different kinds of use. Conventional touch pads, for example, are generally non-flexible and relatively costly to produce and to integrate into objects.

SUMMARY

[0003] This document describes two-layer interactive textiles. An interactive textile includes a grid of conductive thread woven into the interactive textile to form a capacitive touch sensor that is configured to detect touch-input. The interactive textile can process the touch-input to generate touch data that is useable to initiate functionality at various remote devices that are wirelessly coupled to the interactive textile. For example, the interactive textile may aid users in controlling volume on a stereo, pausing a movie playing on a television, or selecting a webpage on a desktop computer. Due to the flexibility of textiles, the interactive textile may be easily integrated within flexible objects, such as clothing, handbags, fabric casings, hats, and so forth.

[0004] In one or more implementations, the interactive textile includes a top textile layer and a bottom textile layer. Conductive threads are woven into the top textile layer and the bottom textile layer. When the top textile layer is combined with the bottom textile layer, the conductive threads from each layer form a capacitive touch sensor that is configured to detect touch-input. The bottom textile layer is not visible and couples the capacitive touch sensor to electronic components, such as a controller, a wireless interface, an output device (e.g., an LED, a display, or speaker), and so forth.

[0005] In one or more implementations, the conductive thread of the interactive textile includes a conductive core that includes at least one conductive wire and a cover layer constructed from flexible threads that covers the conductive core. The conductive core may be formed by twisting one or more flexible threads (e.g., silk threads, polyester threads, or cotton threads) with the conductive wire, or by wrapping flexible threads around the conductive wire. In one or more implementations, the conductive core is formed by braiding the conductive wire with flexible threads (e.g., silk). The cover layer may be formed by wrapping or braiding flexible threads around the conductive core. In one or more implementations, the conductive thread is implemented with a “double-braided” structure in which the conductive core is formed by braiding flexible threads with a conductive wire, and then braiding flexible threads around the braided conductive core.

[0006] In one or more implementations, a gesture manager is implemented at a computing device that is wirelessly coupled to the interactive textile. The gesture manager enables the user to create gestures and assign the gestures to various functionalities of the computing device. The gesture

manager can store mappings between the created gestures and the functionalities in a gesture library to enable the user to initiate a functionality, at a subsequent time, by inputting a gesture assigned to the functionality into the interactive textile.

[0007] In one or more implementations, the gesture manager is configured to select a functionality based on both a gesture to the interactive textile and a context of the computing device. The ability to recognize gestures based on context enables the user to invoke a variety of different functionalities using a subset of gestures. For example, for a first context, a first gesture may initiate a first functionality, whereas for a second context, the same first gesture may initiate a second functionality.

[0008] In one or more implementations, the interactive textile is coupled to one or more output devices (e.g., a light source, a speaker, or a display) that is integrated within the flexible object. The output device can be controlled to provide notifications initiated from the computing device and/or feedback to the user based on the user’s interactions with the interactive textile.

[0009] This summary is provided to introduce simplified concepts concerning two-layer interactive textiles, which is further described below in the Detailed Description. This summary is not intended to identify essential features of the claimed subject matter, nor is it intended for use in determining the scope of the claimed subject matter.

BRIEF DESCRIPTION OF THE DRAWINGS

[0010] Embodiments of techniques and devices for two-layer interactive textiles are described with reference to the following drawings. The same numbers are used throughout the drawings to reference like features and components:

[0011] FIG. 1 is an illustration of an example environment in which techniques using, and an objects including, an interactive textile may be embodied.

[0012] FIG. 2 illustrates an example system that includes an interactive textile and a gesture manager.

[0013] FIG. 3 illustrates an example of an interactive textile in accordance with one or more implementations.

[0014] FIG. 4a which illustrates an example of a conductive core for a conductive thread in accordance with one or more implementations.

[0015] FIG. 4b which illustrates an example of a conductive thread that includes a cover layer formed by wrapping flexible threads around a conductive core.

[0016] FIG. 5 illustrates an example of an interactive textile with multiple textile layers.

[0017] FIG. 6 illustrates an example of a two-layer interactive textile in accordance with one or more implementations.

[0018] FIG. 7 illustrates a more-detailed view of a second textile layer of a two-layer interactive textile in accordance with one or more implementations.

[0019] FIG. 8 illustrates an example of a second textile layer of a two-layer interactive textile in accordance with one or more implementations.

[0020] FIG. 9 illustrates an additional example of a second textile layer of a two-layer interactive textile in accordance with one or more implementations.

[0021] FIG. 10A illustrates an example of generating a control based on touch-input corresponding to a single-finger touch.

[0022] FIG. 10B illustrates an example of generating a control based on touch-input corresponding to a double-tap.

[0023] FIG. 10C illustrates an example of generating a control based on touch-input corresponding to a two-finger touch.

[0024] FIG. 10D illustrates an example of generating a control based on touch-input corresponding to a swipe up.

[0025] FIG. 11 illustrates an example of creating and assigning gestures to functionality of a computing device in accordance with one or more implementations.

[0026] FIG. 12 illustrates an example of a gesture library in accordance with one or more implementations.

[0027] FIG. 13 illustrates an example of contextual-based gestures to an interactive textile in accordance with one or more implementations.

[0028] FIG. 14 illustrates an example of an interactive textile that includes an output device in accordance with one or more implementations.

[0029] FIG. 15 illustrates implementation examples 1500 of interacting with an interactive textile and an output device in accordance with one or more implementations.

[0030] FIG. 16 illustrates various examples of interactive textiles integrated within flexible objects.

[0031] FIG. 17 illustrates an example method of generating touch data using an interactive textile.

[0032] FIG. 18 illustrates an example method of determining gestures usable to initiate functionality of a computing device in accordance with one or more implementations.

[0033] FIG. 19 illustrates an example method 1900 of assigning a gesture to a functionality of a computing device in accordance with one or more implementations.

[0034] FIG. 20 illustrates an example method 2300 of initiating a functionality of a computing device based on a gesture and a context in accordance with one or more implementations.

[0035] FIG. 21 illustrates various components of an example computing system that can be implemented as any type of client, server, and/or computing device as described with reference to the previous FIGS. 1-20 to implement two-layer interactive textiles.

DETAILED DESCRIPTION

Overview

[0036] Currently, producing touch sensors can be complicated and expensive, especially if the touch sensor is intended to be light, flexible, or adaptive to various different kinds of use. This document describes techniques using, and objects embodying, interactive textiles which are configured to sense multi-touch-input. To enable the interactive textiles to sense multi-touch-input, a grid of conductive thread is woven into the interactive textile to form a capacitive touch sensor that can detect touch-input. The interactive textile can process the touch-input to generate touch data that is useable to initiate functionality at various remote devices. For example, the interactive textiles may aid users in controlling volume on a stereo, pausing a movie playing on a television, or selecting a webpage on a desktop computer. Due to the flexibility of textiles, the interactive textile may be easily integrated within flexible objects, such as clothing, handbags, fabric casings, hats, and so forth.

[0037] In one or more implementations, the interactive textile includes a top textile layer and a bottom textile layer. Conductive threads are woven into the top textile layer and the bottom textile layer. When the top textile layer is combined with the bottom textile layer, the conductive threads

from each layer form a capacitive touch sensor that is configured to detect touch-input. The bottom textile layer is not visible and couples the capacitive through sensor to electronic components, such as a controller, a wireless interface, an output device (e.g., an LED, a display, or speaker), and so forth.

[0038] In one or more implementations, the conductive thread of the interactive textile includes a conductive core that includes at least one conductive wire and a cover layer constructed from flexible threads that covers the conductive core. The conductive core may be formed by twisting one or more flexible threads (e.g., silk threads, polyester threads, or cotton threads) with the conductive wire, or by wrapping flexible threads around the conductive wire. In one or more implementations, the conductive core is formed by braiding the conductive wire with flexible threads (e.g., silk). The cover layer may be formed by wrapping or braiding flexible threads around the conductive core. In one or more implementations, the conductive thread is implemented with a “double-braided” structure in which the conductive core is formed by braiding flexible threads with a conductive wire, and then braiding flexible threads around the braided conductive core.

[0039] In one or more implementations, a gesture manager is implemented at a computing device that is wirelessly coupled to the interactive textile. The gesture manager enables the user to create gestures and assign the gestures to various functionalities of the computing device. The gesture manager can store mappings between the created gestures and the functionalities in a gesture library to enable the user to initiate a functionality, at a subsequent time, by inputting a gesture assigned to the functionality into the interactive textile.

