Office de la Propriété Intellectuelle du Canada
Un organisme d'Industrie Canada

Canadian Intellectual Property Office
An agency of Industry Canada

CA 2735325 A1 2011/09/25

(21) **2 735 325**

(12) **DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION**

(13) **A1**

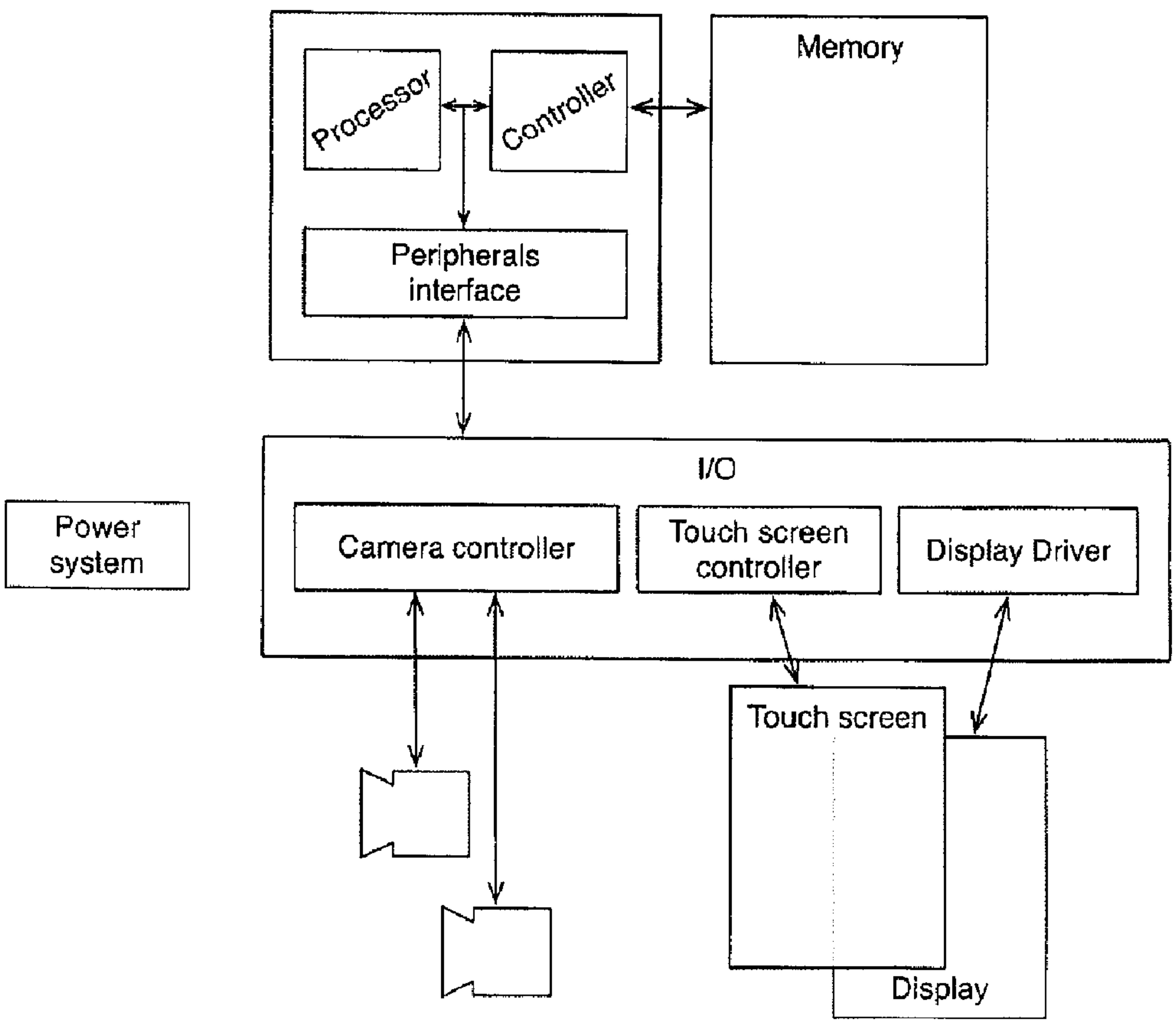
(22) **Date de dépôt/Filing Date:** 2011/03/25
(41) **Mise à la disp. pub./Open to Public Insp.:** 2011/09/25
(30) **Priorité/Priority:** 2010/03/25 (US61/317,288)

(51) **Cl.Int./Int.Cl.** *G06F 3/01* (2006.01), *G06F 3/041* (2006.01)
(71) **Demandeur/Applicant:**
USER INTERFACES IN SWEDEN AB, SE

(72) **Inventeurs/Inventors:**
GARDENFORS, DAN, SE;
JOHANSSON, KARL-ANDERS, SE;
HALIBURTON, JAMES, US

(74) **Agent:** RIDOUT & MAYBEE LLP

(54) Titre : SYSTEME ET PROCEDE DE DETECTION DU MOUVEMENT ET RETROACTION
(54) Title: SYSTEM AND METHOD FOR GESTURE DETECTION AND FEEDBACK



(57) **Abrégé/Abstract:**
A computer device with a sensor subsystem for detecting off-surface objects, that carries out continued processing of the position and shape of objects detected in the vicinity of the device, associates these positions and shapes with predetermined gesture states, determines if the object is transitioning between gesture states and provides feedback based on the determined transition between the gesture states.

ABSTRACT

SYSTEM AND METHOD FOR GESTURE DETECTION AND FEEDBACK

A computer device with a sensor subsystem for detecting off-surface objects, that carries out continued processing of the position and shape of objects detected in the vicinity of the device, associates these positions and shapes with predetermined gesture states, determines if the object is transitioning between gesture states and provides feedback based on the determined transition between the gesture states.

SYSTEM AND METHOD FOR GESTURE DETECTION AND FEEDBACK

FIELD OF INVENTION

This invention relates generally to computer systems and more particularly to 2D and 3D gesture input and recognition, and graphical user interface for computer systems.

BACKGROUND OF THE INVENTION

In the field of gesture interaction with computer system, two primary concerns are prevalent:

1) The user of a gesture powered user interface is often at a loss as to how to interact with it. This is due to many reasons; one being poor feedback of when the system interprets some action on the user's part as a gesture. That, or the system constructors simply tries to reinvent old metaphors such as “pointing and clicking”, not taking into account the inherent physical differences in moving a computer mouse and pointing in mid-air, most often failing due to lack of precision. Only trying to emulate a mouse is a misguided approach, as this closes the door for many novel interaction techniques

2) Off-surface object detection is typically done using one or more cameras; using triangulation if more than one is used. In some cases camera sensors measuring the time-of-flight for infrared light bounced on the objects is used. This gives the actual distance to the object being detected. Cameras of any kind however consume quite a lot of power, and in addition to the cameras, substantial processing power on a central processing unit or a digital signal processor must also be used to interpret the camera images to detect any objects. Since off-surface object detection relies on sensors being switched on constantly, power consumption for such detection is constantly high, making it unsuitable for portable devices.

SUMMARY OF THE INVENTION

The present invention brings together and improves on the prior art in primarily two ways:

1) Feedback, preferably continuous, indicating to the end user what is about to happen if a certain gesture is completed – thus adding an exploratory level to the user interface.

Thus, from one aspect, the present invention provides: a computing device, comprising: a display; a sensor subsystem for detecting off-surface objects; memory having instructions stored on it which when run on a processor, causes the processor to perform the steps of: detecting an off-surface object using the sensor subsystem; determining a position and a shape of said detected off-surface object; matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state; detecting changes in the determined position and shape using the sensor subsystem; matching said changes in the determined position with transitions between said reference zones and matching said changes in determined shapes with transitions between said reference poses to thereby determine a transition between gesture states; and, instructing a user interface to provide feedback based on the determined transition between gesture states.

Certain preferred features of the embodiments of the above aspect of the present invention are set out in the appended claims.

From another aspect, the present invention provides: a computer-implemented method, for operating a computing device comprising a display, sensor sub system, processor and memory, comprising the steps of: detecting an off-surface object using the sensor subsystem; determining a position and a shape of said detected off-surface object; matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state; detecting changes in the determined position and shape using the sensor subsystem; matching said changes in the determined position with transitions between said reference zones and matching said changes in determined shapes with transitions between said reference poses to thereby determine a transition between gesture states; and, instructing a user interface to provide feedback based on the determined transition between gesture states.

From yet another aspect, the present invention provides: A computer program product having instructions which when run on a processor causes the processor to carry out the above method.

These three aspects achieve provision of feedback to the end user by tightly coupling changes in determined positions and shapes of the detected objects to immediate and continuous feedback in the user interface. A state machine keeps track of in what discrete state the object is with respect to position and shape, but by also considering the state transition graph, it is possible to deduce where in between two poses the current shape is, and also in between which activation zones the current position is. That information is used to hint the user as to what would happen if the current gesture is continued or made stronger. This invites to "playing" with the user interface for exploring new features as the user can interact with the system using small/subtle gestures and not risk triggering any functionality, but rather just get a feel for what gestures are linked to what functionality.

The second way the present invention improves on the prior art is:

2) Improved power consumption while at the same time minimizing detection of spurious gestures by giving the end user direct control of when the gesture detection subsystem is in an active state.

Thus, from a fourth aspect, the present invention provides: a computing device, comprising: a display; a sensor subsystem for detecting off-surface objects; a means for detecting user input; memory having instructions stored on it which when run on a processor, causes the processor to perform the steps of: detecting an off-surface object using the sensor subsystem; determining a position and a shape of said detected off-surface object; matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state; setting the sensor subsystem to a powered-on state when the means for detecting a user input has detected a predetermined user input.

Certain preferred features of the embodiments of the first aspect of the present invention set out in the appended claims can also be applied to the embodiments of the fourth aspect of the present invention.

From a further aspect, the present invention provides: a computer-implemented method for use in a computing device comprising a display, a sensor subsystem, a means for detecting user input, a processor and memory, comprising the steps of: detecting an off-surface object using the sensor subsystem; determining a position and a shape of said detected off-surface object; matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state; setting the sensor subsystem to a powered-on state when the means for detecting a user input has detected a predetermined user input.

From a final aspect, the present invention provides: a computer program product having instructions which when run on a processor causes the processor to carry out the above method.

The above three aspects achieve improved power consumption by tightly coupling the powering up of the off-surface gesture detection subsystem to an intentional gesture on the surface of the device. The on-surface gesture can be anything from the pressing of a hardware button or a simple touch on a touch screen, to a complex multi touch gesture. This not only conserves power, but also makes the user aware of the system being in a gesture sensitive mode and makes the user behave accordingly.

As will become clear from reading the detailed description of the preferred embodiments, which are provided by way of example only, and seeing the figures illustrating the use cases, the aspects above are powerful enough by themselves, but the combination opens up for entirely new use cases from an interaction point-of-view with extremely low power usage.

BRIEF DESCRIPTION OF THE DRAWINGS

FIG. 1 is a block diagram of an example computer system in accordance with the present invention.

FIG. 2 contains an example of a modification to the computer system block diagram related to the processor configuration.

FIG. 3 contains further examples of modifications to the computer system block diagram related to camera input, most specifically to the possibilities of using a) a single camera, and b) a depth-sensing camera.

FIG. 4 contains an example of a physical embodiment of the present invention, showing a portable device with a touch screen and stereoscopic cameras for gesture interaction.

FIG. 5 a)-d) contains four example poses that can be classified

FIG. 6 a) contains a generic state transition diagram for moving between M different poses and the undefined Null pose, b)-e) contains specific state transition diagrams for M=0, M=1, M=2 and M=3 respectively.

FIG. 7 shows an example object a) identified as the Null pose, b) identified as lying in between the Null pose and the P1 pose, and c) identified as the P1 pose.

FIG. 8 shows various examples of gesture activation zones: a) 3d volumes, b) 3d points, c) 2d areas, and d) 2d points.

FIG. 9 a) contains a generic state transition diagram for moving between N different gesture activation zones Z and the undefined Null zone, b)-e) contains specific state transition diagrams for N=0, N=1, N=2 and N=3 respectively.

FIG. 10 shows an object determined as a) lying in the Z2 activation zone, b) lying in between zones, but heading most strongly towards the Z4 zone, and c) lying in the Z4 activation zone.

FIG. 11 illustrates a use case where a gesture starts on the touch screen (a-c) and is continued in front of the device (d-f), finishing with touch screen interaction. (g)

FIG. 12 illustrates a first feedback use case: a) a media player in an idle state, b) a subtle motion gesture to the left indicate the functionality behind further motion to the left ("next track"), and c) a distinct motion gesture to the left activates "next track".

FIG. 13 illustrates a second feedback use case: a) a media player in an idle state, b) when a hand is in the vicinity, idle particles appear, c) a subtle motion gesture to the left makes the particles flow with the gesture, and d) a distinct motion gesture to the left makes the particles seem to grab onto the album art which is swept off the screen, triggering "next track"

FIG. 14 illustrates a third feedback use case: a) a media player in an idle state, b) when a hand is in the vicinity, idle particles appear, c) the hand forms a pointed pose and the particles gather in a tight spot where the hand points, and d) moving the hand moves the particle spot and the underlying album art follows along as it is "grabbed" by the particles.

FIG. 15 illustrates a simple combination gesture: a) idle state, b) the system is not yet activated for identifying gestures, so gestures does not affect the user interface, c) activating the gesture detection system by holding a finger on the touch screen, and d) the gesture in front of the device now has effect on the user interface.

FIG. 16 illustrates another combination gesture: a) idle state showing two 3D cubes in the user interface, b) activating the gesture detection system by holding a finger on one of the cubes, c) a gesture in front of the device now affects the orientation of the selected object, and d) alternate interaction paradigm where the object held by the finger is still and the other cube is spun instead.

FIG. 17 illustrates another combination gesture: a) idle state, b) the system is not yet activated for identifying gestures, so gestures does not affect the user interface, c) activating the gesture detection system by holding a finger on a dedicated hardware button, and d) the gesture in front of the device has effect on the user interface while the button is pressed. In an alternate embodiment e) the gesture in front of the device has effect on the user interface as long as the button was pressed before, within a certain time limit.

FIG. 18 illustrates a feedback use case for video telephony/conference: a) the user is in front of the device and sees his/her counterpart, and b) the user has moved to the side, so as feedback to the user to make him/her move to the left, the counterpart image is moved to the right.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENTS

We will now examine in detail a small number of preferred embodiments of the invention. The accompanying drawings are used to illustrate aspects of these embodiments in particular and aspects of the invention in general. While the invention will be described in conjunction with a set of preferred embodiments, it will be understood that it is not intended to limit the invention to one or a few preferred embodiment. To the contrary, the appended claims are intended to be interpreted as covering all alterations, modifications, and equivalents of the described embodiments within the true spirit and scope of the invention.

Referring to the figures now, figure 4 illustrates an example physical embodiment of the invention: a portable device equipped with a touch screen and two front-facing cameras in a stereoscopic configuration. Figure 1 shows an example block diagram, showing how a processor can access and process the camera data, storing determined object positions and shapes in memory and also drawing a graphical user interface on the display, partly based on the information about the determined object positions and shapes stored in memory. The power system is under the processor's control and controls the power to the various system blocks. Further, figure 2 illustrates an example alternative processor macro block, featuring a dedicated graphical processor that could aid in drawing the user interface, and a dedicated signal processor that could aid in processing the camera data. Figure 3 illustrates example alternative camera setups, a) using a single camera, b) using a true depth sensing time-of-flight camera module. In the end, the exact choice of object detection method is not the topic for this invention, but rather the continued processing of the determined position and shapes of the detected objects in the vicinity of the device, ending up with a more usable, more energy efficient, and more exploratory gesture user interface.

Focusing on one aspect, the present invention is particularly well suited for gesture recognition on a portable device where power consumption is a major concern. This is due to the novel combination gesture approach illustrated in the simplified use cases of figures 15, 16 and 17. In figure 15 a) and b) the subsystem for detecting objects, including the cameras, is powered down and the gesture in b) goes undetected. Not until the user, as in c), activates the system with a finger on the display, can the user as in d) interact with the user interface using gestures. The interaction is illustrated with a simple square transitioning into a star. It is implied that any user interface change can occur due to the gesture interaction. Note that in this use case, it is enough to touch anywhere on the screen to activate the gesture detection. Figure 16 illustrates how touching specific objects in the user interface can affect the user interface differently. In a) we see two 3D cubes. In b) the user holds a finger one of the cubes and therefore powers up the sensor subsystem for detecting objects so that in c) the selected cube is rotated when the user performs an appropriate gesture in front of the device. d) shows a different interaction paradigm where the cube beneath the finger is held still and the other cube is rotated instead. This use case of course extends to being able to press outside the cubes and rotate the "world" or "camera". Of course this also extends to other interactions than rotation as well, such as scrolling a list, selecting items, switching between discrete views etc. Figure 17 illustrates yet another example of how one interaction can trigger the off-screen gesture detection. In figure 17 a) and b), just like in figure 15 a) and b), the subsystem for detecting objects, including the cameras, is powered down and the gesture in b) goes undetected. Not until the user, as in 17 c), activates the system by pressing a dedicated hardware button can the user as in 17 d) interact with the user interface using gestures. In an alternate interaction paradigm, illustrated in 17 e) is the off-surface gesture detection subsystem activated after pressing the button once, as opposed to keeping it pressed during the gesture. The interaction in figure 17 is just as in figure 15 illustrated with a simple square transitioning into a star. Like above, it is implied that any user interface change can occur due to the gesture interaction.