[0040] In one or more implementations, the gesture manager is configured to select a functionality based on both a gesture to the interactive textile and a context of the computing device. The ability to recognize gestures based on context enables the user to invoke a variety of different functionalities using a subset of gestures. For example, for a first context, a first gesture may initiate a first functionality, whereas for a second context, the same first gesture may initiate a second functionality.

[0041] In one or more implementations, the interactive textile is coupled to one or more output devices (e.g., a light source, a speaker, or a display) that is integrated within the flexible object. The output device can be controlled to provide notifications initiated from the computing device and/or feedback to the user based on the user’s interactions with the interactive textile.

Example Environment

[0042] FIG. 1 is an illustration of an example environment 100 in which techniques using, and objects including, an interactive textile may be embodied. Environment 100 includes an interactive textile 102, which is shown as being integrated within various objects 104. Interactive textile 102 is a textile that is configured to sense multi-touch input. As described herein, a textile corresponds to any type of flexible woven material consisting of a network of natural or artificial fibers, often referred to as thread or yarn. Textiles may be formed by weaving, knitting, crocheting, knotting, or pressing threads together.

[0043] In environment 100, objects 104 include “flexible” objects, such as a shirt 104-1, a hat 104-2, and a handbag 104-3. It is to be noted, however, that interactive textile 102

may be integrated within any type of flexible object made from fabric or a similar flexible material, such as articles of clothing, blankets, shower curtains, towels, sheets, bed spreads, or fabric casings of furniture, to name just a few. As discussed in more detail below, interactive textile **102** may be integrated within flexible objects **104** in a variety of different ways, including weaving, sewing, gluing, and so forth.

[0044] In this example, objects **104** further include “hard” objects, such as a plastic cup **104-4** and a hard smart phone casing **104-5**. It is to be noted, however, that hard objects **104** may include any type of “hard” or “rigid” object made from non-flexible or semi-flexible materials, such as plastic, metal, aluminum, and so on. For example, hard objects **104** may also include plastic chairs, water bottles, plastic balls, or car parts, to name just a few. Interactive textile **102** may be integrated within hard objects **104** using a variety of different manufacturing processes. In one or more implementations, injection molding is used to integrate interactive textiles **102** into hard objects **104**.

[0045] Interactive textile **102** enables a user to control object **104** that the interactive textile **102** is integrated with, or to control a variety of other computing devices **106** via a network **108**. Computing devices **106** are illustrated with various non-limiting example devices: server **106-1**, smart phone **106-2**, laptop **106-3**, computing spectacles **106-4**, television **106-5**, camera **106-6**, tablet **106-7**, desktop **106-8**, and smart watch **106-9**, though other devices may also be used, such as home automation and control systems, sound or entertainment systems, home appliances, security systems, networks, and e-readers. Note that computing device **106** can be wearable (e.g., computing spectacles and smart watches), non-wearable but mobile (e.g., laptops and tablets), or relatively immobile (e.g., desktops and servers).

[0046] Network **108** includes one or more of many types of wireless or partly wireless communication networks, such as a local-area-network (LAN), a wireless local-area-network (WLAN), a personal-area-network (PAN), a wide-area-network (WAN), an intranet, the Internet, a peer-to-peer network, point-to-point network, a mesh network, and so forth.

[0047] Interactive textile **102** can interact with computing devices **106** by transmitting touch data through network **108**. Computing device **106** uses the touch data to control computing device **106** or applications at computing device **106**. As an example, consider that interactive textile **102** integrated at shirt **104-1** may be configured to control the user’s smart phone **106-2** in the user’s pocket, television **106-5** in the user’s home, smart watch **106-9** on the user’s wrist, or various other appliances in the user’s house, such as thermostats, lights, music, and so forth. For example, the user may be able to swipe up or down on interactive textile **102** integrated within the user’s shirt **104-1** to cause the volume on television **106-5** to go up or down, to cause the temperature controlled by a thermostat in the user’s house to increase or decrease, or to turn on and off lights in the user’s house. Note that any type of touch, tap, swipe, hold, or stroke gesture may be recognized by interactive textile **102**.

[0048] In more detail, consider FIG. 2 which illustrates an example system **200** that includes an interactive textile and a gesture manager. In system **200**, interactive textile **102** is integrated in an object **104**, which may be implemented as a flexible object (e.g., shirt **104-1**, hat **104-2**, or handbag **104-3**) or a hard object (e.g., plastic cup **104-4** or smart phone casing **104-5**).

[0049] Interactive textile **102** is configured to sense multi-touch-input from a user when one or more fingers of the user’s hand touch interactive textile **102**. Interactive textile **102** may also be configured to sense full-hand touch input from a user, such as when an entire hand of the user touches or swipes interactive textile **102**. To enable this, interactive textile **102** includes a capacitive touch sensor **202**, a textile controller **204**, and a power source **206**.

[0050] Capacitive touch sensor **202** is configured to sense touch-input when an object, such as a user’s finger, hand, or a conductive stylus, approaches or makes contact with capacitive touch sensor **202**. Unlike conventional hard touch pads, capacitive touch sensor **202** uses a grid of conductive thread **208** woven into interactive textile **102** to sense touch-input. Thus, capacitive touch sensor **202** does not alter the flexibility of interactive textile **102**, which enables interactive textile **102** to be easily integrated within objects **104**.

[0051] Power source **206** is coupled to textile controller **204** to provide power to textile controller **204**, and may be implemented as a small battery. Textile controller **204** is coupled to capacitive touch sensor **202**. For example, wires from the grid of conductive threads **208** may be connected to textile controller **204** using flexible PCB, creping, gluing with conductive glue, soldering, and so forth.

[0052] In one or more implementations, interactive textile **102** (or object **104**) may also include one or more output devices, such as light sources (e.g., LED’s), displays, or speakers. In this case, the output devices may also be connected to textile controller **204** to enable textile controller **204** to control their output.

[0053] Textile controller **204** is implemented with circuitry that is configured to detect the location of the touch-input on the grid of conductive thread **208**, as well as motion of the touch-input. When an object, such as a user’s finger, touches capacitive touch sensor **202**, the position of the touch can be determined by controller **204** by detecting a change in capacitance on the grid of conductive thread **208**. Textile controller **204** uses the touch-input to generate touch data usable to control computing device **102**. For example, the touch-input can be used to determine various gestures, such as single-finger touches (e.g., touches, taps, and holds), multi-finger touches (e.g., two-finger touches, two-finger taps, two-finger holds, and pinches), single-finger and multi-finger swipes (e.g., swipe up, swipe down, swipe left, swipe right), and full-hand interactions (e.g., touching the textile with a user’s entire hand, covering textile with the user’s entire hand, pressing the textile with the user’s entire hand, palm touches, and rolling, twisting, or rotating the user’s hand while touching the textile). Capacitive touch sensor **202** may be implemented as a self-capacitance sensor, or a projective capacitance sensor, which is discussed in more detail below.

[0054] Object **104** may also include network interfaces **210** for communicating data, such as touch data, over wired, wireless, or optical networks to computing devices **106**. By way of example and not limitation, network interfaces **210** may communicate data over a local-area-network (LAN), a wireless local-area-network (WLAN), a personal-area-network (PAN) (e.g., Bluetooth™), a wide-area-network (WAN), an intranet, the Internet, a peer-to-peer network, point-to-point network, a mesh network, and the like (e.g., through network **108** of FIG. 1).

[0055] In this example, computing device **106** includes one or more computer processors **212** and computer-readable storage media (storage media) **214**. Storage media **214**

includes applications 216 and/or an operating system (not shown) embodied as computer-readable instructions executable by computer processors 212 to provide, in some cases, functionalities described herein. Storage media 214 also includes a gesture manager 218 (described below).

[0056] Computing device 106 may also include a display 220 and network interfaces 222 for communicating data over wired, wireless, or optical networks. For example, network interfaces 222 can receive touch data sensed by interactive textile 102 from network interfaces 210 of object 104. By way of example and not limitation, network interface 222 may communicate data over a local-area-network (LAN), a wireless local-area-network (WLAN), a personal-area-network (PAN) (e.g., Bluetooth™), a wide-area-network (WAN), an intranet, the Internet, a peer-to-peer network, point-to-point network, a mesh network, and the like.

[0057] Gesture manager 218 is capable of interacting with applications 216 and interactive textile 102 effective to activate various functionalities associated with computing device 106 and/or applications 216 through touch-input (e.g., gestures) received by interactive textile 102. Gesture manager 218 may be implemented at a computing device 106 that is local to object 104, or remote from object 104.

[0058] Having discussed a system in which interactive textile 102 can be implemented, now consider a more-detailed discussion of interactive textile 102.

[0059] FIG. 3 illustrates an example 300 of interactive textile 102 in accordance with one or more implementations. In this example, interactive textile 102 includes non-conductive threads 302 woven with conductive threads 208 to form interactive textile 102. Non-conductive threads 302 may correspond to any type of non-conductive thread, fiber, or fabric, such as cotton, wool, silk, nylon, polyester, and so forth.

[0060] At 304, a zoomed-in view of conductive thread 208 is illustrated. Conductive thread 208 includes a conductive wire 306 twisted with a flexible thread 308. Twisting conductive wire 306 with flexible thread 308 causes conductive thread 208 to be flexible and stretchy, which enables conductive thread 208 to be easily woven with non-conductive threads 302 to form interactive textile 102.

[0061] In one or more implementations, conductive wire 306 is a thin copper wire. It is to be noted, however, that conductive wire 306 may also be implemented using other materials, such as silver, gold, or other materials coated with a conductive polymer. Flexible thread 308 may be implemented as any type of flexible thread or fiber, such as cotton, wool, silk, nylon, polyester, and so forth.

[0062] In one or more implementations, conductive thread 208 includes a conductive core that includes at least one conductive wire 306 (e.g., one or more copper wires) and a cover layer, configured to cover the conductive core, that is constructed from flexible threads 308. In some cases, conductive wire 306 of the conductive core is insulated. Alternately, conductive wire 306 of the conductive core is not insulated.

[0063] In one or more implementations, the conductive core may be implemented using a single, straight, conductive wire 306. Alternately, the conductive core may be implemented using a conductive wire 306 and one or more flexible threads 308. For example, the conductive core may be formed by twisting one or more flexible threads 308 (e.g., silk threads, polyester threads, or cotton threads) with conductive wire 306 (e.g., as shown at 304 of FIG. 3), or by wrapping flexible threads 308 around conductive wire 306.

[0064] In one or more implementations, the conductive core includes flexible threads 308 braided with conductive wire 306. As an example, consider FIG. 4a which illustrates an example 400 of a conductive core 402 for a conductive thread in accordance with one or more implementations. In this example, conductive core 402 is formed by braiding conductive wire 306 (not pictured) with flexible threads 308. A variety of different types of flexible threads 308 may be utilized to braid with conductive wire 306, such as polyester or cotton, in order to form the conductive core.

[0065] In one or more implementations, however, silk threads are used for the braided construction of the conductive core. Silk threads are slightly twisted which enables the silk threads to “grip” or hold on to conductive wire 306. Thus, using silk threads may increase the speed at which the braided conductive core can be manufactured. In contrast, a flexible thread like polyester is slippery, and thus does not “grip” the conductive wire as well as silk. Thus, a slippery thread is more difficult to braid with the conductive wire, which may slow down the manufacturing process.

[0066] An additional benefit of using silk threads to create the braided conductive core is that silk is both thin and strong, which enables the manufacture of a thin conductive core that will not break during the interaction textile weaving process. A thin conductive core is beneficial because it enables the manufacturer to create whatever thickness they want for conductive thread 208 (e.g., thick or thin) when covering the conductive core with the second layer.

[0067] After forming the conductive core, a cover layer is constructed to cover the conductive core. In one or more implementations, the cover layer is constructed by wrapping flexible threads (e.g., polyester threads, cotton threads, wool threads, or silk threads) around the conductive core. As an example, consider FIG. 4b which illustrates an example 404 of a conductive thread that includes a cover layer formed by wrapping flexible threads around a conductive core. In this example, conductive thread 208 is formed by wrapping flexible threads 308 around the conductive core (not pictured). For example, the cover layer may be formed by wrapping polyester threads around the conductive core at approximately 1900 turns per yard.

[0068] In one or more implementations, the cover layer includes flexible threads braided around the conductive core. The braided cover layer may be formed using the same type of braiding as described above with regards to FIG. 4a. Any type of flexible thread 308 may be used for the braided cover layer. The thickness of the flexible thread and the number of flexible threads that are braided around the conductive core can be selected based on the desired thickness of conductive thread 208. For example, if conductive thread 208 is intended to be used for denim, a thicker flexible thread (e.g., cotton) and/or a greater number of flexible threads may be used to form the cover layer.

[0069] In one or more implementations, conductive thread 208 is constructed with a “double-braided” structure. In this case, the conductive core is formed by braiding flexible threads, such as silk, with a conductive wire (e.g., copper), as described above. Then, the cover layer is formed by braiding flexible threads (e.g., silk, cotton, or polyester) around the braided conductive core. The double-braided structure is strong, and thus is unlikely to break when being pulled during the weaving process. For example, when the double-braided conductive thread is pulled, the braided structure contracts and forces the braided core of copper to contract also with

makes the whole structure stronger. Further, the double-braided structure is soft and looks like normal yarn, as opposed to a cable, which is important for aesthetics and feel.

[0070] Interactive textile 102 can be formed cheaply and efficiently, using any conventional weaving process (e.g., jacquard weaving or 3D-weaving), which involves interlacing a set of longer threads (called the warp) with a set of crossing threads (called the weft). Weaving may be implemented on a frame or machine known as a loom, of which there are a number of types. Thus, a loom can weave non-conductive threads 302 with conductive threads 208 to create interactive textile 102.

[0071] In example 300, conductive thread 208 is woven into interactive textile 102 to form a grid that includes a set of substantially parallel conductive threads 208 and a second set of substantially parallel conductive threads 208 that crosses the first set of conductive threads to form the grid. In this example, the first set of conductive threads 208 are oriented horizontally and the second set of conductive threads 208 are oriented vertically, such that the first set of conductive threads 208 are positioned substantially orthogonal to the second set of conductive threads 208. It is to be appreciated, however, that conductive threads 208 may be oriented such that crossing conductive threads 208 are not orthogonal to each other. For example, in some cases crossing conductive threads 208 may form a diamond-shaped grid. While conductive threads 208 are illustrated as being spaced out from each other in FIG. 3, it is to be noted that conductive threads 208 may be weaved very closely together. For example, in some cases two or three conductive threads may be weaved closely together in each direction.

[0072] Conductive wire 306 may be insulated to prevent direct contact between crossing conductive threads 208. To do so, conductive wire 306 may be coated with a material such as enamel or nylon. Alternately, rather than insulating conductive wire 306, interactive textile may be generated with three separate textile layers to ensure that crossing conductive threads 208 do not make direct contact with each other.

[0073] Consider, for example, FIG. 5 which illustrates an example 500 of an interactive textile 102 with multiple textile layers. In example 500, interactive textile 102 includes a first textile layer 502, a second textile layer 504, and a third textile layer 506. The three textile layers may be combined (e.g., by sewing or gluing the layers together) to form interactive textile 102. In this example, first textile layer 502 includes horizontal conductive threads 208, and second textile layer 504 includes vertical conductive threads 208. Third textile layer 506 does not include any conductive threads, and is positioned between first textile layer 502 and second textile layer 504 to prevent vertical conductive threads from making direct contact with horizontal conductive threads 208.

[0074] In one or more implementations, interactive textile 102 includes a top textile layer and a bottom textile layer. The top textile layer includes conductive threads 208 woven into the top textile layer, and the bottom textile layer also includes conductive threads woven into the bottom textile layer. When the top textile layer is combined with the bottom textile layer, the conductive threads from each layer form capacitive touch sensor 202.

[0075] Consider for example, FIG. 6 which illustrates an example 600 of a two-layer interactive textile 102 in accordance with one or more implementations. In this example, interactive textile 102 includes a first textile layer 602 and a

second textile layer 604. First textile layer 602 is considered the “top textile layer” and includes first conductive threads 606 woven into first textile layer 602. Second textile layer 604 is considered the “bottom textile layer” of interactive textile 102 and includes second conductive threads 608 woven into second textile layer 604. When integrated into flexible object 104, such as a clothing item, first textile layer 602 is visible and faces the user such that the user is able to interact with first textile layer 602, while second textile layer 604 is not visible. For instance, first textile layer 602 may be part of an “outside surface” of the clothing item, while second textile layer may be the “inside surface” of the clothing item.