Another aspect of the invention is the continuous feedback mechanism as illustrated by the different examples in figures 12-14. Figure 12 illustrates a media player use case. In 12 a) we see the media player in its idle state. There is album art displaying the currently playing track and there are navigation buttons at the bottom of the screen. No gesture is currently detected. In b) a hand has been detected and it is slowly moving to the left. The motion is not distinct enough to trigger an actual effect in the state of the media player, but subtle hints as to what this gesture means are given, including starting to move the album art to the left and slightly highlighting the "next track" button. If the gesture is discontinued, these hints fade out and the album art is restored to its original position. If as in c) the gesture is made stronger, i.e. more distinct, the feedback is stronger as well, with for instance a more distinctly marked “next track” button indicating that the gesture has taken effect.

This is achieved by analyzing the gesture state transition diagrams, illustrated in general in figures 6 and 9. Gestures are broken down to changes in position and shape, and those are the state diagrams illustrated in figures 6 and 9. Figure 6 shows the state transition diagrams for identifying the current pose of the object based on the current shape of the object. 6 a) is a generic pose state transition diagram for M poses, whereas 6 b)-e) are specific examples for M=0, M=1, M=2 and M=3 respectively. Example poses for a hand is given in Figure 5, where a) indicates an undetermined “Null” pose, b) indicates a pointing pose, c) indicates a rigid whole hand pose, and d) indicates a pinching pose. This list of poses is by no means complete but rather serves as example poses that could be detected by the system. The example in figure 7 refers to the state change from a) the undetermined “Null” pose to c) a pointing hand pose via b) the intermediate pose. The pose state transition diagram shows all possible ways to navigate between the poses, and each transition can be given its own special feedback mechanism in the user interface. Of equal importance to the pose state transition diagram is the activation zone state transition diagram for identifying activation zone changes based on position changes. Example activation zones implementations are illustrated in figure 8 with a) defining four example activation zones as interaction volumes in 3D, b) interaction points in 3D, c) interaction areas in 2D, and d) interaction points in 2D. The exact placement of the activation zones is application dependant. Figure 9 shows the state transition diagrams for identifying the current activation zone of the object based on the current position of the object. 9 a) is a generic activation zone state transition diagram for N poses, whereas 9 b)-e) are specific examples for N=0, N=1, N=2 and N=3 respectively. Figure 10 illustrates and example where a hand is moving in front of the system, a) starting in activation zone Z2, b) being in an intermediary state in between the other zones, identified as moving most strongly towards activation zone Z4, and c) having reached activation zone Z4. As with the pose state transition diagram, each activation zone state transition can be given its own specific user interface feedback mechanism.

Continuing on the continuous feedback examples, in figure 13 another implementation of the media player is illustrated. In a) a media player in its idle state is shown. In b) a hand enters and is recognized by the system. This is illustrated in the UI by a particle system with particles in an idle state appearing. The particles' random motion shows that they are idle. In c) a subtle motion to the right is made with the hand, which the particles follow, and the "previous track" icon is slightly highlighted, indicating that this is the functionality triggered by that kind of gesture. In d) a distinct motion to the left is made and the "next track" is triggered as the particles move sharply to the left as well. The particle system concept is explored further in figure 14 that starts off just like the previous example a) and b). In figure 14 c) however, the user makes a pointed gesture that shows the intent of the user and that makes the particles focus in a tight spot. In d) it is shown that the grouped-together particles grab onto the underlying album art and drags it along when the finger is moved.

The particle system examples are included to illustrate the plethora of available visualization techniques that can be used to indicate to the user that the system is aware of the subtle changes in the user's gestures. Alternative means of visualizations includes, but is not limited to, color scheme changes, size changes of user interface elements, and various renditions of fields, such as magnetic fields or wind direction.

Although references has been heavy on graphical / visual feedback, auditory and other sensory feedback, such as haptic feedback, is of great importance. Audio in particular gives invaluable feedback in use cases where no hand is in contact with the device. It is very much so an aspect of this invention that many of the same parameters that could control a particle system in user feedback visualization would also control an audio synthesizer for giving continuous and relevant auditory feedback.

Moving back to the use cases, referring to figure 11, a compound use case comprising both the user triggered gesture subsystem activation and the continuous feedback is illustrated next. In a) a photo album application is illustrated. The off-screen gesture detection is powered down to conserve power. In b) the user starts a pinch gesture on the touch screen surface, indicating that the user wants to grab hold of a photo. Still, the sensor subsystem for detecting objects is without power. In c) the pinch gesture has been completed, and heuristics in the system has concluded the same thing, determining that this is a gesture that could be continued in front of the device, so the object detection subsystem is powered up, giving the system the ability to track the hand. In d) the fingers have been lifted up off the surface, and the photo is faded out more and more the further away the hand is. In e) the user releases the pinch pose. The photo still hovers translucently in the user interface. In f) the user points distinctly, indicates that he/she wants to take action with the photo, so an action list is brought up on the screen. Finally, in g) the user presses one icon on the action list on the touch screen, bringing the compound gesture to an end.

This use case illustrates that there are several ways to trigger the activation of the off-surface gesture detection subsystem, i.e. the camera(s), digital signal processor(s) and programs running on one or more of the processors. In this use case the touch screen gesture was a pinch gesture, but it should be obvious that any similar gesture, such as a single- or multi-finger slide, a long-press, a rotary gesture, a two handed gesture or even a simple tap or double-tap could be followed by an off-screen gesture.

Moving away from the previous use cases of having as the only object a hand, we can also treat a user's face as an object that can be detected by the sensor subsystem. In figure 18 we illustrate this concept with a video telephony or a video conferencing application. Instead of occupying valuable screen estate with an image of the user to show him/her if he/she is in the picture, that feedback can be given indirectly by moving the image of the other party in the conversation. As illustrated in 18 a) the user is directly in front of the device, which is therefore showing the entire image of the other party. In b) the user has moved too far to the right and the image of the other party is subsequently moved to the right, giving the user all feedback he/she needs to correct his/her position. Similar activation zones as in figure 8 or figure 10 could very well be used for this use case.

In accordance with one example, there is provided a computing device, comprising: a display; a sensor subsystem for detecting objects; one or more processors coupled to the display, and to the sensor subsystem for detecting objects; memory; and one or more programs, wherein the one or more programs are stored in the memory and configured to be executed by the one or more processors; wherein the sensor subsystem for detecting objects and at least one of the programs running on one or more of the one or more processors include at least one means for determining positions and shapes of one or more objects in the vicinity of the sensor and store this in the memory; wherein at least one of the programs executed on one or more of the one or more processors includes instructions for detecting changes in the determined positions and shapes of one or more objects in the vicinity of the sensor; wherein at least one of the programs running on one or more of the one or more processors includes instructions for matching the detected changes in the determined positions and shapes of one or more objects in the vicinity of the sensor with a set of reference zones and reference poses to maintain a state of ongoing gestures; and wherein at least one of the programs running on one or more of the one or more processors includes instructions for managing a user interface coupled with the display.

In some examples, the user interface gives continuous feedback based on the state of ongoing gestures.

In some examples, the computing device comprises an input device coupled to at least one of the one or more processors.

In some examples, the subsystem for detecting objects can detect body parts of the human body.

In some examples, the sensor subsystem for detecting objects includes a camera.

In some examples, the reference zones are represented as volumes in a three-dimensional space.

In some examples, the state of ongoing gestures includes at least one of: where in between the reference zones the one or more objects' determined positions are present, which reference zones are the closest ones to the one or more determined positions, where in between the reference poses the one ore more objects' determined shapes are, and which reference pose is the closest ones to the one or more determined shapes.

In some examples, the continuous user interface feedback is graphical in nature.

In some examples, the continuous user interface feedback is auditory in nature.

In some examples, the continuous user interface feedback is haptic in nature.

In some examples, the sensor subsystem for detecting objects can be in a powered-up or a powered-down state.

In some examples, the input device is a touch screen.

In some examples, the input device is a hardware button.

In some examples, the object is a hand.

In some examples, the object is a face.

In some examples, the sensor subsystem for detecting objects includes two cameras in a stereoscopic configuration.

In some examples, the camera includes infrared time-of-flight technology for measuring distance on a per-pixel basis.

In some examples, the reference poses include facial expressions.

In some examples, the reference zones are further restricted to be represented as areas on a two-dimensional plane.

In some examples, the reference zones are further restricted to be represented as points.

In some examples, the set of reference zones and reference poses is not fixed.

In some examples, wherein the state of the computing device is changed as the result of a heuristic comprising change in the state of ongoing gestures and timing of the state changes.

In some examples, the continuous feedback includes highlighting graphical elements.

In some examples, the continuous feedback includes a visual representation of particles.

In some examples, the continuous feedback includes synthesizing sound.

In some examples, the input device is a touch screen.

In some examples, the input device is a hardware button.

In some examples, the one or more processors only process input from the sensor subsystem for detecting objects when at least one finger is making contact with the touch screen.

In some examples, the one or more processors only process input from the sensor subsystem for detecting objects after at least one finger has made contact with the touch screen.

In some examples, at least one of the programs running on one or more of the one or more processors includes instructions for only processing input from the sensor subsystem for detecting objects when a heuristic determines that at least one finger have been identified as performing a gesture on the touch screen that may be continued as a gesture in the vicinity of the device.

In some examples, one or more processors only process input from the sensor subsystem for detecting objects when the hardware button is being actuated.

In some examples, one or more processors only process input from the sensor subsystem for detecting objects after the hardware button has been actuated.

In some examples, the reference zones are further restricted to be represented as points.

In some examples, the set of reference zones and reference poses can be altered using machine-learning techniques.

In some examples, the particles' motion vectors are affected in a wind-like manner by the change in the one or more determined positions.

In some examples, the sensor subsystem for detecting objects is in a powered-up state only when at least one finger is held on the touch screen.

In some examples, the sensor subsystem for detecting objects is set to a powered-up state when at least one finger is held on the touch screen.

In some examples, at least one of the programs running on one or more of the one or more processors includes instructions for setting the sensor subsystem for detecting objects to a powered-up state when a heuristic determines that at least one finger have been identified as performing a gesture on the touch screen that may be continued as a gesture in the vicinity of the device.

In some examples, the sensor subsystem for detecting objects is in a powered-up state only when the hardware button is actuated.

In some examples, the sensor subsystem for detecting objects is set to a powered-up state when the hardware button is actuated.

In some examples, the at least one finger is making contact with at least one defined area of the touch screen.

In some examples, the particles are appearing to affect one or more user interface elements directly.

In some examples, the at least one finger is held within at least one defined area of the touch screen.

In accordance with another example, there is provided a computer-implemented method, comprising: at a computing device with a user interface and a sensor subsystem for detecting objects, applying one or more heuristics for determining position and shape of zero, one or more objects using the sensor subsystem for detecting objects, matching at least one of: detected object positions with a set of reference zones, and detected shapes with a set of reference poses.

In some examples, when at least of one of: one or more detected object shapes is recognized as a different reference pose than before, and one or more detected object positions is recognized as being in a different reference zone than before, changing a state of the computing device.

In some examples, when at least of one of one or more detected object shapes lie between the reference object shapes, and one or more detected object positions lie between the reference object positions, giving feedback, through cues in the user interface, of what will happen if the current change in object shapes and/or object positions of the one or more detected object is progressed further.

In some examples, the computing implemented method comprises a touch screen input device, including: when a heuristic determines that user activity on the touch screen input device could be followed by a gesture in the vicinity of the device, activating the sensor subsystem for detecting objects.

CLAIMS

1. A computing device, comprising:

a display;

a sensor subsystem for detecting off-surface objects;

memory having instructions stored on it which when run on a processor, causes the processor to perform the steps of:

detecting an off-surface object using the sensor subsystem;

determining a position and a shape of said detected off-surface object;

matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state;

detecting changes in the determined position and shape using the sensor subsystem;

matching said changes in the determined position with transitions between said reference zones and matching said changes in determined shapes with transitions between said reference poses to thereby determine a transition between gesture states; and,

instructing a user interface to provide feedback based on the determined transition between gesture states.

2. The computing device of claim 1, further comprising a means for detecting user input.

3. The computing device of claim 2, wherein the processor performs the further step of:

setting the sensor subsystem to a powered-on state when the means for detecting a user input has detected a predetermined user input.

4. The computing device of claim 2 or 3, wherein the means for detecting a user input is a touch screen.

5. The computer device of claim 4, wherein the processor performs the further steps of:

detecting an on-surface gesture via the touch screen; and,

determining a compound gesture using a combination of the detected on-surface gesture and an off-surface gesture determined using the sensor subsystem.

6. The computing device of claim 2 or 3, wherein the means for detecting user input is a hardware button.

7. The computing device of any preceding claim, wherein the feedback is at least one of graphical, auditory and haptic in nature.

8. The computing device of any preceding claim, wherein the feedback is a continuous feedback.

9. The computing device of claim 8, wherein the processor performs the further step of:

determining a degree of transition between gesture states and providing the continuous feedback based on said degree of transition.

10. The computing device of claim 9, wherein the degree of transition includes at least one of: where in between the reference zones the determined positions are present, where in between the reference poses the determined shapes are, which reference zone the determined position is closest and which reference pose the determined shape is closest.

11. The computing device of any preceding claim, wherein the reference zones are represented as at least one of volumes in a three-dimensional space, points in a three-dimensional space, areas in a two-dimensional space and points in a two-dimensional space.

12. The computing device of any preceding claim, wherein the sensor subsystem includes at least one of a single camera and two cameras in stereoscopic configuration.

13. The computing device of any preceding claim, wherein the set of reference zones and the set of reference poses are changeable.

14. A computer-implemented method, for operating a computing device comprising a display, sensor sub system, processor and memory, comprising the steps of:

detecting an off-surface object using the sensor subsystem;

determining a position and a shape of said detected off-surface object;

matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state;

detecting changes in the determined position and shape using the sensor subsystem;

matching said changes in the determined position with transitions between said reference zones and matching said changes in determined shapes with transitions between said reference poses to thereby determine a transition between gesture states; and,

instructing a user interface to provide feedback based on the determined transition between gesture states.

15. A computer program product having instructions which when run on a processor causes the processor to carry out the method of claim 14.

16. A computing device, comprising:

a display;

a sensor subsystem for detecting off-surface objects;

a means for detecting user input;

memory having instructions stored on it which when run on a processor, causes the processor to perform the steps of:

detecting an off-surface object using the sensor subsystem;

determining a position and a shape of said detected off-surface object;

matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state;

setting the sensor subsystem to a powered-on state when the means for detecting a user input has detected a predetermined user input.