[0076] When first textile layer 602 and second textile layer 604 are combined, first conductive threads 606 of first textile layer 602 couples to second conductive threads 608 of second textile layer 604 to form capacitive touch sensor 202, as described above. In one or more implementations, the direction of the conductive threads changes from first textile layer 602 to second textile layer 604 to form a grid of conductive threads, as described above. For example, first conductive threads 606 in first textile layer 602 may be positioned substantially orthogonal to second conductive threads 608 in second textile layer 604 to form the grid of conductive threads.

[0077] In some cases, first conductive threads 606 may be oriented substantially horizontally and second conductive threads 608 may be oriented substantially vertically. Alternately, first conductive threads 606 may be oriented substantially vertically and second conductive threads 608 may be oriented substantially horizontally. Alternately, first conductive threads 606 may be oriented such that crossing conductive threads 608 are not orthogonal to each other. For example, in some cases crossing conductive threads 606 and 608 may form a diamond-shaped grid.

[0078] First textile layer 602 and second textile layer 604 can be formed independently, or at different times. For example, a manufacturer may weave second conductive threads 608 into second textile layer 604. A designer could then purchase second textile layer 604 with the conductive threads already woven into the second textile layer 604, and create first textile layer 602 by weaving conductive thread into a textile design. First textile layer 602 can then be combined with second textile layer 604 to form interactive textile 102.

[0079] First textile layer and second textile layer may be combined in a variety of different ways, such as by weaving, sewing, or gluing the layers together to form interactive textile 102. In one or more implementations, first textile layer 602 and second textile layer 604 are combined using a jacquard weaving process or any type of 3D-weaving process. When first textile layer 602 and second textile layer 604 are combined, the first conductive threads 606 of first textile layer 602 couple to second conductive threads 608 of second textile layer 604 to form capacitive touch sensor 202, as described above.

[0080] In one or more implementations, second textile layer 604 implements a standard configuration or pattern of second conductive threads 608. Consider, for example, FIG. 7 which illustrates a more-detailed view 700 of second textile layer 604 of two-layer interactive textile 102 in accordance with one or more implementations. In this example, second textile layer 604 includes horizontal conductive threads 702 and vertical conductive threads 704 which intersect to form multiple grids 706 of conductive thread. It is to be noted,

however, that any standard configuration may be used, such as different sizes of grids or just lines without grids. The standard configuration of second conductive threads 608 in the second level enables a precise size, shape, and placement of interactive areas anywhere on interactive textile 102. In example 700, second textile layer 604 utilizes connectors 708 to form grids 706. Connectors 708 may be configured from a harder material, such as polyester.

[0081] Second conductive threads 608 of second textile layer 604 can be connected to electronic components of interactive textile 102, such as textile controller 204, output devices (e.g., an LED, display, or speaker), and so forth. For example, second conductive threads 608 of second textile layer 604 may be connected to electronic components, such as textile controller 204, using flexible PCB, creping, gluing with conductive glue, soldering, and so forth. Since second textile layer 604 is not visible, this enables coupling to the electronics in a way that the electronics and lines running to the electronics are not visible in the clothing item or soft object.

[0082] In one or more implementations, the pitch of second conductive threads 608 in second textile layer 604 is constant. As described herein, the “pitch” of the conductive threads refers to a width of the line spacing between conductive threads. Consider, for example, FIG. 8 which illustrates an additional example 800 of second textile layer 604 in accordance with one or more implementations. In this example, first textile layer 602 is illustrated as being folded back to reveal second textile layer 604. Horizontal conductive threads 802 and vertical conductive threads 804 are completely woven into second textile layer 604. As can be seen, the distance between each of the lines does not change, and thus the pitch is considered to be constant.

[0083] Alternately, in one or more implementations, the pitch of second conductive threads 608 in second textile layer 604 is not constant. The pitch can be varied in a variety of different ways. In one or more implementations, the pitch can be changed using shrinking materials, such as heat shrinking polymers. For example, the pitch can be changed by weaving polyester or heated yarn with the conductive threads of the second textile layer.

[0084] In one or more implementations second conductive threads 608 may be partially woven into the second textile layer 604. Then, the pitch of second conductive threads 608 can be changed by weaving first textile layer 602 with second textile layer 604. Consider, for example, FIG. 9 which illustrates an additional example 900 of a second textile layer 604 in accordance with one or more implementations. In this example, horizontal conductive threads 902 and vertical conductive threads 904 are only partially woven into second textile layer 604. The pitch of the horizontal and vertical conductive threads can then be altered by weaving first textile layer 602 with second textile layer 604.

[0085] During operation, capacitive touch sensor 202 may be configured to determine positions of touch-input on the grid of conductive thread 208 using self-capacitance sensing or projective capacitive sensing.

[0086] When configured as a self-capacitance sensor, textile controller 204 charges crossing conductive threads 208 (e.g., horizontal and vertical conductive threads) by applying a control signal (e.g., a sine signal) to each conductive thread 208. When an object, such as the user’s finger, touches the grid of conductive thread 208, the conductive threads 208 that

are touched are grounded, which changes the capacitance (e.g., increases or decreases the capacitance) on the touched conductive threads 208.

[0087] Textile controller 204 uses the change in capacitance to identify the presence of the object. To do so, textile controller 204 detects a position of the touch-input by detecting which horizontal conductive thread 208 is touched, and which vertical conductive thread 208 is touched by detecting changes in capacitance of each respective conductive thread 208. Textile controller 204 uses the intersection of the crossing conductive threads 208 that are touched to determine the position of the touch-input on capacitive touch sensor 202. For example, textile controller 204 can determine touch data by determining the position of each touch as X,Y coordinates on the grid of conductive thread 208.

[0088] When implemented as a self-capacitance sensor, “ghosting” may occur when multi-touch input is received. Consider, for example, that a user touches the grid of conductive thread 208 with two fingers. When this occurs, textile controller 204 determines X and Y coordinates for each of the two touches. However, textile controller 204 may be unable to determine how to match each X coordinate to its corresponding Y coordinate. For example, if a first touch has the coordinates X1, Y1 and a second touch has the coordinates X4, Y4, textile controller 204 may also detect “ghost” coordinates X1, Y4 and X4, Y1.

[0089] In one or more implementations, textile controller 204 is configured to detect “areas” of touch-input corresponding to two or more touch-input points on the grid of conductive thread 208. Conductive threads 208 may be woven closely together such that when an object touches the grid of conductive thread 208, the capacitance will be changed for multiple horizontal conductive threads 208 and/or multiple vertical conductive threads 208. For example, a single touch with a single finger may generate the coordinates X1, Y1 and X2, Y1. Thus, textile controller 204 may be configured to detect touch-input if the capacitance is changed for multiple horizontal conductive threads 208 and/or multiple vertical conductive threads 208. Note that this removes the effect of ghosting because textile controller 204 will not detect touch-input if two single-point touches are detected which are spaced apart.

[0090] Alternately, when implemented as a projective capacitance sensor, textile controller 204 charges a single set of conductive threads 208 (e.g., horizontal conductive threads 208) by applying a control signal (e.g., a sine signal) to the single set of conductive threads 208. Then, textile controller 204 senses changes in capacitance in the other set of conductive threads 208 (e.g., vertical conductive threads 208).

[0091] In this implementation, vertical conductive threads 208 are not charged and thus act as a virtual ground. However, when horizontal conductive threads 208 are charged, the horizontal conductive threads capacitively couple to vertical conductive threads 208. Thus, when an object, such as the user’s finger, touches the grid of conductive thread 208, the capacitance changes on the vertical conductive threads (e.g., increases or decreases). Textile controller 204 uses the change in capacitance on vertical conductive threads 208 to identify the presence of the object. To do so, textile controller 204 detects a position of the touch-input by scanning vertical conductive threads 208 to detect changes in capacitance. Textile controller 204 determines the position of the touch-input as the intersection point between the vertical conductive thread 208 with the changed capacitance, and the horizontal

conductive thread **208** on which the control signal was transmitted. For example, textile controller **204** can determine touch data by determining the position of each touch as X,Y coordinates on the grid of conductive thread **208**.

[0092] Whether implemented as a self-capacitance sensor or a projective capacitance sensor, capacitive sensor **208** is configured to communicate the touch data to gesture manager **218** to enable gesture manager **218** to determine gestures based on the touch data, which can be used to control object **104**, computing device **106**, or applications **216** at computing device **106**.