17. A computer-implemented method for use in a computing device comprising a display, a sensor subsystem, a means for detecting user input , a processor and memory, comprising the steps of:

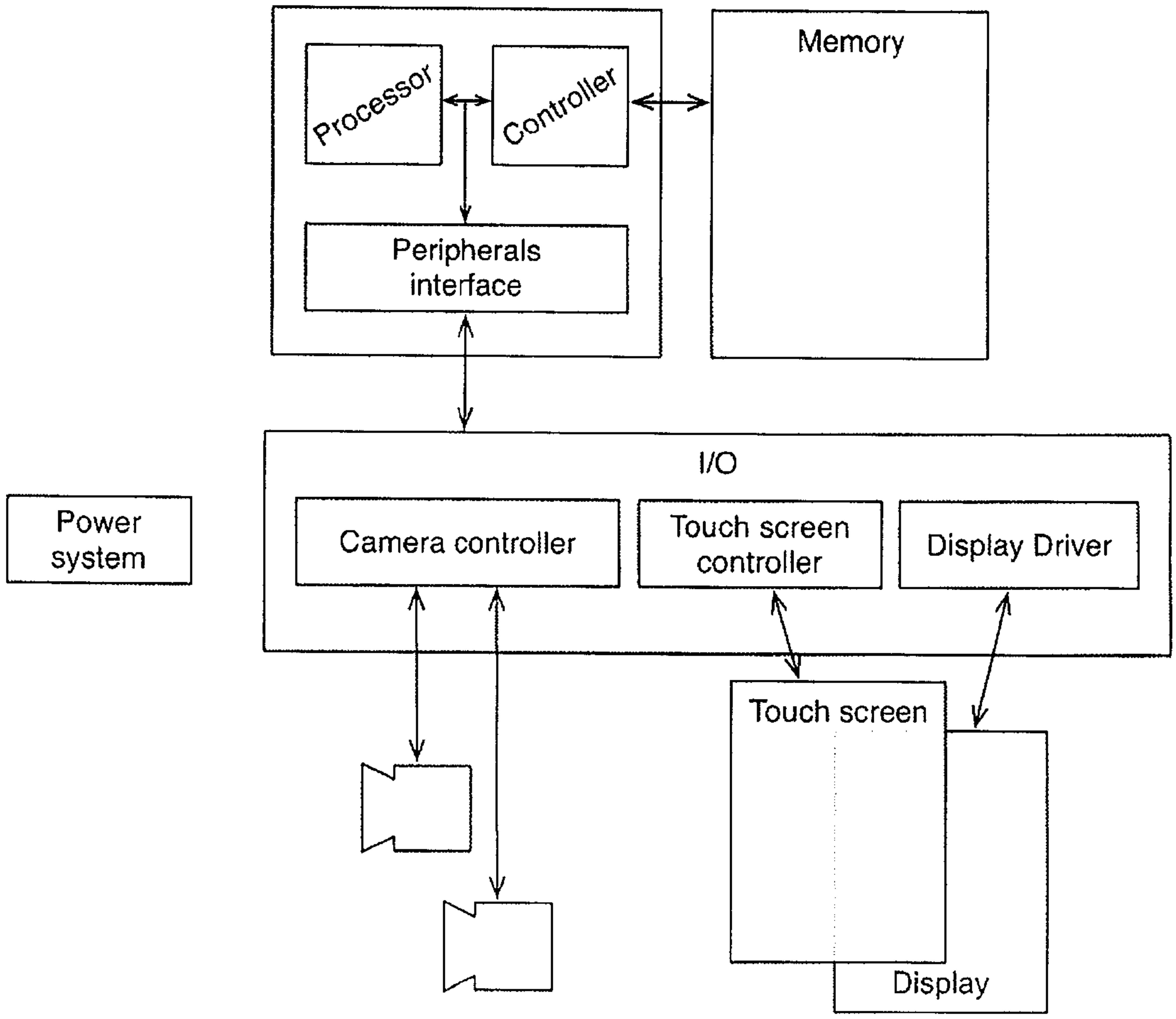
detecting an off-surface object using the sensor subsystem;

determining a position and a shape of said detected off-surface object;

matching the determined position with a set of reference zones and the determined shape with a set of reference poses to thereby determine a gesture state;

setting the sensor subsystem to a powered-on state when the means for detecting a user input has detected a predetermined user input.

18. A computer program product having instructions which when run on a processor causes the processor to carry out the method of claim 17.



Processor
Controller
Memory
Peripherals interface
I/O
Power system
Camera controller
Touch screen controller
Display Driver
Touch screen
Display

Figure 1

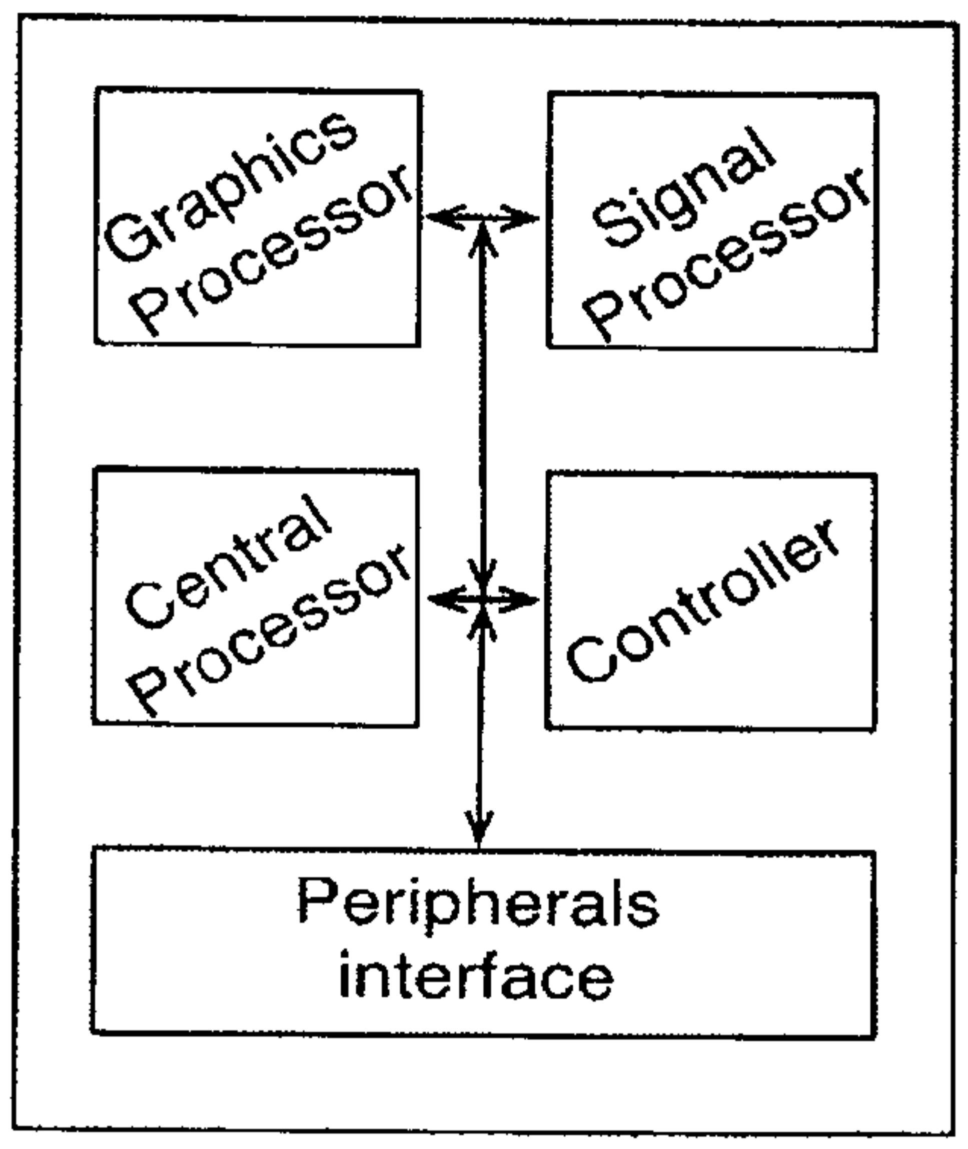
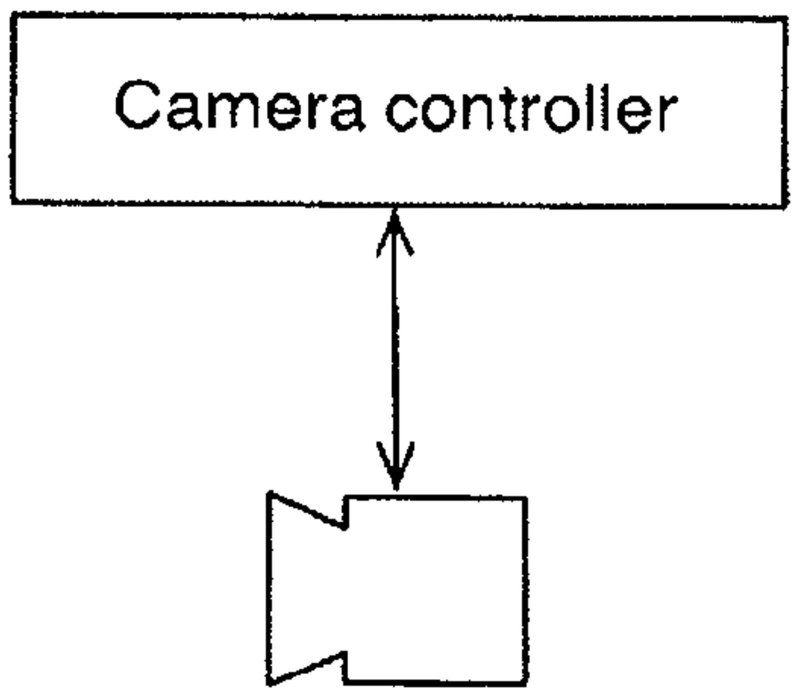

Graphics Processor
Signal Processor
Central Processor
Controller
Peripherals interface

Figure 2

a)



b)

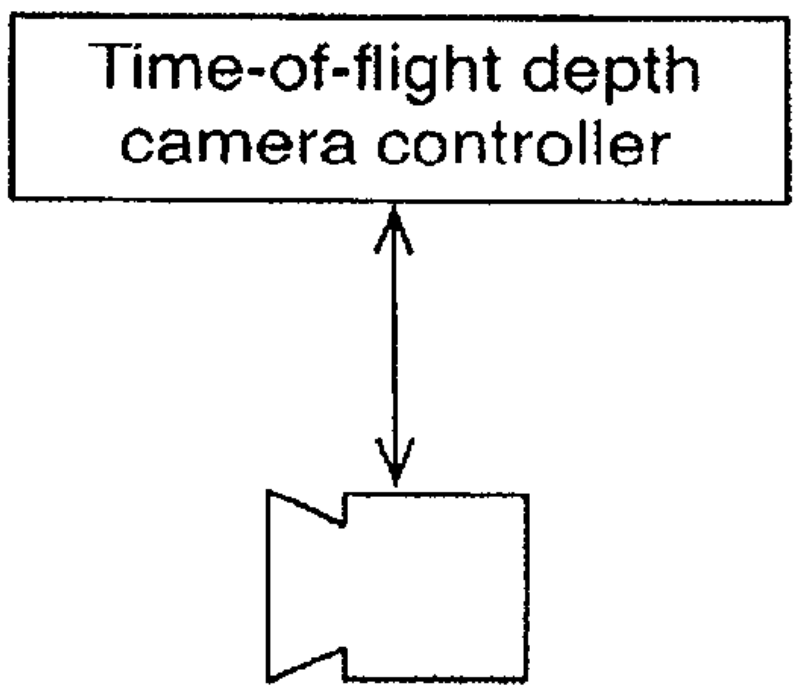


Figure 3

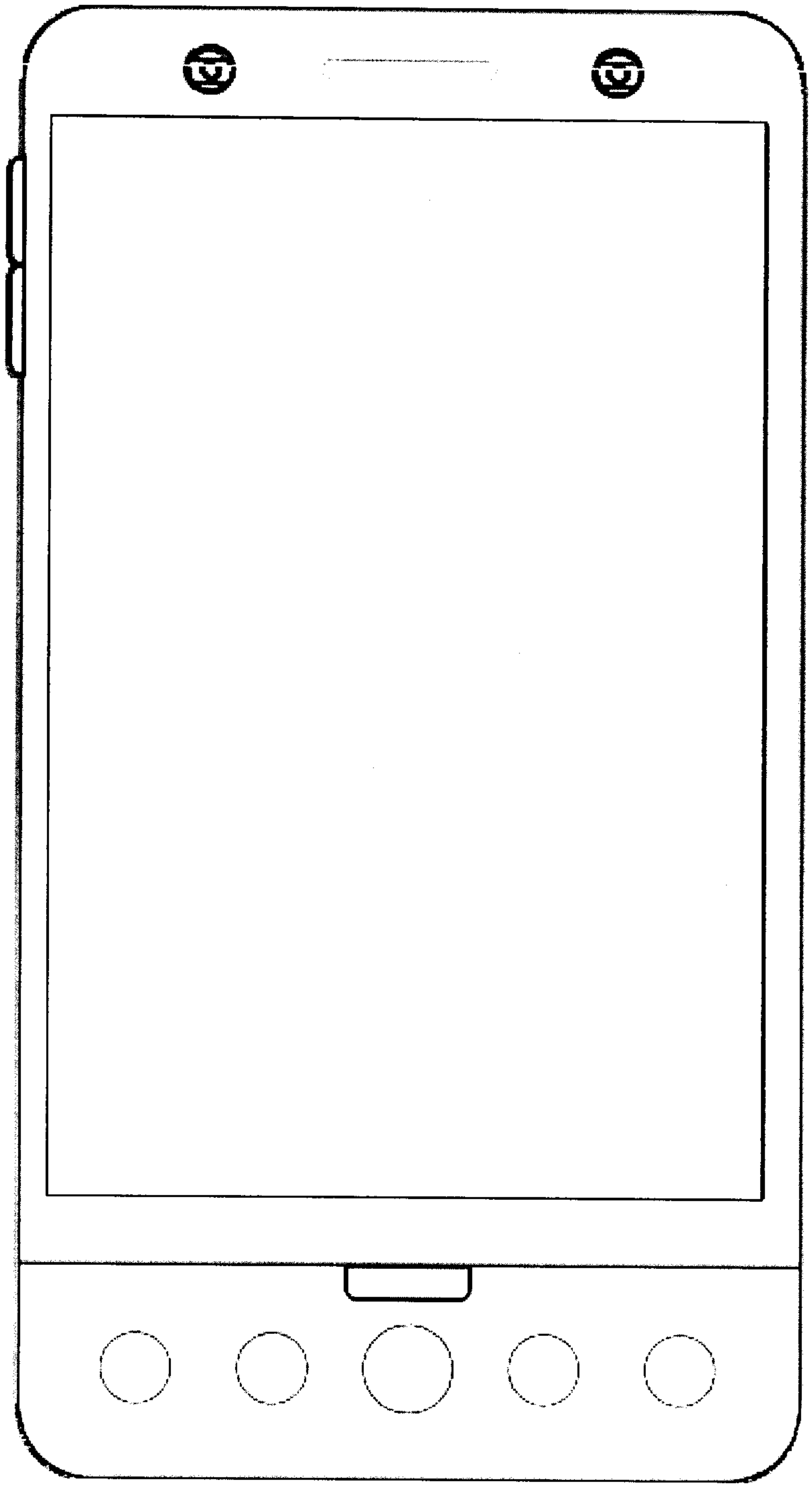


Figure 4

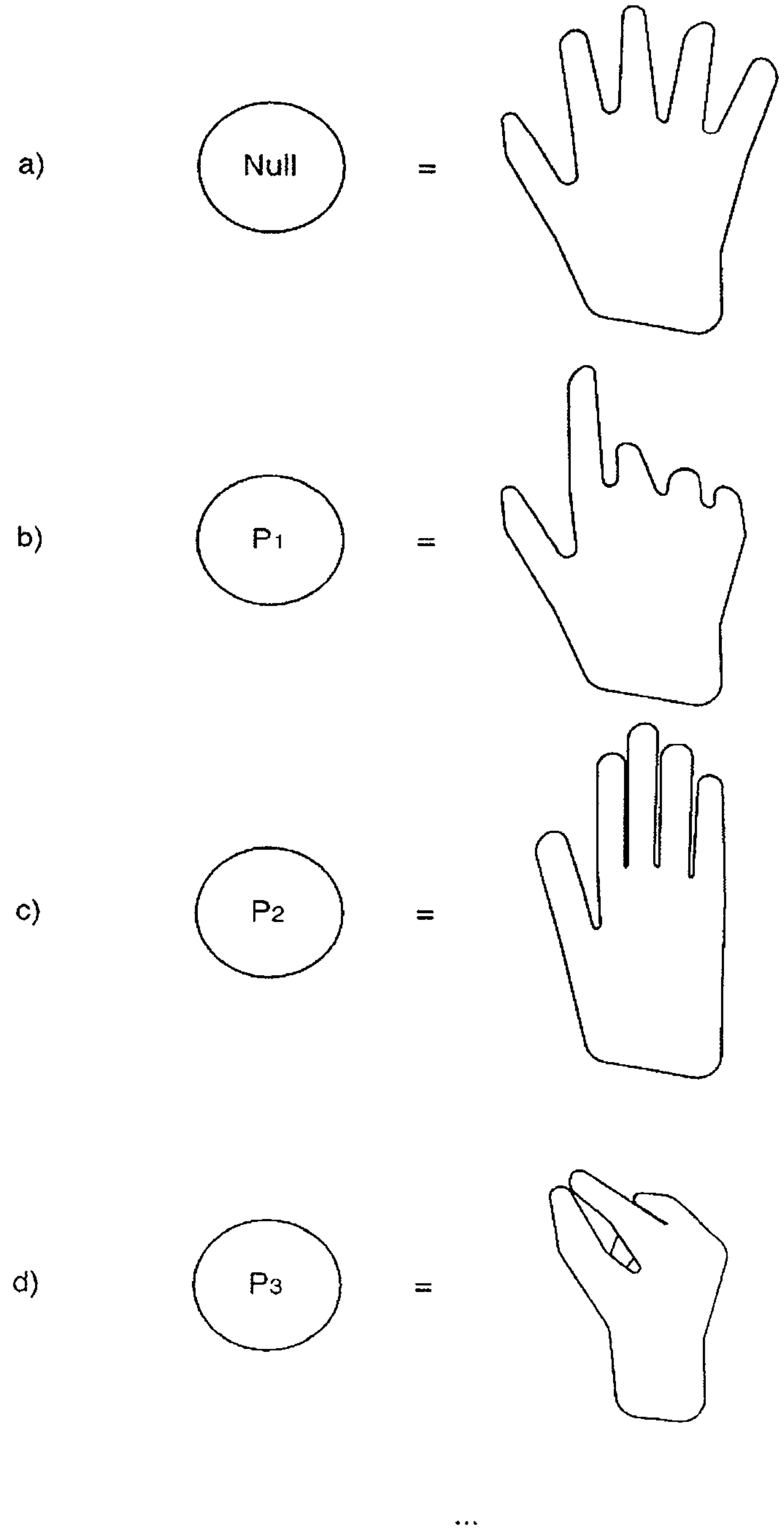
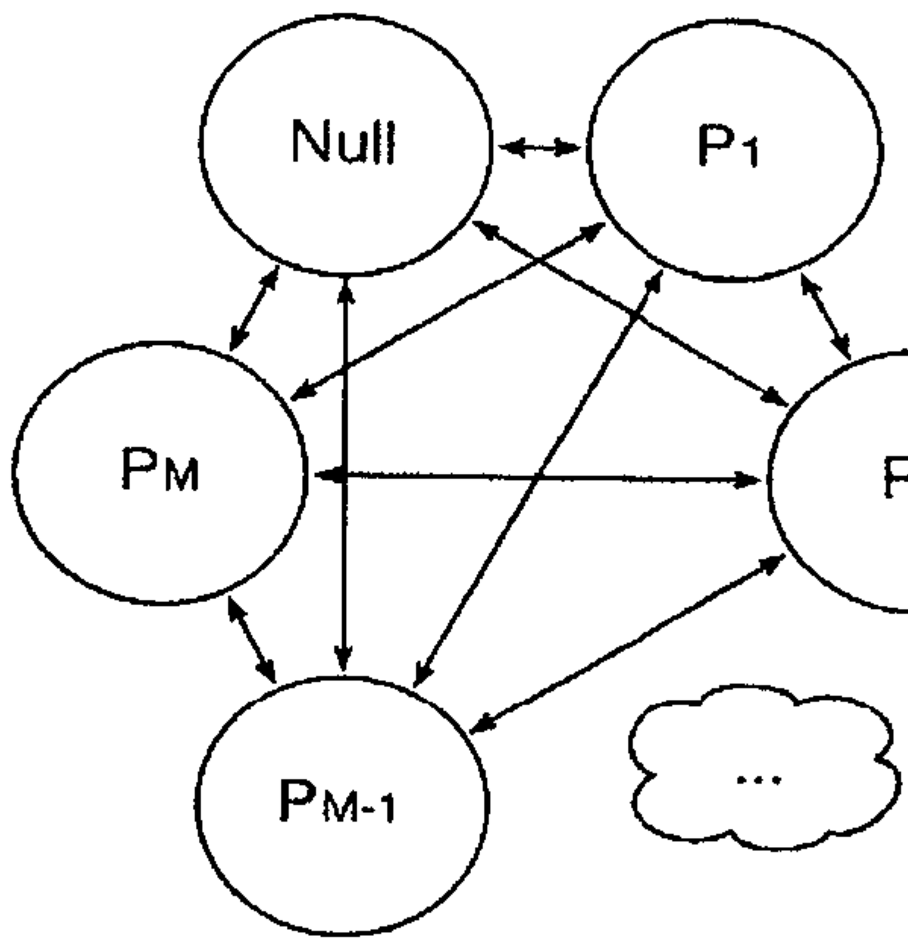
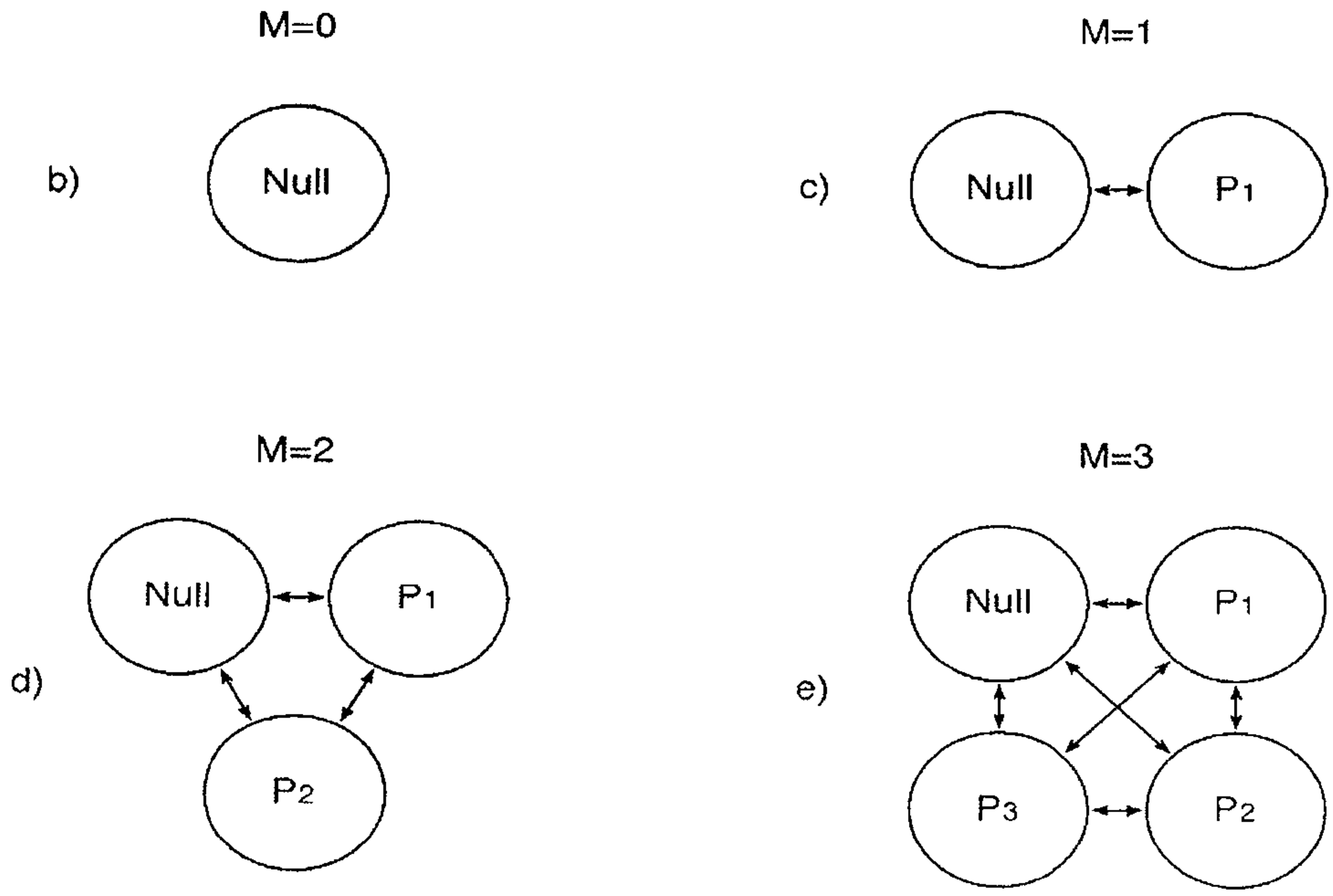


Figure 5

a)


Null
P1
PM
P2
PM-1
...

...


M=0
b)
Null
M=1
c)
Null
P1
M=2
d)
Null
P1
P2
M=3
e)
Null
P1
P3
P2

...

Figure 6

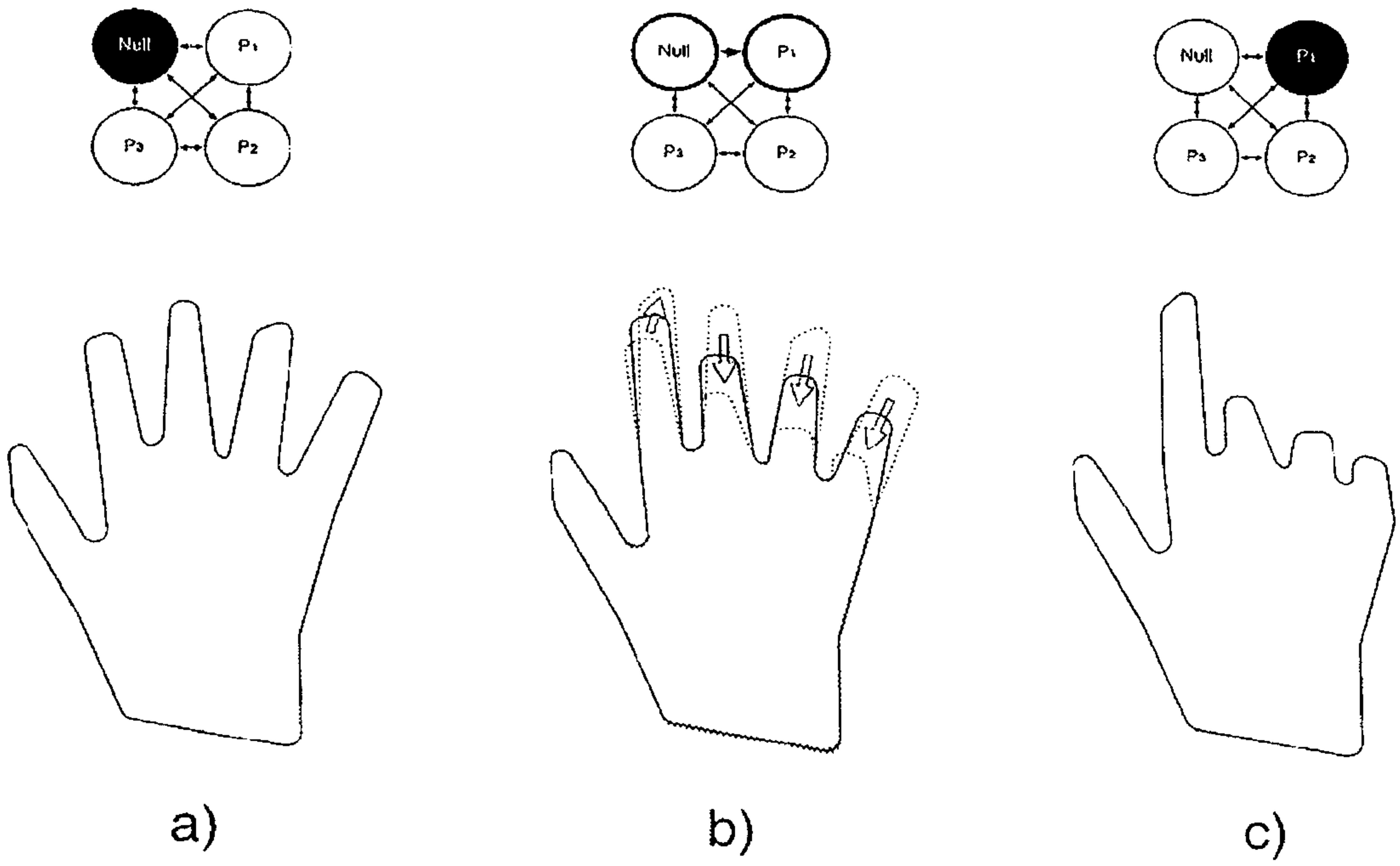


Figure 7

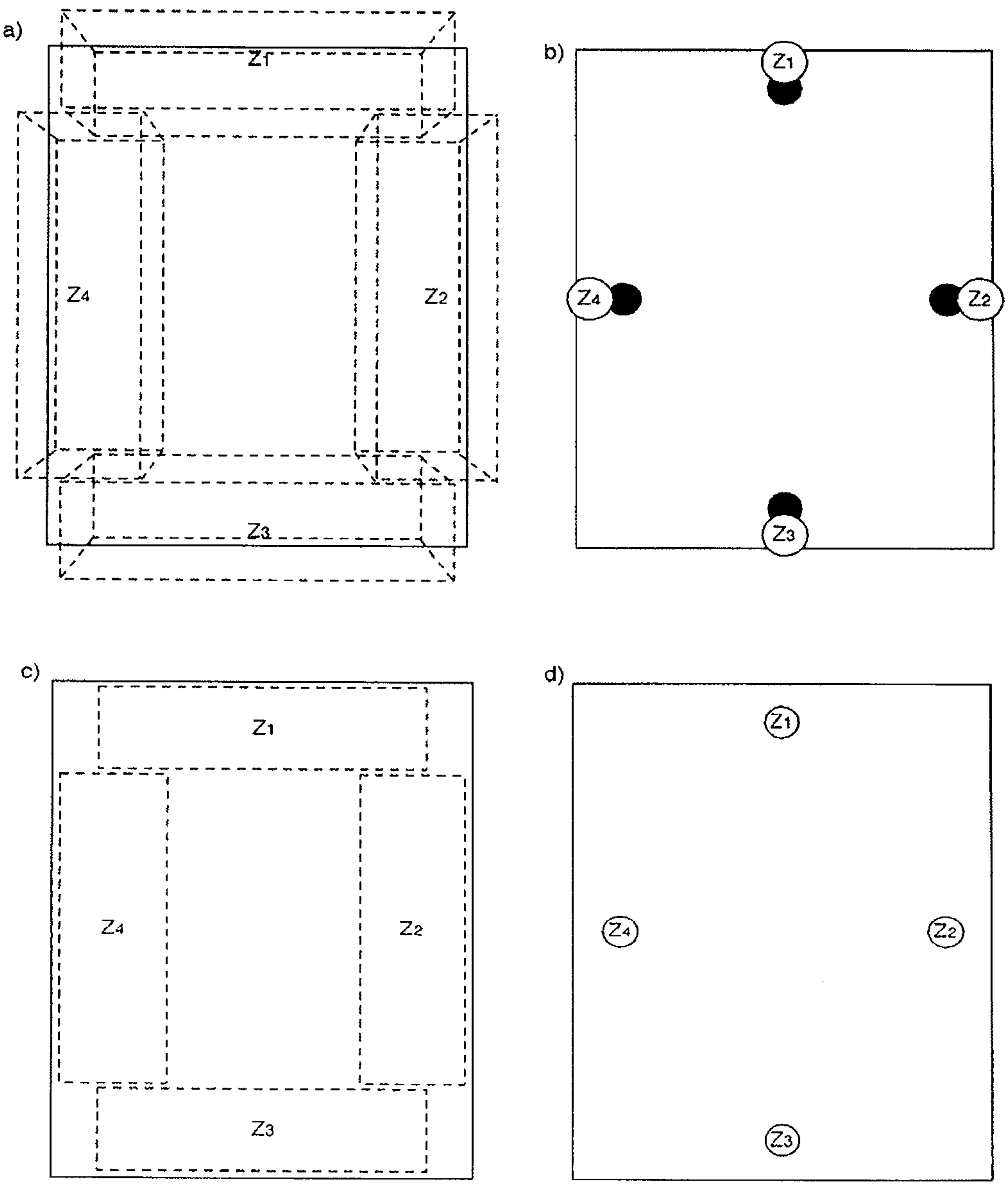


Figure 8

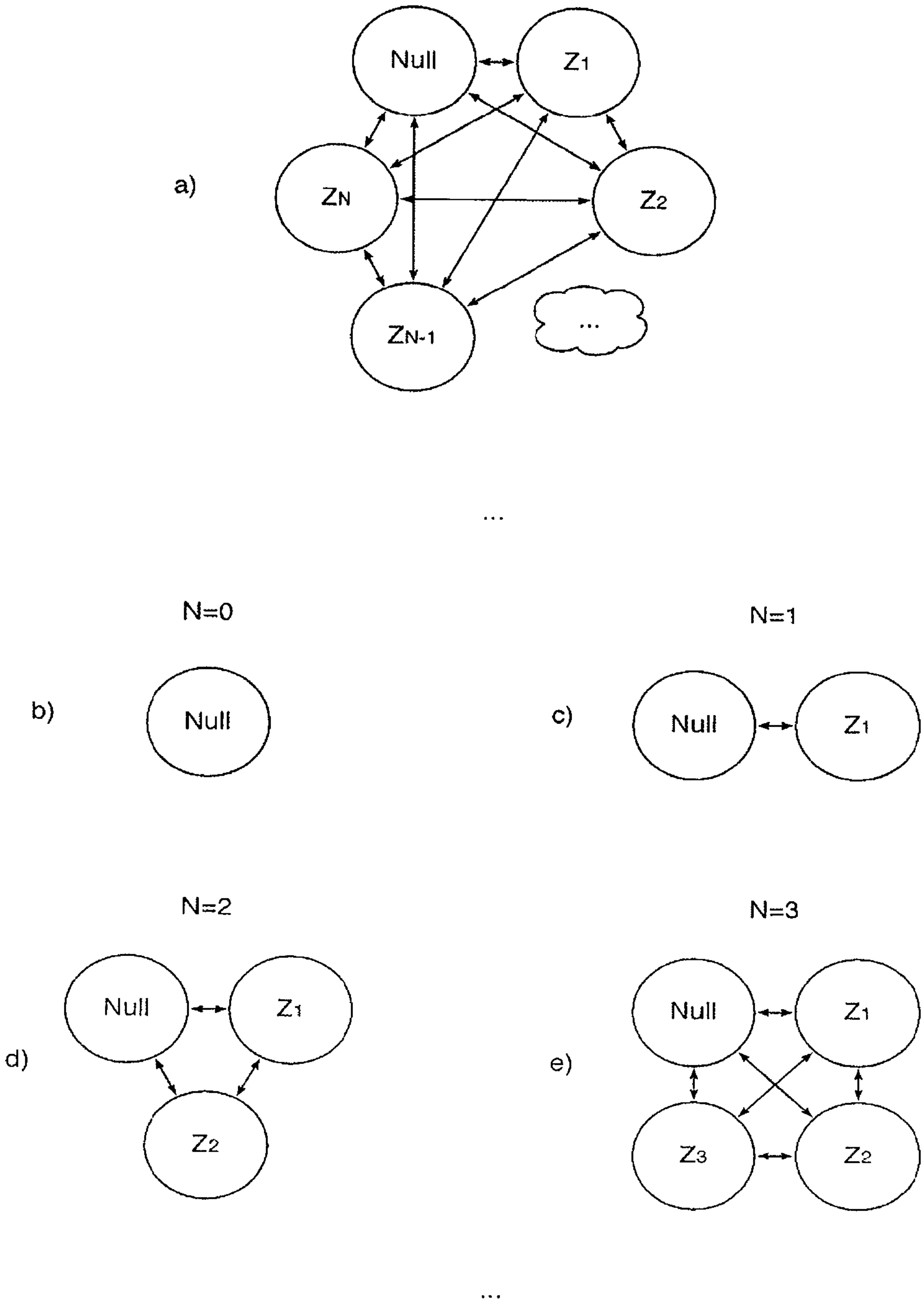

a)
Null
Z1
ZN
Z2
ZN-1
...
...
N=0
b)
Null
N=1
c)
Null
Z1
N=2
d)
Null
Z1
Z2
N=3
e)
Null
Z1
Z3
Z2
...