[0093] Gesture manager **218** can be implemented to recognize a variety of different types of gestures, such as touches, taps, swipes, holds, and covers made to interactive textile **102**. To recognize the various different types of gestures, gesture manager **218** is configured to determine a duration of the touch, swipe, or hold (e.g., one second or two seconds), a number of the touches, swipes, or holds (e.g., a single tap, a double tap, or a triple tap), a number of fingers of the touch, swipe, or hold (e.g., a one finger-touch or swipe, a two-finger touch or swipe, or a three-finger touch or swipe), a frequency of the touch, and a dynamic direction of a touch or swipe (e.g., up, down, left, right). With regards to holds, gesture manager **218** can also determine an area of capacitive touch sensor **202** of interactive textile **102** that is being held (e.g., top, bottom, left, right, or top and bottom). Thus, gesture manager **218** can recognize a variety of different types of holds, such as a cover, a cover and hold, a five finger hold, a five finger cover and hold, a three finger pinch and hold, and so forth.

[0094] FIG. **10A** illustrates an example **1000** of generating a control based on touch-input corresponding to a single-finger touch. In example **1000**, horizontal conductive threads **208** and vertical conductive threads **208** of capacitive touch sensor **202** form an X,Y grid. The X-axis in this grid is labeled X1, X2, X3, and X4, and the Y-axis is labeled Y1, Y2, and Y3. As described above, textile controller **204** can determine the location of each touch on this X,Y grid using self-capacitance sensing or projective capacitance sensing.

[0095] In this example, touch-input **1002** is received when a user touches interactive textile **102**. When touch-input **1002** is received, textile controller **204** determines the position and time of touch-input **1002** on the grid of conductive thread **208**, and generates touch data **1004** which includes the position of the touch: "X1,Y1", and a time of the touch: T0. Then, touch data **1004** is communicated to gesture manager **218** at computing device **106** (e.g., over network **108** via network interface **210**).

[0096] Gesture manager **218** receives touch data **1004**, and generates a gesture **1006** corresponding to touch data **1004**. In this example, gesture manager **218** determines gesture **1006** to be "single-finger touch" because the touch data corresponds to a single touch-input point (X1,Y1) at a single time period (T0). Gesture manager **218** may then initiate a control **1008** to activate a functionality of computing device **106** based on the single-finger touch gesture **1006** to control object **104**, computing device **106**, or an application **216** at computing device **106**. A single-finger touch gesture, for example, may be used to control computing device **106** to power-on or power-off, to control an application **216** to open or close, to control lights in the user's house to turn on or off, and so on.

[0097] FIG. **10B** illustrates an example **1000** of generating a control based on touch-input corresponding to a double-tap. In this example, touch-input **1010** and **1012** is received when

a user double taps interactive textile **102**, such as by quickly tapping interactive textile **102**. When touch-input **1010** and **1012** is received, textile controller **204** determines the positions and time of the touch-input on the grid of conductive thread **208**, and generates touch data **1014** which includes the position of the first touch: "X1,Y1", and a time of the first touch: T0. The touch data **1014** further includes the position of the second touch: "X1,Y1", and the time of the second touch: T1. Then, touch data **1014** is communicated to gesture manager **218** at computing device **106** (e.g., over network **108** via network interface **210**).

[0098] Gesture manager **218** receives touch data **1014**, and generates a gesture **1016** corresponding to the touch data. In this example, gesture manager **218** determines gesture **1016** as a "double-tap" based on two touches being received at substantially the same position at different times. Gesture manager **218** may then initiate a control **1018** to activate a functionality of computing device **106** based on the double-tap touch gesture **1016** to control object **104**, computing device **106**, or an application **216** at computing device **106**. A double-tap gesture, for example, may be used to control computing device **106** to power-on an integrated camera, start the play of music via a music application **216**, lock the user's house, and so on.

[0099] FIG. **10C** illustrates an example **1000** of generating a control based on touch-input corresponding to a two-finger touch. In this example, touch-input **1020** and **1022** is received when a user touches interactive textile **102** with two fingers at substantially the same time. When touch-input **1020** and **1022** is received, textile controller **204** determines the positions and time of the touch-input on the grid of conductive thread **208**, and generates touch data **1024** which includes the position of the touch by a first finger: "X1,Y1", at a time T0. Touch data **1024** further includes the position of the touch by a second finger: "X3,Y2", at the same time T0. Then, touch data **1024** is communicated to gesture manager **218** at computing device **106** (e.g., over network **108** via network interface **210**).

[0100] Gesture manager **218** receives touch data **1024**, and generates a gesture **1026** corresponding to the touch data. In this case, gesture manager **218** determines gesture **1026** as a "two-finger touch" based on two touches being received in different positions at substantially the same time. Gesture manager may then initiate a control **1028** to activate a functionality of computing device **106** based on two-finger touch gesture **1026** to control object **104**, computing device **106**, or an application **216** at computing device **106**. A two-finger touch gesture, for example, may be used to control computing device **106** to take a photo using an integrated camera, pause the playback of music via a music application **216**, turn on the security system at the user's house and so on.

[0101] FIG. **10D** which illustrates an example **1000** of generating a control based on touch-input corresponding to a single-finger swipe up. In this example, touch-input **1030**, **1032**, and **1034** is received when a user swipes upwards on interactive textile **102** using a single finger. When touch-input **1030**, **1032**, and **1034** is received, textile controller **204** determines the positions and time of the touch-input on the grid of conductive thread **208**, and generates touch data **1036** corresponding to the position of a first touch as "X1,Y1" at a time T0, a position of a second touch as "X1,Y2" at a time T1, and a position of a third touch as "X1,Y3" at a time T2. Then, touch data **1036** is communicated to gesture manager **218** at computing device **106** (e.g., over network **108** via network interface **210**).

[0102] Gesture manager 218 receives touch data 1036, and generates a gesture 1038 corresponding to the touch data. In this case, the gesture manager 218 determines gesture 1038 as a “swipe up” based on three touches being received in positions moving upwards on the grid of conductive thread 208. Gesture manager may then initiate a control 1040 to activate a functionality of computing device 106 based on the swipe up gesture 1038 to control object 104, computing device 106, or an application 216 at computing device 106. A swipe up gesture, for example, may be used to control computing device 106 to accept a phone call, increase the volume of music being played by a music application 216, or turn on lights in the user’s house.

[0103] While examples above describe, generally, various types of touch-input gestures that are recognizable by interactive textile 102, it is to be noted that virtually any type of touch-input gestures may be detected by interactive textile 102. For example, any type of single or multi-touch taps, touches, holds, swipes, and so forth, that can be detected by conventional touch-enabled smart phones and tablet devices, may also be detected by interactive textile 102.

[0104] In one or more implementations, gesture manager 218 enables the user to create gestures and assign the gestures to functionality of computing device 106. The created gestures may include taps, touches, swipes and holds as described above. In addition, gesture manager 218 can recognize gesture strokes, such as gesture strokes corresponding to symbols, letters, numbers, and so forth.

[0105] Consider, for example, FIG. 11 which illustrates an example 1100 of creating and assigning gestures to functionality of computing device 106 in accordance with one or more implementations.

[0106] In this example, at a first stage 1102, gesture manager 218 causes display of a record gesture user interface 1104 on a display of computing device 106 during a gesture mapping mode. The gesture mapping mode may be initiated by gesture manager 218 automatically when interactive textile 102 is paired with computing device 106, or responsive to a control or command initiated by the user to create and assign gestures to functionalities of computing device 106.

[0107] In the gesture mapping mode, gesture manager 218 prompts the user to input a gesture to interactive textile 102. Textile controller 204, at interactive textile 102, monitors for gesture input to interactive textile 102 woven into an item of clothing (e.g., a jacket) worn by the user, and generates touch data based on the gesture. The touch data is then communicated to gesture manager 218.

[0108] In response to receiving the touch data from interactive textile 102, gesture manager 218 analyzes the touch data to identify the gesture. Gesture manager 218 may then cause display of a visual representation 1106 of the gesture on display 220 of computing device 106. In this example, visual representation 1106 of the gesture is a “v” which corresponds to the gesture that is input to interactive textile 102. Gesture user interface includes a next control 1108 which enables the user to transition to a second stage 1110.