Figure 9

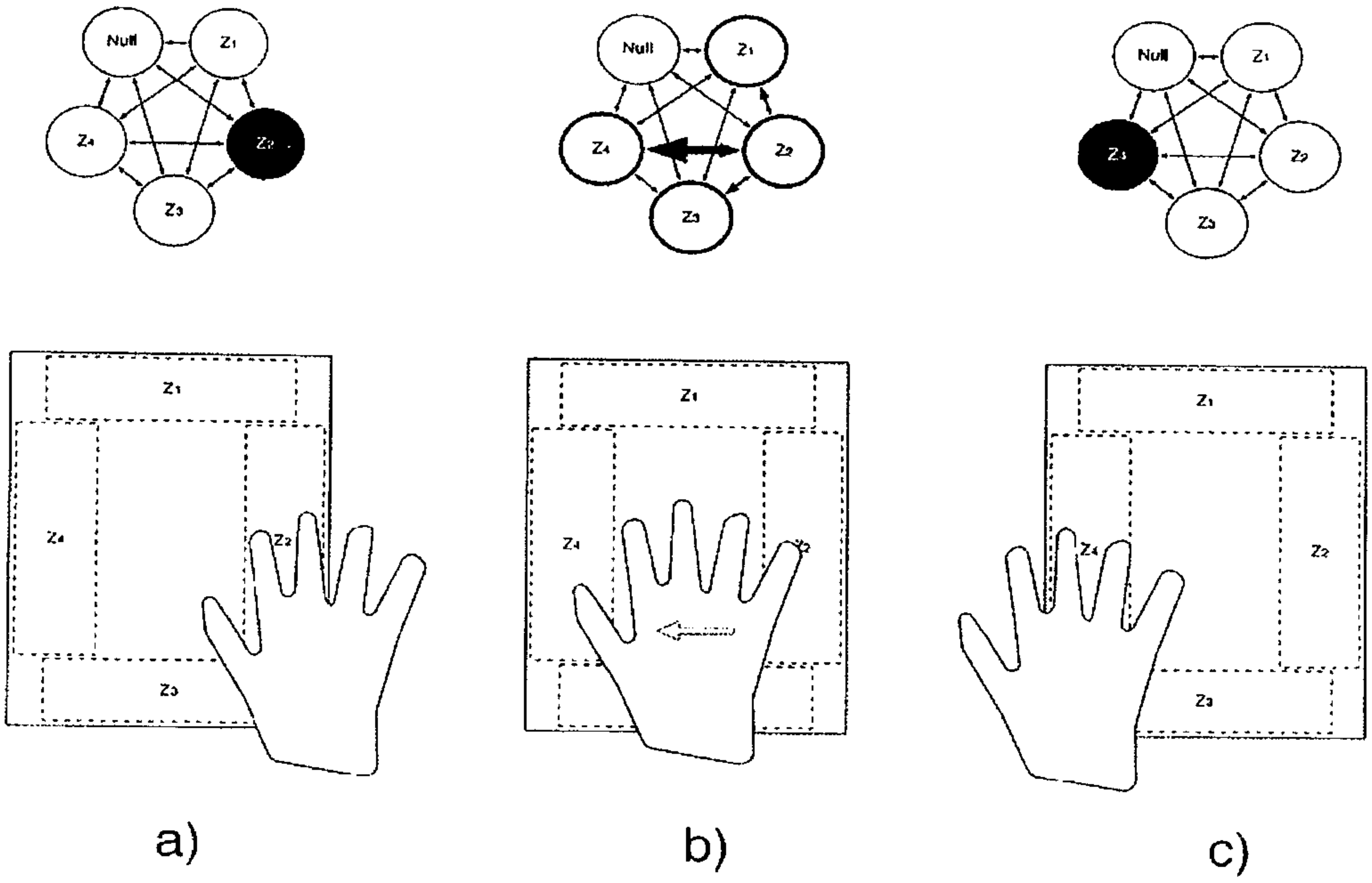


Figure 10

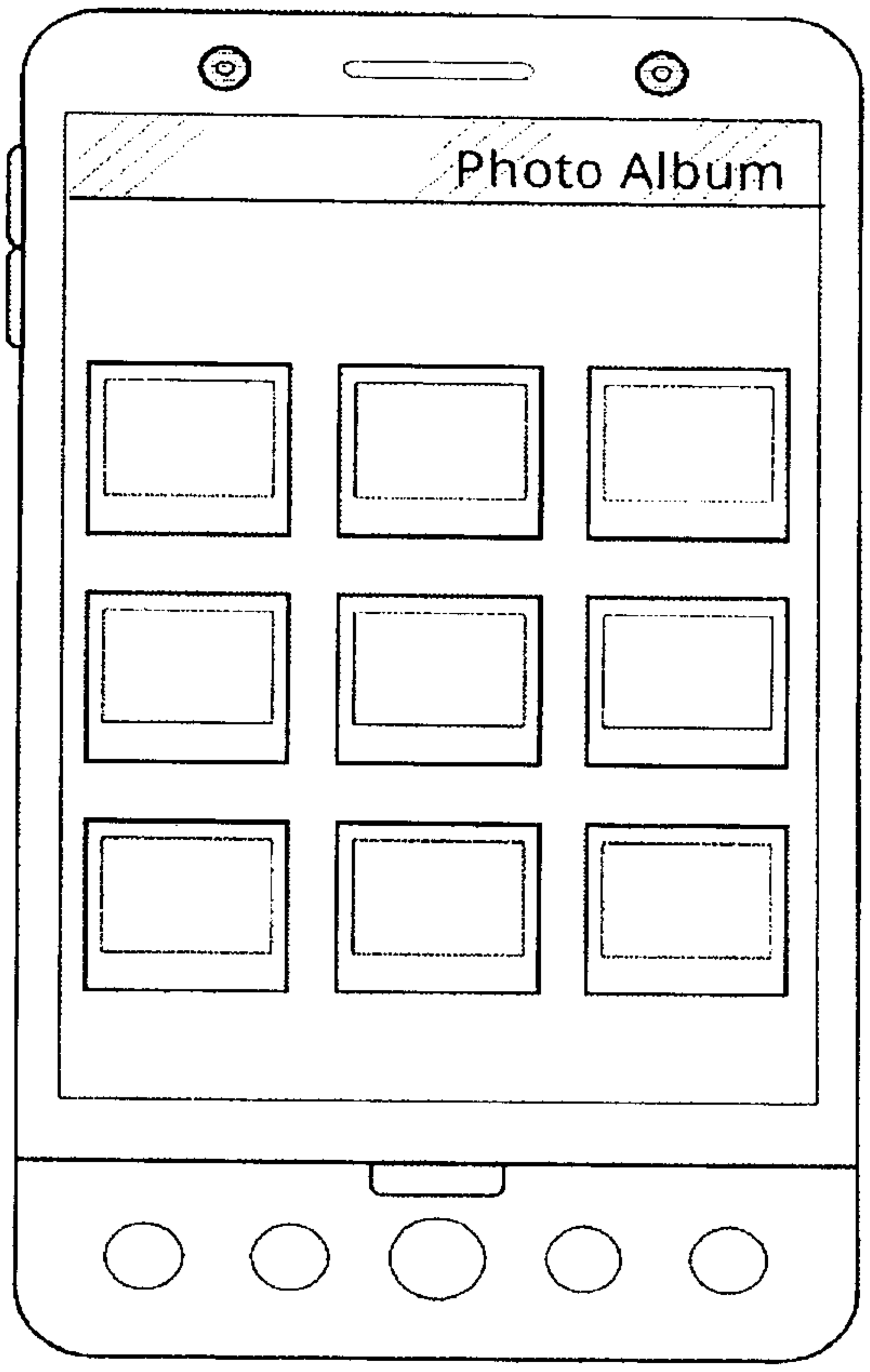


Figure 11a

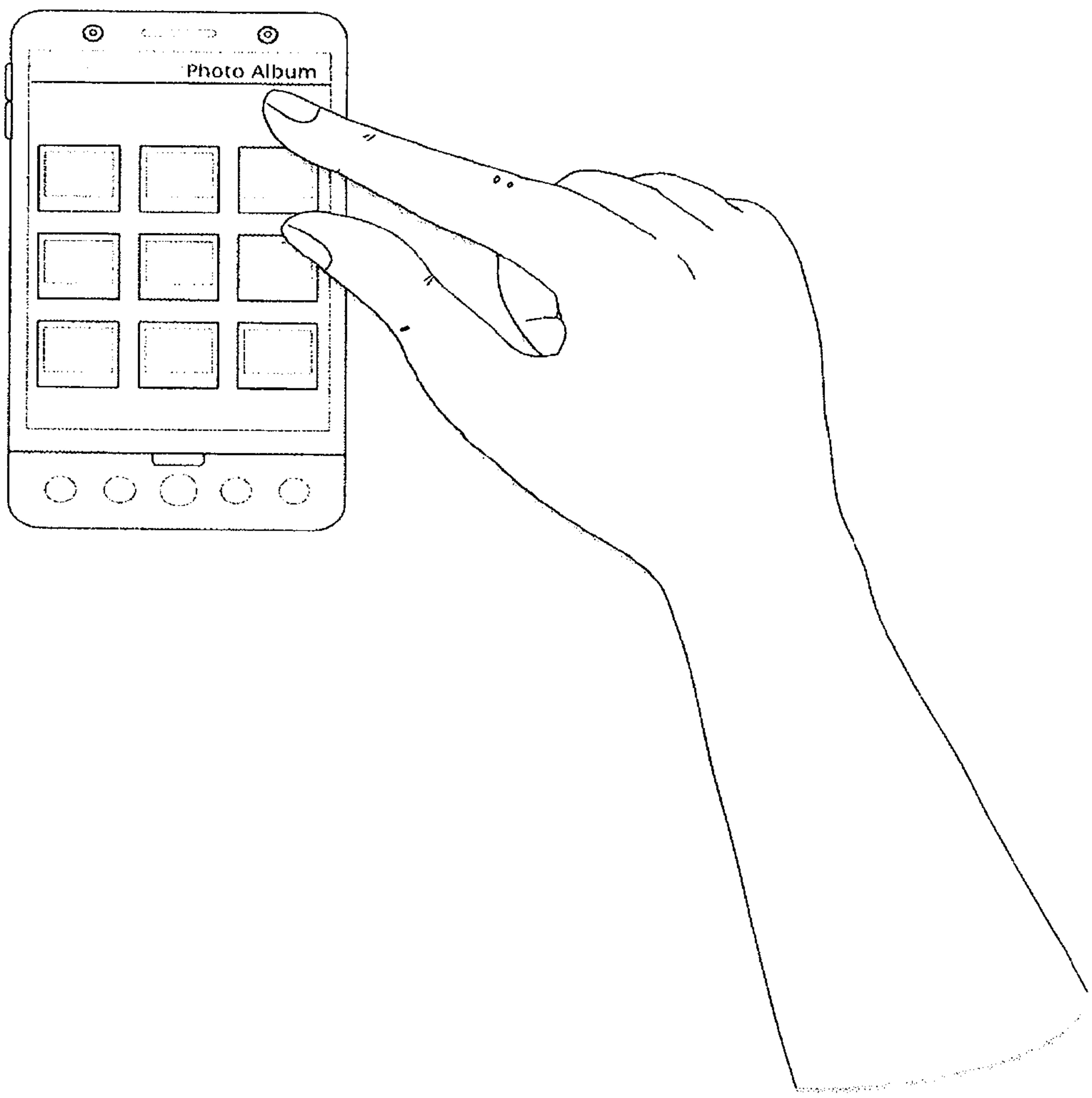


Figure 11b

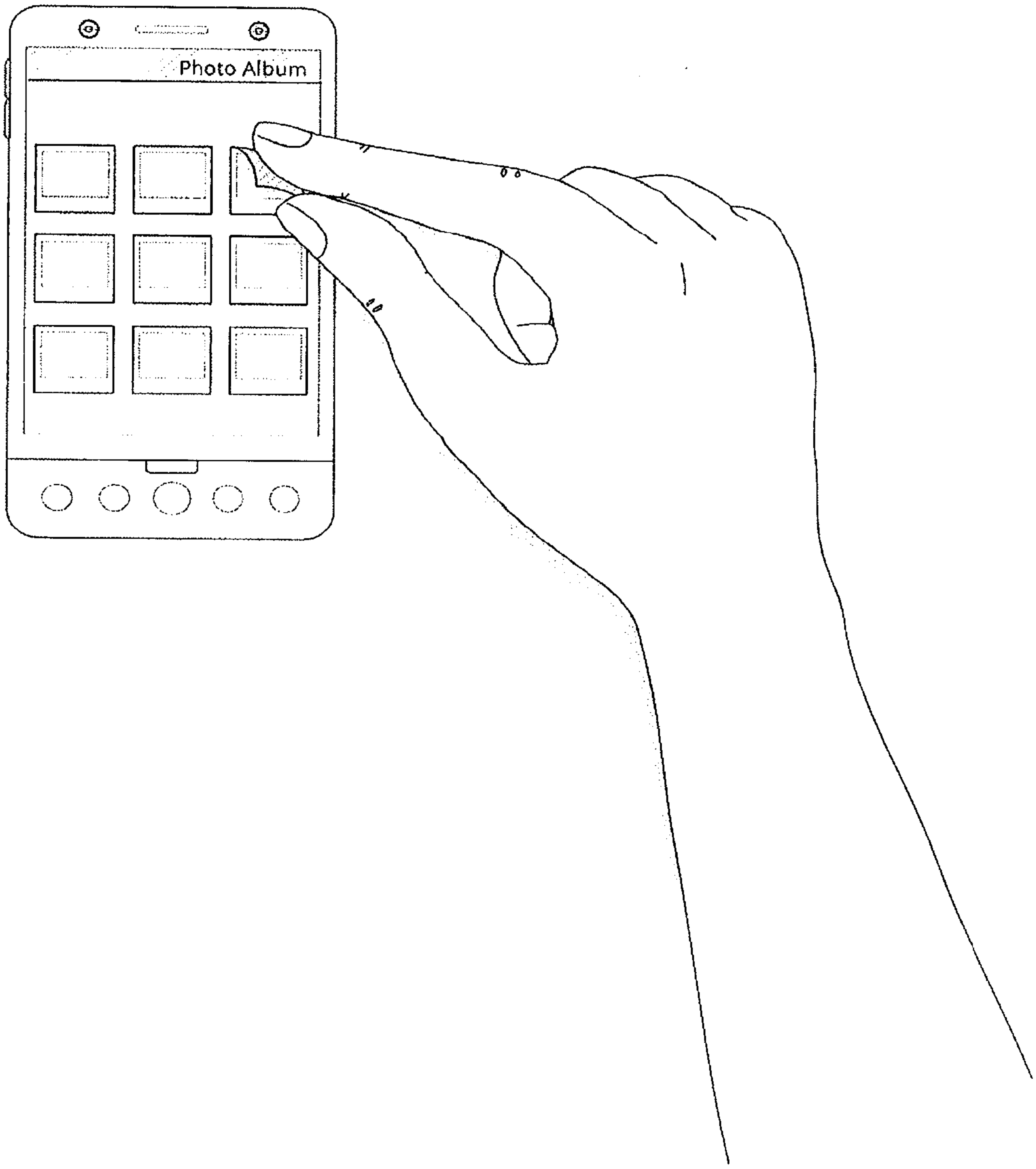


Figure 11c

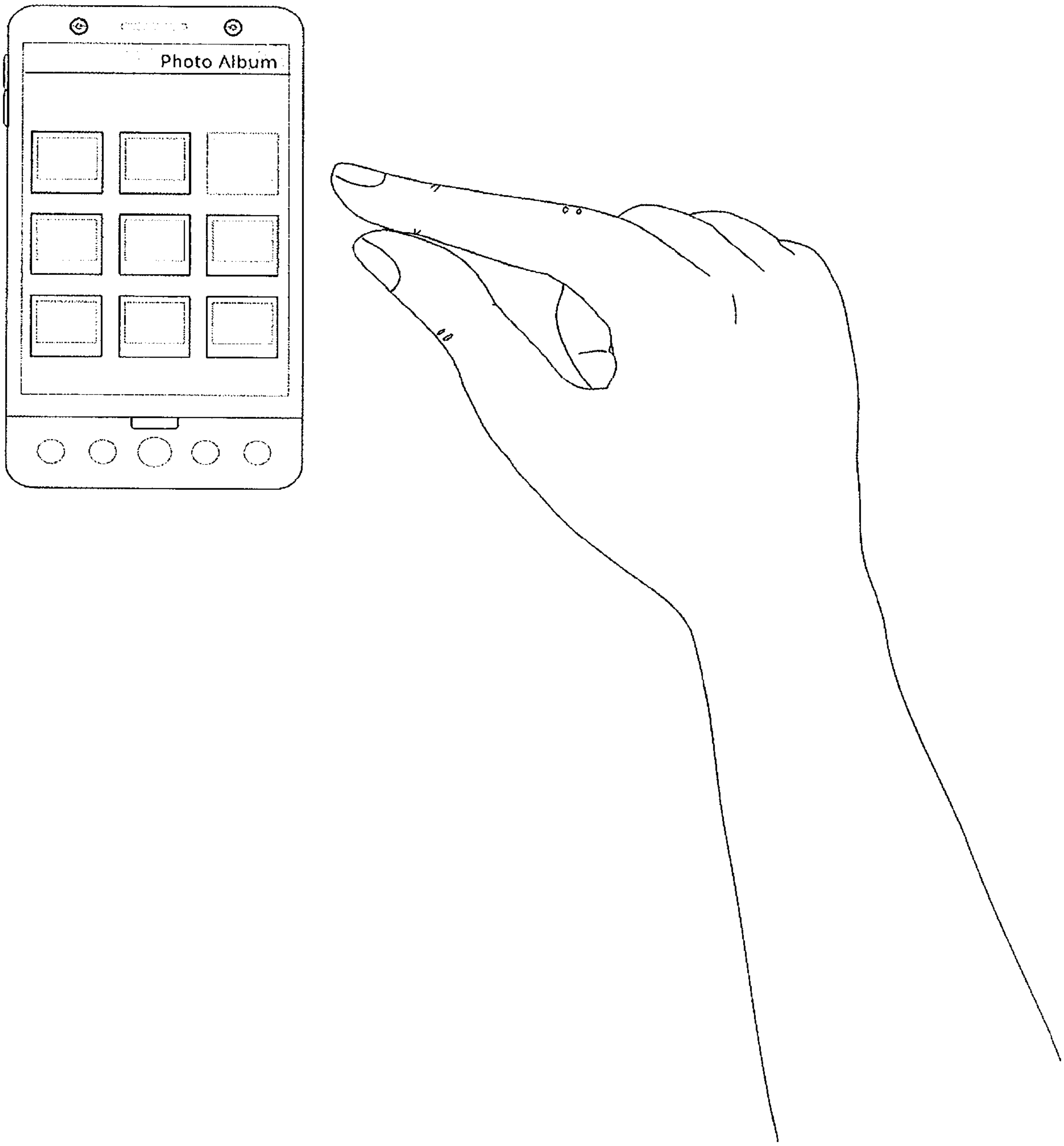


Figure 11d

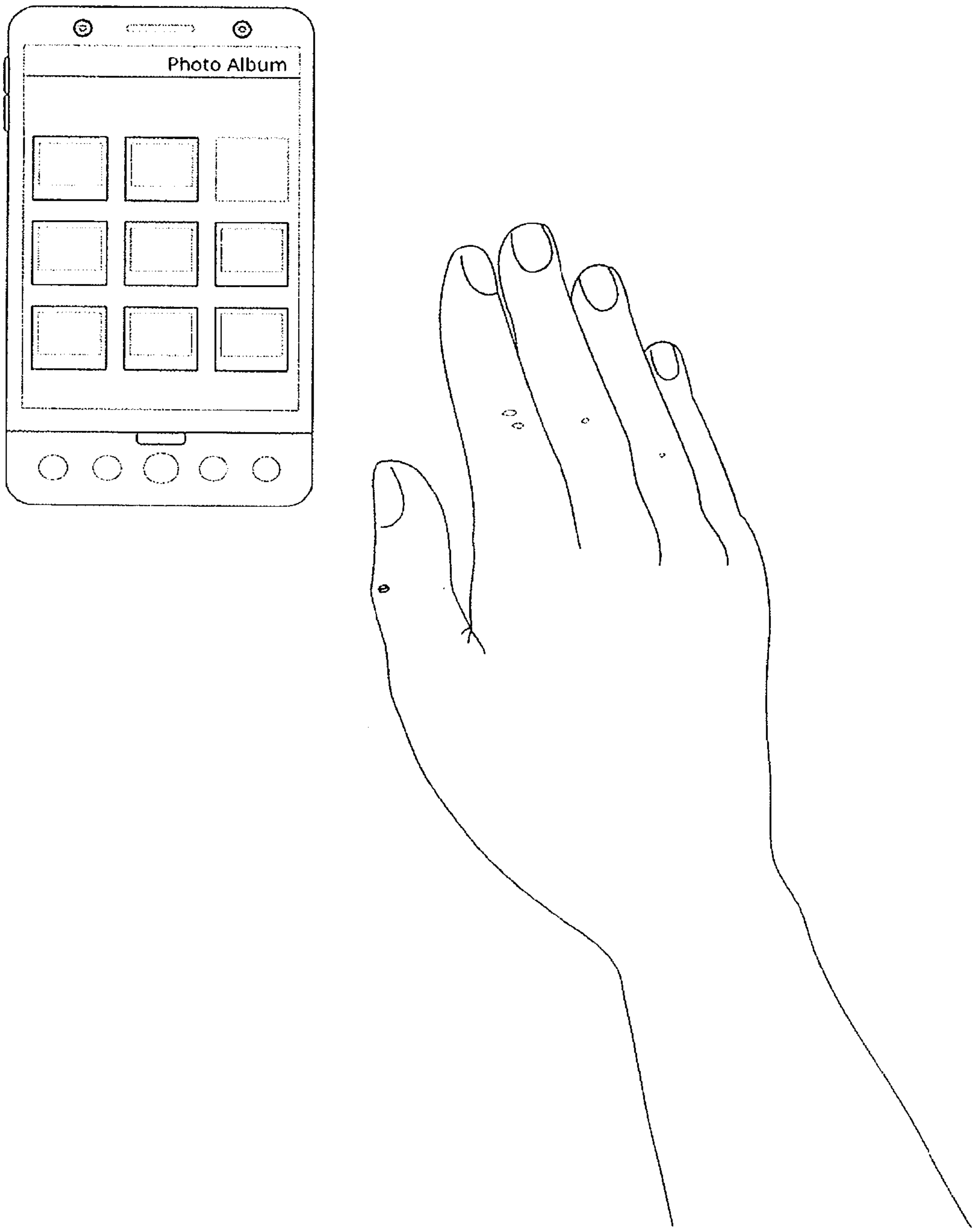


Figure 11e

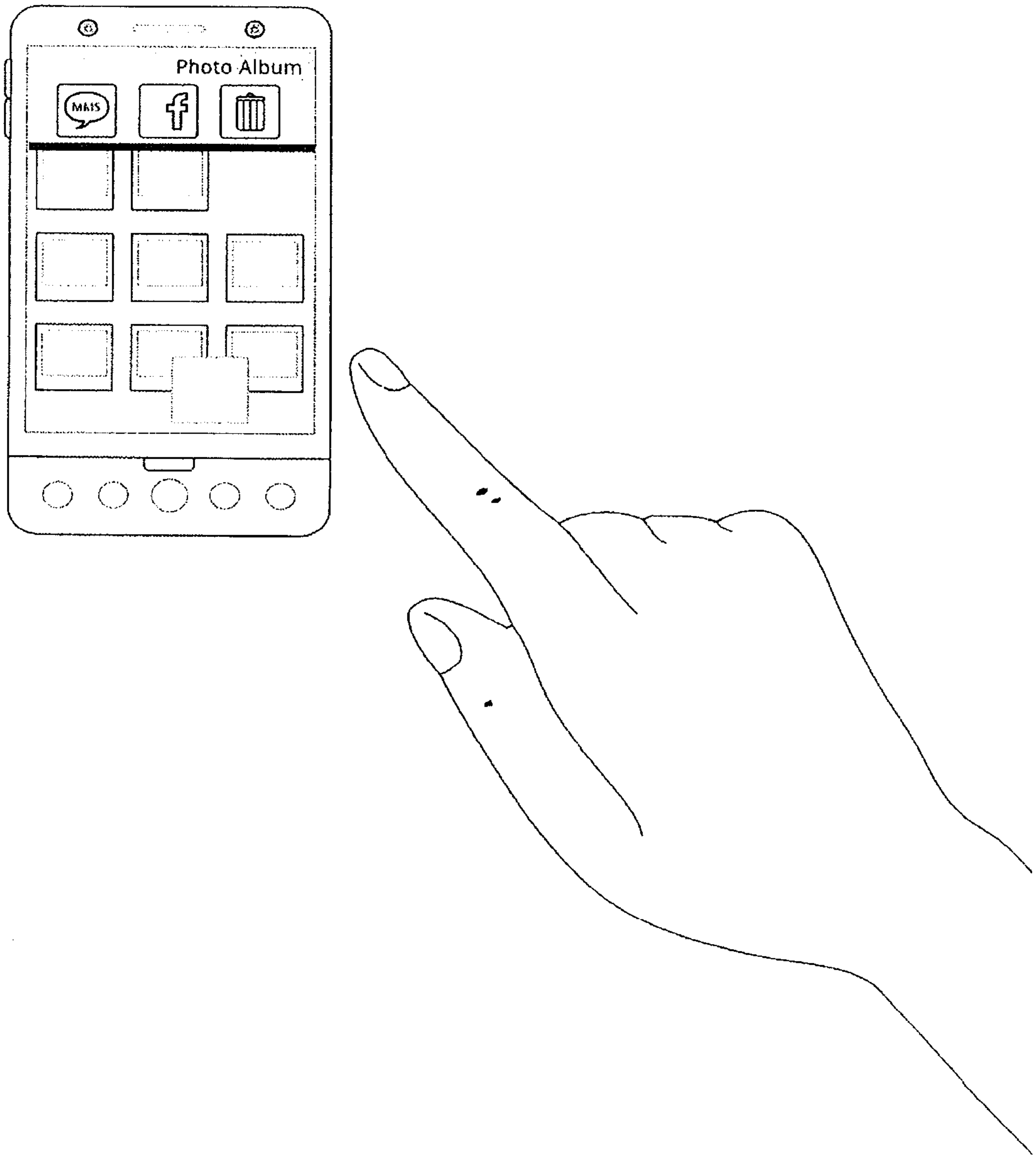


Figure 11f

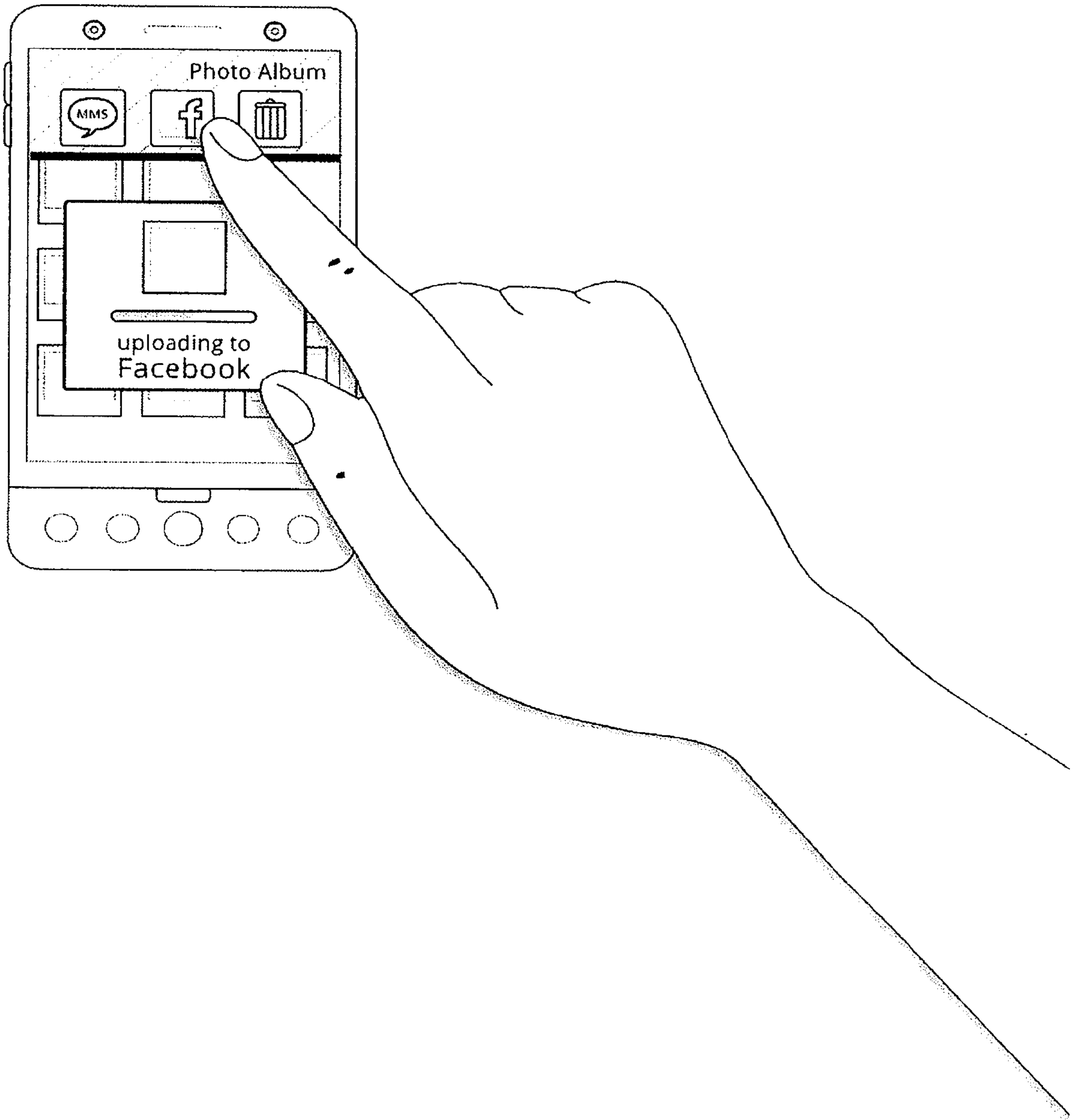


Figure 11g

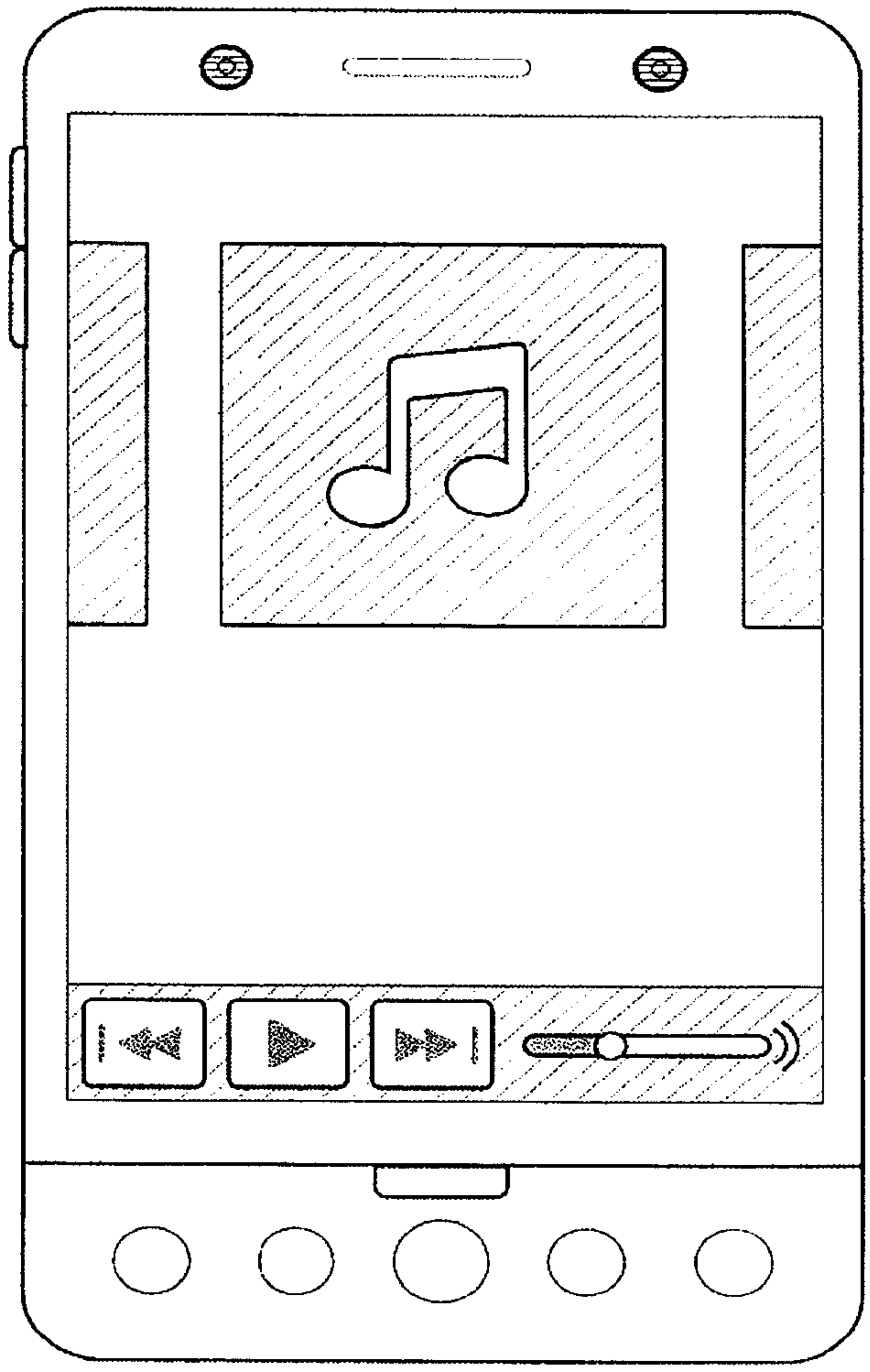