[0109] At second stage 1110, gesture manager 218 enables the user to assign the gesture created at first stage 1102 to a functionality of computing device 106. As described herein, a “functionality” of computing device 106 can include any command, control, or action at computing device 102. Examples of functionalities of computing device 106 may include, by way of example and not limitation, answering a

call, music playing controls (e.g., next song, previous song, pause, and play), requesting the current weather, and so forth.

[0110] In this example, gesture manager 218 causes display of an assign function user interface 1112 which enables the user to assign the gesture created at first stage 1102 to one or more functionalities of computing device 102. Assign function user interface 1112 includes a list 1114 of functionalities that are selectable by the user to assign or map the gesture to the selected functionality. In this example, list 1114 of functionalities includes “refuse call”, “accept call”, “play music”, “call home”, and “silence call”.

[0111] Gesture manager receives user input to assign function user interface 1112 to assign the gesture to a functionality, and assigns the gesture to the selected functionality. In this example, the user selects the “accept call” functionality, and gesture manager 218 assigns the “v” gesture created at first stage 1102 to the accept call functionality.

[0112] Assigning the created gesture to the functionality of computing device 106 enables the user to initiate the functionality, at a subsequent time, by inputting the gesture into interactive textile 102. In this example, the user can now make the “v” gesture on interactive textile 102 in order to cause computing device 106 to accept a call to computing device 106.

[0113] Gesture manager 218 is configured to maintain mappings between created gestures and functionalities of computing device 106 in a gesture library. The mappings can be created by the user, as described above. Alternately or additionally, the gesture library can include predefined mappings between gestures and functionalities of computing device 106.

[0114] As an example, consider FIG. 12 which illustrates an example 1200 of a gesture library in accordance with one or more implementations. In example 1200, the gesture library includes multiple different mappings between gestures and device functionalities of computing device 106. At 1202, a “circle” gesture is mapped to a “tell me the weather” function, at 1204 a “v” gesture is mapped to an accept call function, at 1206 an “x” gesture is mapped to a “refuse call” function, at 1208 a “triangle” gesture is mapped to a “call home” function, at 1210 an “m” gesture is mapped to a “play music” function, and at 1212 a “w” gesture is mapped to a “silence call” function.

[0115] As noted above, the mappings at 1202, 1204, 1206, 1208, 1210, and 1212 may be created by the user or may be predefined such that the user does not need to first create and assign the gesture. Further, the user may be able to change or modify the mappings by selecting the mapping and creating a new gesture to replace the currently assigned gesture.

[0116] Notably, there may be a variety of different functionalities that the user may wish to initiate via a gesture to interactive textile 102. However, there is a limited number of different gestures that a user can realistically be expected to remember. Thus, in one or more implementations gesture manager 218 is configured to select a functionality based on both a gesture to interactive textile 102 and a context of computing device 106. The ability to recognize gestures based on context enables the user to invoke a variety of different functionalities using a subset of gestures. For example, for a first context, a first gesture may initiate a first functionality, whereas for a second context, the same first gesture may initiate a second functionality.

[0117] In some cases, the context of computing device 106 may be based on an application that is currently running on

computing device **106**. For example, the context may correspond to listening to music when the user is utilizing a music player application to listen to music, and to “receiving a call” when a call is communicated to computing device **106**. In these cases, gesture manager **218** can determine the context by determining the application that is currently running on computing device **106**.

[0118] Alternately or additionally, the context may correspond to an activity that the user is currently engaged in, such as running, working out, driving a car, and so forth. In these cases, gesture manager **218** can determine the context based on sensor data received from sensors implemented at computing device **106**, interactive textile **102**, or another device that is communicably coupled to computing device **106**. For example, acceleration data from an accelerometer may indicate that the user is currently running, driving in a car, riding a bike, and so forth. Other non-limiting examples of determining context include determining the context based on calendar data (e.g., determining the user is in a meeting based on the user’s calendar), determining context based on location data, and so forth.

[0119] After the context is determined, textile controller **204**, at interactive textile **102**, monitors for gesture input to interactive textile **102** woven into an item of clothing (e.g., a jacket) worn by the user, and generates touch data based on the gesture input. The touch data is then communicated to gesture manager **218**.

[0120] In response to receiving the touch data from interactive textile **102**, gesture manager **218** analyzes the touch data to identify the gesture. Then, gesture manager **218** initiates a functionality of computing device based on the gesture and the context. For example, gesture manager **218** can compare the gesture to a mapping that assigns gestures to different contexts. A given gesture, for example, may be associated with multiple different contexts and associated functionalities. Thus, when a first gesture is received, gesture manager **218** may initiate a first functionality if a first context is detected, or initiate a second, different functionality if a second, different context is detected.

[0121] As an example, consider FIG. **13** which illustrates an example **1300** of contextual-based gestures to an interactive textile in accordance with one or more implementations.

[0122] In this example, computing device **106** is implemented as a smart phone **1302** that is communicably coupled to interactive textile **102**. For example, interactive textile **102** may be woven into a jacket worn by the user, and coupled to smart phone **1302** via a wireless connection such as Bluetooth.

[0123] At **1304**, smart phone **1302** is in a “music playing” context because a music player application is playing music on smart phone **1302**. In the music playing context, gesture manager **218** has assigned a first subset of functionalities to a first subset of gestures at **1306**. For example, the user can play a previous song by swiping left on interactive textile **102**, play or pause a current song by tapping interactive textile **102**, or play a next song by swiping right on interactive textile **102**.

[0124] At **1308**, the context of smart phone **1302** changes to an “incoming call” context when smart phone **1302** receives an incoming call. In the incoming call context, the same subset of gestures is assigned to a second subset of functionalities which are associated with the incoming call context at **1310**. For example, by swiping left on interactive textile **102** the user can now reject the call, whereas before swiping left would have caused the previous song to be played in the

music playing context. Similarly, by tapping interactive textile **102** the user can accept the call, and by swiping right on interactive textile **102** the user can silence the call.

[0125] In one or more implementations, interactive textile **102** further includes one or more output devices, such as one or more light sources (e.g., LED’s), displays, speakers, and so forth. These output devices can be configured to provide feedback to the user based on touch-input to interactive textile **102** and/or notifications based on control signals received from computing device **106**.

[0126] FIG. **14** which illustrates an example **1400** of a jacket that includes an interactive textile **102** and an output device in accordance with one or more implementations. In this example, interactive textile **102** is integrated into the sleeve of a jacket **1402**, and is coupled to a light source **1404**, such as an LED, that is integrated into the cuff of jacket **1402**.

[0127] Light source **1404** is configured to output light, and can be controlled by textile controller **204**. For example, textile controller **204** can control a color and/or a frequency of the light output by light source **1404** in order to provide feedback to the user or to indicate a variety of different notifications. For example, textile controller **204** can cause the light source to flash at a certain frequency to indicate a particular notification associated with computing device **106**, e.g., a phone call is being received, a text message or email message has been received, a timer has expired, and so forth. Additionally, textile controller **204** can cause the light source to flash with a particular color of light to provide feedback to the user that a particular gesture or input to interactive textile **102** has been recognized and/or that an associated functionality is activated based on the gesture.

[0128] FIG. **15** illustrates implementation examples **1500** of interacting with an interactive textile and an output device in accordance with one or more implementations.

[0129] At **1502**, textile controller **204** causes a light source to flash at a specific frequency to indicate a notification that is received from computing device **106**, such as an incoming call or a text message.

[0130] At **1504**, the user places his hand over interactive textile **102** to cover the interactive textile. This “cover” gesture may be mapped to a variety of different functionalities. For example, this gesture may be used to silence a call or to accept a call. In response, the light source can be controlled to provide feedback that the gesture is recognized, such as by turning off when the call is silenced.

[0131] At **1506**, the user taps the touch sensor with a single finger to initiate a different functionality. For example, the user may be able to place one finger on the touch sensor to listen to a voicemail on computing device **106**. In this case, the light source can be controlled to provide feedback that the gesture is recognized, such as by outputting orange light when the voicemail begins to play.

[0132] Having discussed interactive textiles **102**, and how interactive textiles **102** detect touch-input, consider now a discussion of how interactive textiles **102** may be easily integrated within flexible objects **104**, such as clothing, handbags, fabric casings, hats, and so forth.

[0133] FIG. **16** illustrates various examples **1600** of interactive textiles integrated within flexible objects. Examples **1600** depict interactive textile **102** integrated in a hat **1602**, a shirt **1604**, and a handbag **1606**.