Figure 12a

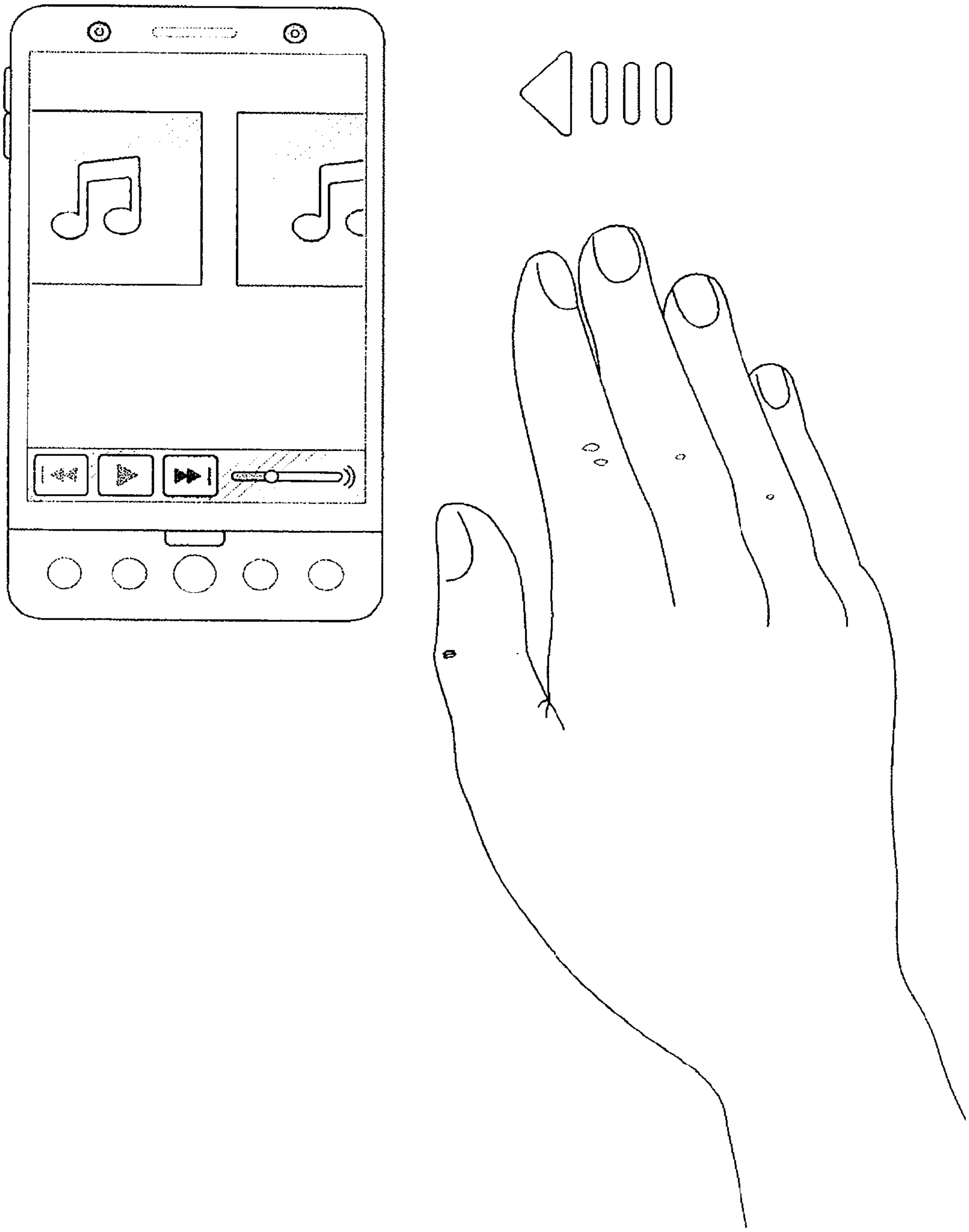


Figure 12b

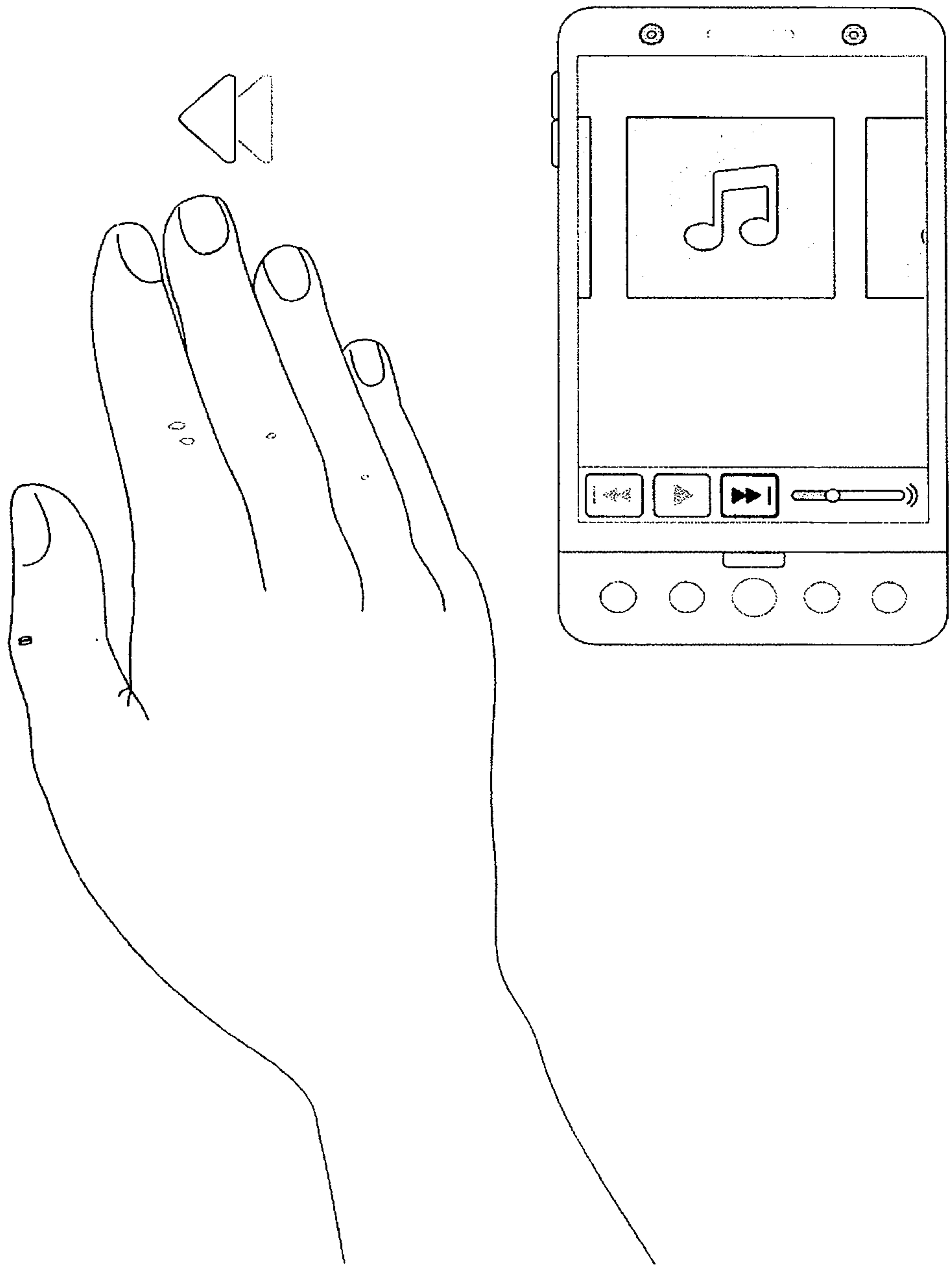


Figure 12c

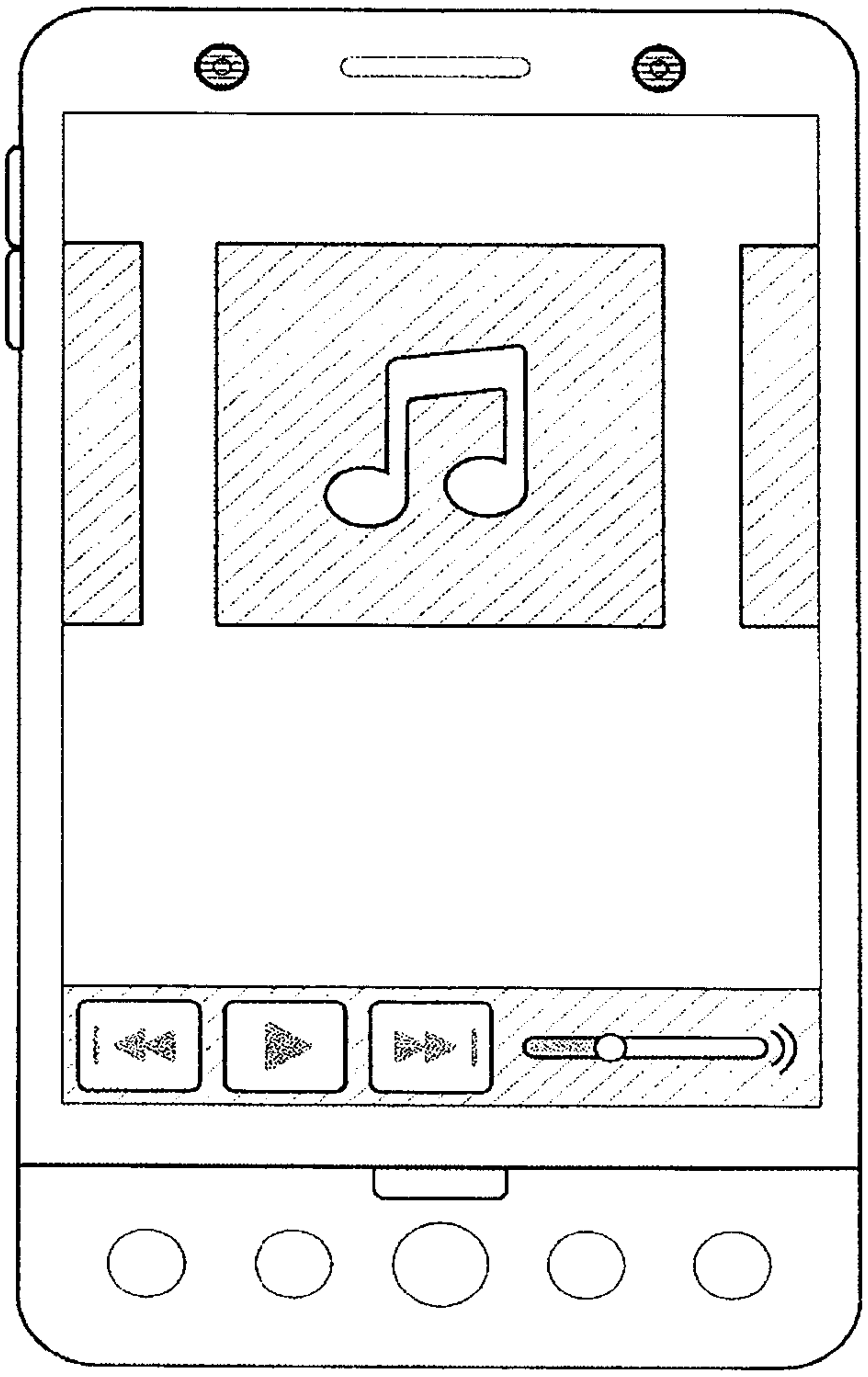


Figure 13a

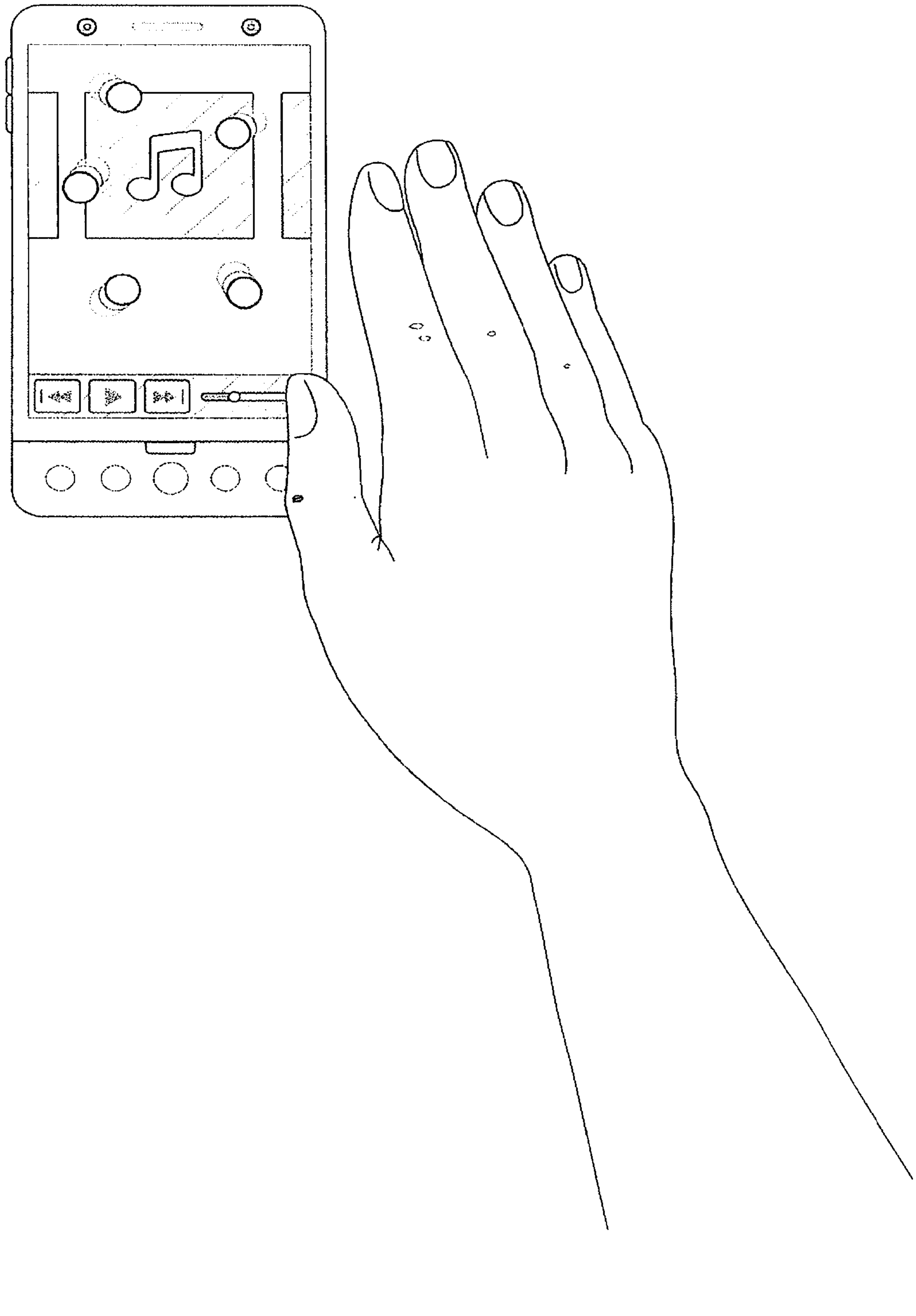