[0134] Interactive textile **102** is integrated within the bill of hat **1602** to enable the user to control various computing devices **106** by touching the bill of the user’s hat. For

example, the user may be able to tap the bill of hat **1602** with a single finger at the position of interactive textile **102**, to answer an incoming call to the user's smart phone, and to touch and hold the bill of hat **1602** with two fingers to end the call.

[0135] Interactive textile **102** is integrated within the sleeve of shirt **1604** to enable the user to control various computing devices **106** by touching the sleeve of the user's shirt. For example, the user may be able to swipe to the left or to the right on the sleeve of shirt **1604** at the position of interactive textile **102** to play a previous or next song, respectively, on a stereo system of the user's house.

[0136] In examples **1602** and **1604**, the grid of conductive thread **208** is depicted as being visible on the bill of the hat **1602** and on the sleeve of shirt **1604**. It is to be noted, however, that interactive textile **102** may be manufactured to be the same texture and color as object **104** so that interactive textile **102** is not noticeable on the object.

[0137] In some implementations, a patch of interactive textile **102** may be integrated within flexible objects **104** by sewing or gluing the patch of interactive textile **102** to flexible object **104**. For example, a patch of interactive textile **102** may be attached to the bill of hat **1602**, or to the sleeve of shirt **1604** by sewing or gluing the patch of interactive textile **102**, which includes the grid of conductive thread **208**, directly onto the bill of hat **1602** or the sleeve of shirt **1604**, respectively. Interactive textile **102** may then be coupled to textile controller **204** and power source **206**, as described above, to enable interactive textile **102** to sense touch-input.

[0138] In other implementations, conductive thread **208** of interactive textile **102** may be woven into flexible object **104** during the manufacturing of flexible object **104**. For example, conductive thread **208** of interactive textile **102** may be woven with non-conductive threads on the bill of hat **1602** or the sleeve of a shirt **1604** during the manufacturing of hat **1602** or shirt **1604**, respectively.

[0139] In one or more implementations, interactive textile **102** may be integrated with an image on flexible object **104**. Different areas of the image may then be mapped to different areas of capacitive touch sensor **202** to enable a user to initiate different controls for computing device **106**, or application **216** at computing device **106**, by touching the different areas of the image. In FIG. **16**, for example, interactive textile **102** is weaved with an image of a flower **1608** onto handbag **1606** using a weaving process such as jacquard weaving. The image of flower **1608** may provide visual guidance to the user such that the user knows where to touch the handbag in order to initiate various controls. For example, one petal of flower **1608** could be used to turn on and off the user's smart phone, another petal of flower **1608** could be used to cause the user's smart phone to ring to enable the user to find the smart phone when it is lost, and another petal of flower **1608** could be mapped to the user's car to enable the user to lock and unlock the car.

[0140] Similarly, in one or more implementations interactive textile **102** may be integrated with a three-dimensional object on flexible object **104**. Different areas of the three-dimensional object may be mapped to different areas of capacitive touch sensor **202** to enable a user to initiate different controls for computing device **106**, or application **216** at computing device **106**, by touching the different areas of the three-dimensional object. For example, bumps or ridges can be created using a material such as velvet or corduroy and woven with interactive textile **102** onto object **104**. In this

way, the three-dimensional objects may provide visual and tactile guidance to the user to enable the user to initiate specific controls. A patch of interactive textile **102** may be weaved to form a variety of different 3D geometric shapes other than a square, such as a circle, a triangle, and so forth.

[0141] In various implementations, interactive textile **102** may be integrated within a hard object **104** using injection molding. Injection molding is a common process used to manufacture parts, and is ideal for producing high volumes of the same object. For example, injection molding may be used to create many things such as wire spools, packaging, bottle caps, automotive dashboards, pocket combs, some musical instruments (and parts of them), one-piece chairs and small tables, storage containers, mechanical parts (including gears), and most other plastic products available today.

Example Methods

[0142] FIGS. **17**, **18**, **19**, and **20** illustrate an example method **1700** (FIG. **17**) of generating touch data using an interactive textile, an example method **1800** (FIG. **18**) of determining gestures usable to initiate functionality of a computing device, an example method **1900** (FIG. **19**) of assigning a gesture to a functionality of a computing device, and an example method **2000** (FIG. **20**) of initiating a functionality of a computing device based on a gesture and a context. These methods and other methods herein are shown as sets of blocks that specify operations performed but are not necessarily limited to the order or combinations shown for performing the operations by the respective blocks. In portions of the following discussion reference may be made to environment **100** of FIG. **1** and system **200** of FIG. **2**, reference to which is made for example only. The techniques are not limited to performance by one entity or multiple entities operating on one device.

[0143] FIG. **17** illustrates an example method **1700** of generating touch data using an interactive textile.

[0144] At **1702**, touch-input to a grid of conductive thread woven into an interactive textile is detected. For example, textile controller **204** (FIG. **2**) detects touch-input to the grid of conductive thread **208** woven into interactive textile **102** (FIG. **1**) when an object, such as a user's finger, touches interactive textile **102**.

[0145] Interactive textile **102** may be integrated within a flexible object, such as shirt **104-1**, hat **104-2**, or handbag **104-3**. Alternately, interactive textile **102** may be integrated with a hard object, such as plastic cup **104-4** or smart phone casing **104-5**.

[0146] At **1704**, touch data is generated based on the touch-input. For example, textile controller **204** generates touch data based on the touch-input. The touch data may include a position of the touch-input on the grid of conductive thread **208**.

[0147] As described throughout, the grid of conductive thread **208** may include horizontal conductive threads **208** and vertical conductive threads **208** positioned substantially orthogonal to the horizontal conductive threads. To detect the position of the touch-input, textile controller **204** can use self-capacitance sensing or projective capacitance sensing.

[0148] At **1706**, the touch data is communicated to a computing device to control the computing device or one or more applications at the computing device. For example, network interface **210** at object **104** communicates the touch data generated by textile controller **204** to gesture manager **218** implemented at computing device **106**. Gesture manager **218**

and computing device **106** may be implemented at object **104**, in which case interface may communicate the touch data to gesture manager **218** via a wired connection. Alternately, gesture manager **218** and computing device **106** may be implemented remote from interactive textile **102**, in which case network interface **210** may communicate the touch data to gesture manager **218** via network **108**.

[0149] FIG. **18** illustrates an example method **1800** of determining gestures usable to initiate functionality of a computing device in accordance with one or more implementations.

[0150] At **1802**, touch data is received from an interactive textile. For example, network interface **222** (FIG. **2**) at computing device **106** receives touch data from network interface **210** at interactive textile **102** that is communicated to gesture manager **218** at step **906** of FIG. **9**.

[0151] At **1804**, a gesture is determined based on the touch data. For example, gesture manager **218** determines a gesture based on the touch data, such as single-finger touch gesture **506**, a double-tap gesture **516**, a two-finger touch gesture **526**, a swipe gesture **538**, and so forth.

[0152] At **1806**, a functionality is initiated based on the gesture. For example, gesture manager **218** generates a control based on the gesture to control an object **104**, computing device **106**, or an application **216** at computing device **106**. For example, a swipe up gesture may be used to increase the volume on a television, turn on lights in the user's house, open the automatic garage door of the user's house, and so on.

[0153] FIG. **19** illustrates an example method **1900** of assigning a gesture to a functionality of a computing device in accordance with one or more implementations.

[0154] At **1902**, touch data is received at a computing device from an interactive textile woven into an item of clothing worn by the user. For example, network interface **222** (FIG. **2**) at computing device **106** receives touch data from network interface **210** at interactive textile **102** that is woven into an item of clothing worn by a user, such as a jacket, shirt, hat, and so forth.

[0155] At **1904**, the touch data is analyzed to identify a gesture. For example, gesture manager **218** analyzes the touch data to identify a gesture, such as a touch, tap, swipe, hold, or gesture stroke.

[0156] At **1906**, user input to assign the gesture to a functionality of the computing device is received. For example, gesture manager **218** receives user input to assign function user interface **1112** to assign the gesture created at step **1904** to a functionality of computing device **106**.

[0157] At **1908**, the gesture is assigned to the functionality of the computing device. For example, gesture manager **218** assigns the functionality selected at step **1906** to the gesture created at step **1904**.

[0158] FIG. **20** illustrates an example method **2000** of initiating a functionality of a computing device based on a gesture and a context in accordance with one or more implementations.

[0159] At **2002**, a context associated with a computing device or a user of the computing device is determined. For example, gesture manager **218** determines a context associated with computing device **106** or a user of computing device **106**.

[0160] At **2004**, touch data is received at the computing device from an interactive textile woven into a clothing item worn by the user. For example, touch data is received at

computing device **106** from interactive textile **102** woven into a clothing item worn by the user, such as jacket, shirt, or hat.

[0161] At **2006**, the touch data is analyzed to identify a gesture. For example, gesture manager **218** analyzes the touch data to identify a gesture, such as a touch, tap, swipe, hold, stroke, and so forth.

[0162] At **2008**, a functionality is activated based on the gesture and the context. For example, gesture manager **218** activates a functionality based on the gesture identified at step **2006** and the context determined at step **2002**.

[0163] The preceding discussion describes methods relating to gestures for interactive textiles. Aspects of these methods may be implemented in hardware (e.g., fixed logic circuitry), firmware, software, manual processing, or any combination thereof. These techniques may be embodied on one or more of the entities shown in FIGS. **1-16** and **21** (computing system **2100** is described in FIG. **21** below), which may be further divided, combined, and so on. Thus, these figures illustrate some of the many possible systems or apparatuses capable of employing the described techniques. The entities of these figures generally represent software, firmware, hardware, whole devices or networks, or a combination thereof.

Example Computing System

[0164] FIG. **21** illustrates various components of an example computing system **2100** that can be implemented as any type of client, server, and/or computing device as described with reference to the previous FIGS. **1-20** to implement two-layer interactive textiles. In embodiments, computing system **2100** can be implemented as one or a combination of a wired and/or wireless wearable device, System-on-Chip (SoC), and/or as another type of device or portion thereof. Computing system **2100** may also be associated with a user (e.g., a person) and/or an entity that operates the device such that a device describes logical devices that include users, software, firmware, and/or a combination of devices.

[0165] Computing system **2100** includes communication devices **2102** that enable wired and/or wireless communication of device data **2104** (e.g., received data, data that is being received, data scheduled for broadcast, data packets of the data, etc.). Device data **2104** or other device content can include configuration settings of the device, media content stored on the device, and/or information associated with a user of the device. Media content stored on computing system **2100** can include any type of audio, video, and/or image data. Computing system **2100** includes one or more data inputs **2106** via which any type of data, media content, and/or inputs can be received, such as human utterances, touch data generated by interactive textile **102**, user-selectable inputs (explicit or implicit), messages, music, television media content, recorded video content, and any other type of audio, video, and/or image data received from any content and/or data source.

[0166] Computing system **2100** also includes communication interfaces **2108**, which can be implemented as any one or more of a serial and/or parallel interface, a wireless interface, any type of network interface, a modem, and as any other type of communication interface. Communication interfaces **2108** provide a connection and/or communication links between computing system **2100** and a communication network by which other electronic, computing, and communication devices communicate data with computing system **2100**.

[0167] Computing system **2100** includes one or more processors **2110** (e.g., any of microprocessors, controllers, and the like), which process various computer-executable instructions to control the operation of computing system **2100** and to enable techniques for, or in which can be embodied, interactive textiles. Alternatively or in addition, computing system **2100** can be implemented with any one or combination of hardware, firmware, or fixed logic circuitry that is implemented in connection with processing and control circuits which are generally identified at **2112**. Although not shown, computing system **2100** can include a system bus or data transfer system that couples the various components within the device. A system bus can include any one or combination of different bus structures, such as a memory bus or memory controller, a peripheral bus, a universal serial bus, and/or a processor or local bus that utilizes any of a variety of bus architectures.

[0168] Computing system **2100** also includes computer-readable media **2114**, such as one or more memory devices that enable persistent and/or non-transitory data storage (i.e., in contrast to mere signal transmission), examples of which include random access memory (RAM), non-volatile memory (e.g., any one or more of a read-only memory (ROM), flash memory, EPROM, EEPROM, etc.), and a disk storage device. A disk storage device may be implemented as any type of magnetic or optical storage device, such as a hard disk drive, a recordable and/or rewritable compact disc (CD), any type of a digital versatile disc (DVD), and the like. Computing system **2100** can also include a mass storage media device **2116**.

[0169] Computer-readable media **2114** provides data storage mechanisms to store device data **2104**, as well as various device applications **2118** and any other types of information and/or data related to operational aspects of computing system **2100**. For example, an operating system **2120** can be maintained as a computer application with computer-readable media **2114** and executed on processors **2110**. Device applications **2118** may include a device manager, such as any form of a control application, software application, signal-processing and control module, code that is native to a particular device, a hardware abstraction layer for a particular device, and so on.

[0170] Device applications **2118** also include any system components, engines, or managers to implement interactive textiles. In this example, device applications **2118** include gesture manager **218**.

CONCLUSION

[0171] Although embodiments of techniques using, and objects including, two-layer interactive textiles have been described in language specific to features and/or methods, it is to be understood that the subject of the appended claims is not necessarily limited to the specific features or methods described. Rather, the specific features and methods are disclosed as example implementations of two-layer interactive textiles.

What is claimed is:

1. An interactive textile configured to be integrated within a flexible object, the interactive textile comprising:
 - a top textile layer comprising first conductive threads woven into the top textile layer;
 - a bottom textile layer comprising second conductive threads woven into the bottom textile layer, the first

conductive threads and the second conductive threads configured to form a capacitive touch sensor; and

- a textile controller coupled to the capacitive touch sensor, the textile controller configured to detect touch-input to the capacitive touch sensor when an object touches the capacitive touch sensor, and process the touch-input to provide touch data usable to control a computing device wirelessly coupled to the interactive textile.

2. The interactive textile as recited in claim 1, wherein the textile controller is coupled to the capacitive touch sensor via the second conductive threads of the bottom textile layer.

3. The interactive textile as recited in claim 1, wherein the first conductive threads woven into the top textile layer and the second conductive threads woven into the bottom textile layer form a grid of conductive threads.

4. The interactive textile as recited in claim 3, wherein the first conductive threads woven into the top textile layer are positioned substantially orthogonal to the second conductive threads woven into the second textile layer to form the grid of conductive threads.

5. The interactive textile as recited in claim 1, wherein a pitch of the second conductive threads woven into the second textile layer is constant.

6. The interactive textile as recited in claim 1, wherein a pitch of the second conductive threads woven into the second textile layer is not constant.

7. The interactive textile as recited in claim 6, wherein shrinking polymers are used to vary the pitch of the second conductive threads woven into the second textile layer.

8. The interactive textile as recited in claim 6, wherein the second conductive threads are partially woven into the second textile layer.

9. The interactive textile as recited in claim 1, wherein the first textile layer and the second textile layer are combined using a 3D-weaving process.

10. The interactive textile as recited in claim 1, wherein the flexible object comprises a clothing item.

11. The interactive textile as recited in claim 10, wherein the first textile layer comprises an outside surface of the clothing item.

12. The interactive textile as recited in claim 1, wherein the second textile layer comprises a standardized structure of second conductive threads.

13. The interactive textile as recited in claim 1, wherein the first and second conductive threads each comprise a conductive wire that is twisted, wrapped, or braided with one or more flexible threads.

14. The interactive textile as recited in claim 13, wherein the conductive wire comprises a copper wire, a gold wire, or a silver wire.

15. A flexible object comprising:

- an interactive textile integrated within the flexible object, the interactive textile comprising a top textile layer and a bottom textile layer, the top textile layer comprising first conductive threads woven into the top textile layer and the bottom textile layer comprising second conductive threads woven into the bottom textile layer; and

a textile controller coupled to the interactive textile, the textile controller configured to detect touch-input to the interactive textile, and process the touch-input to provide touch data usable to initiate functionality of a computing device wireless coupled to the interactive textile.

16. The flexible object as recited in claim **15**, wherein the textile controller is coupled to the interactive textile via the second conductive threads of the bottom textile layer.

17. The flexible object as recited in claim **15**, wherein the flexible object comprises a clothing item.

18. The flexible object as recited in claim **15**, wherein the second conductive threads and the textile controller are not visible when the clothing item is worn by a user.

19. The flexible object as recited in claim **15**, wherein a pitch of the second conductive threads woven into the second textile layer is constant.

20. The flexible object as recited in claim **15**, wherein the first textile layer and the second textile layer are combined using a 3D-weaving process.

* * * * *