Figure 13b

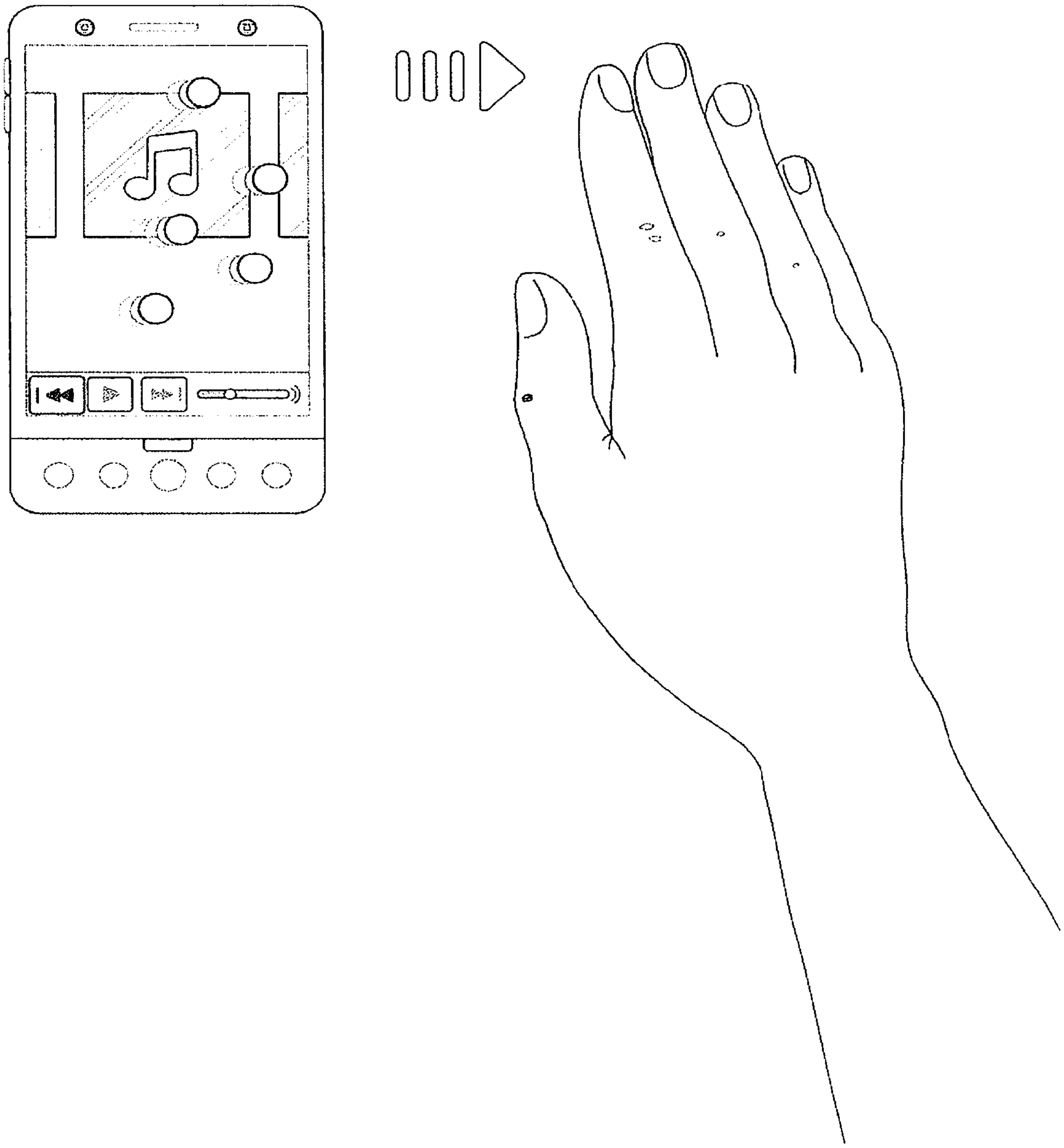


Figure 13c

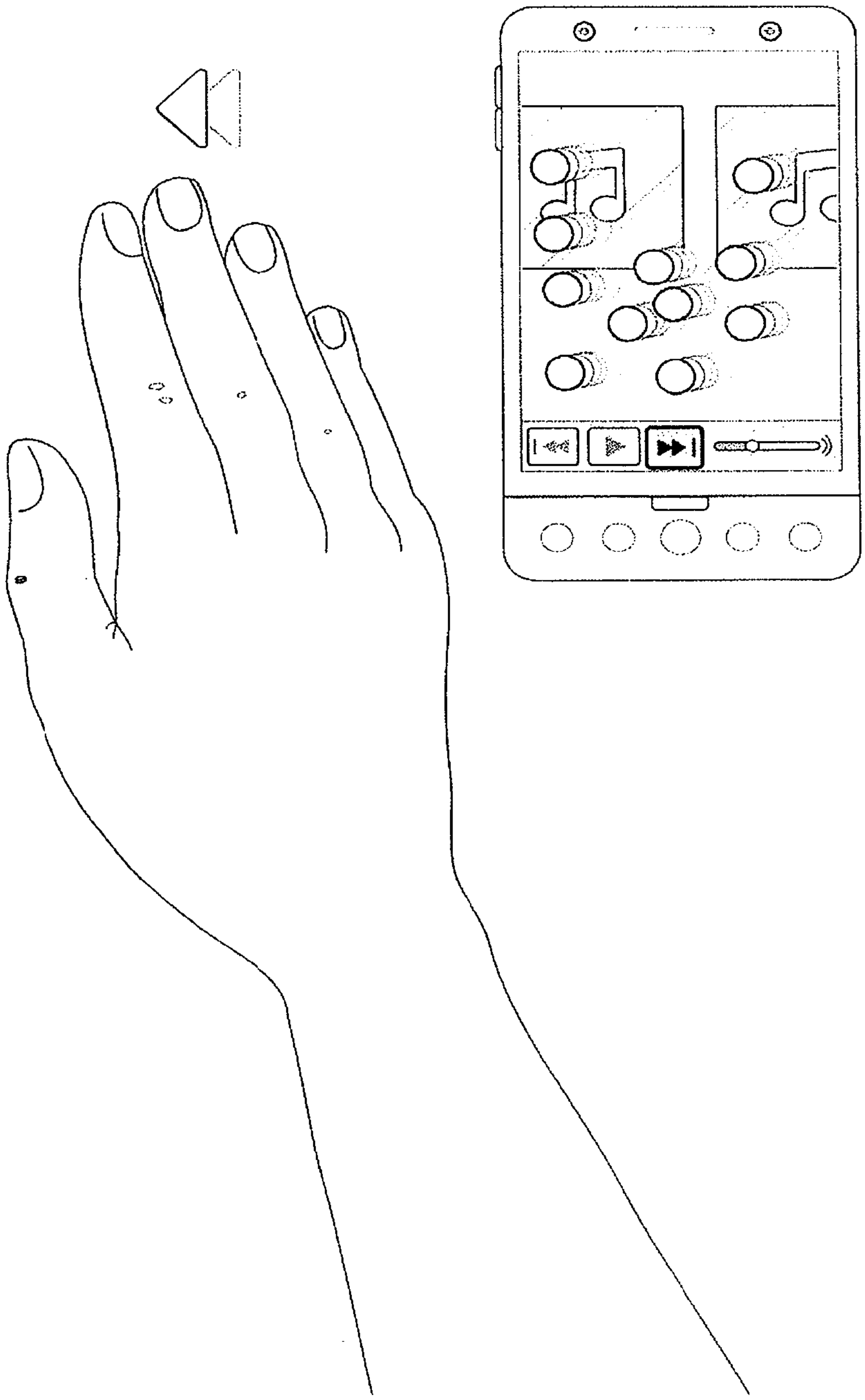


Figure 13d

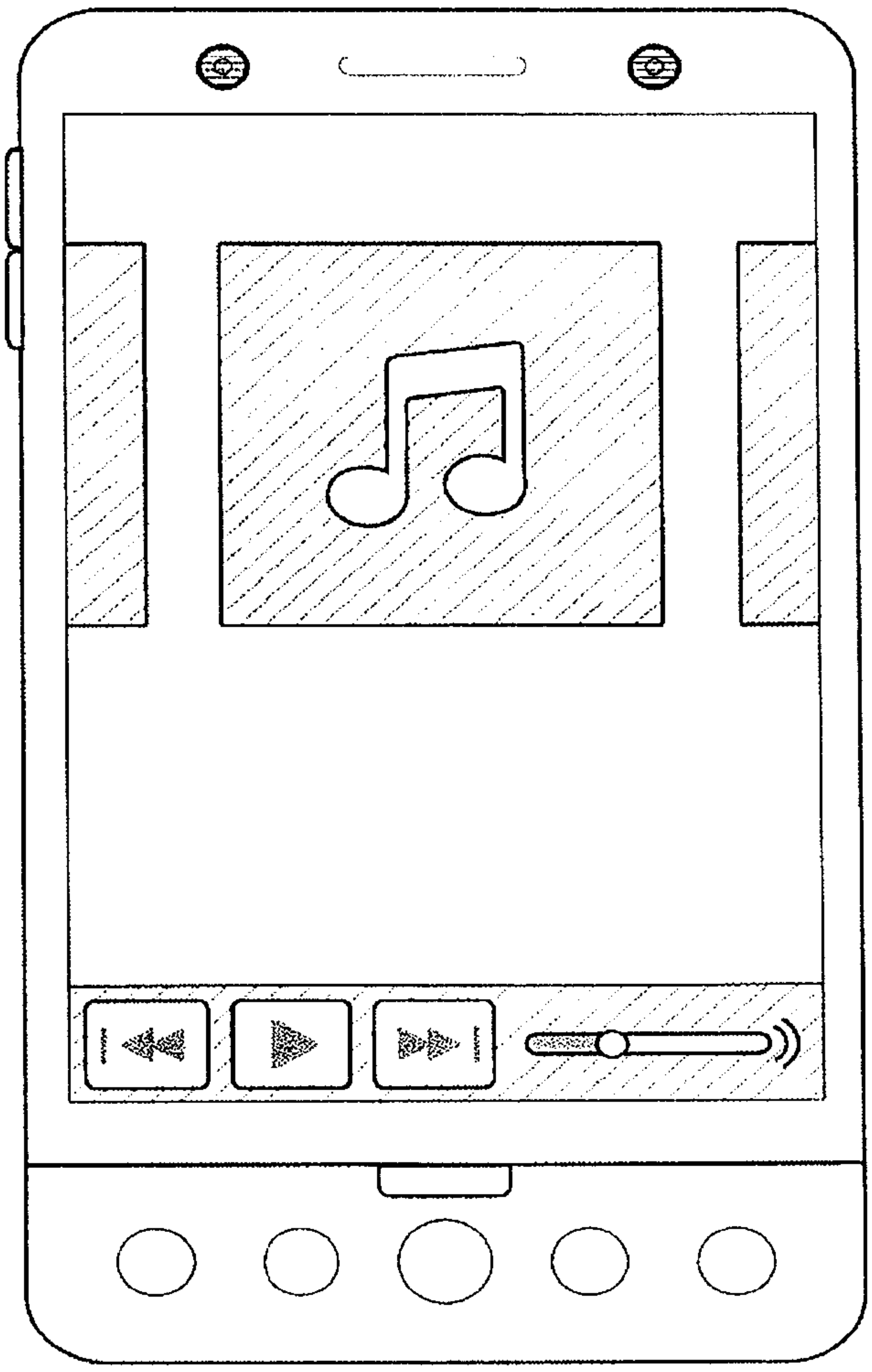


Figure 14a

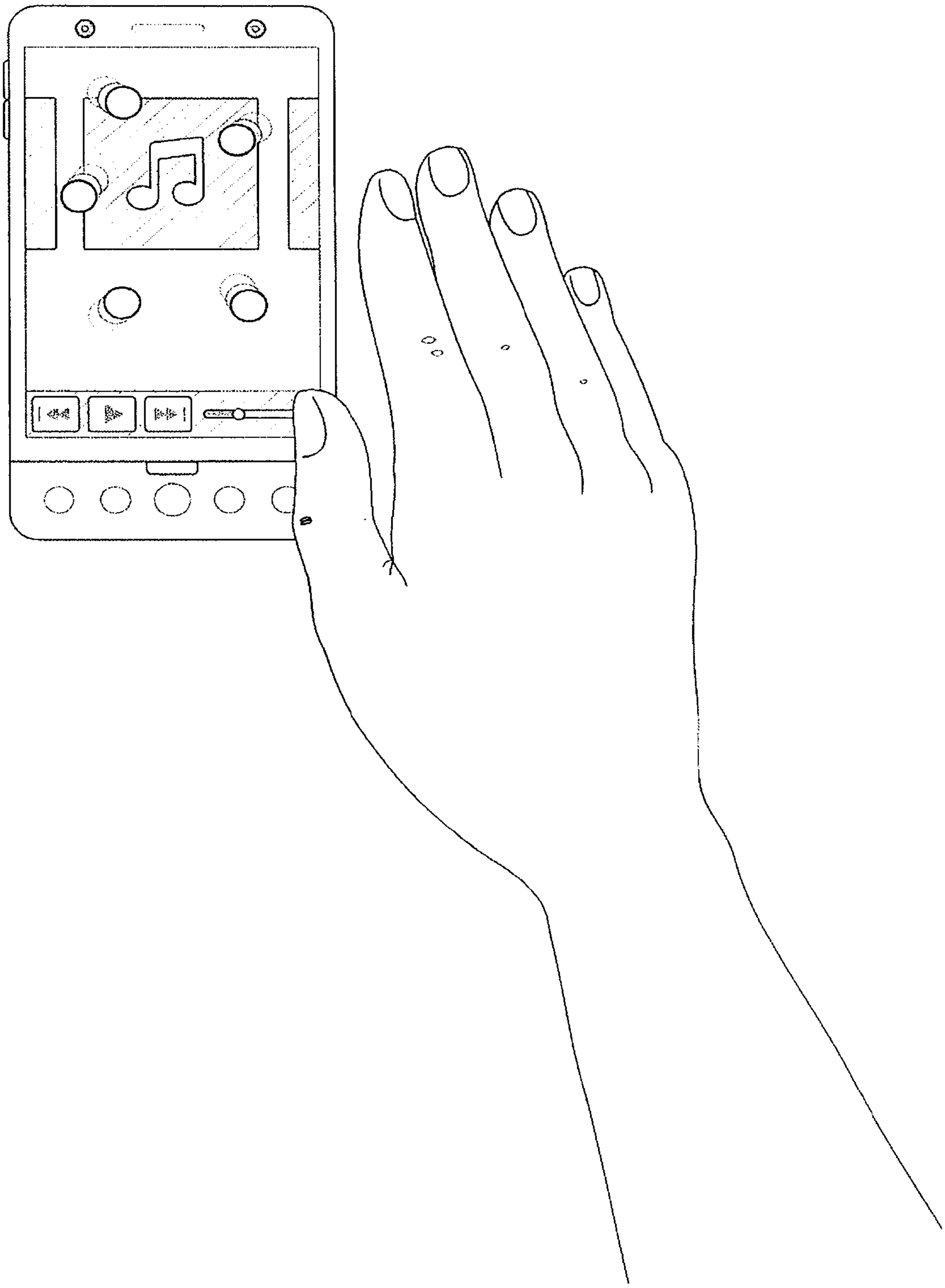


Figure 14b

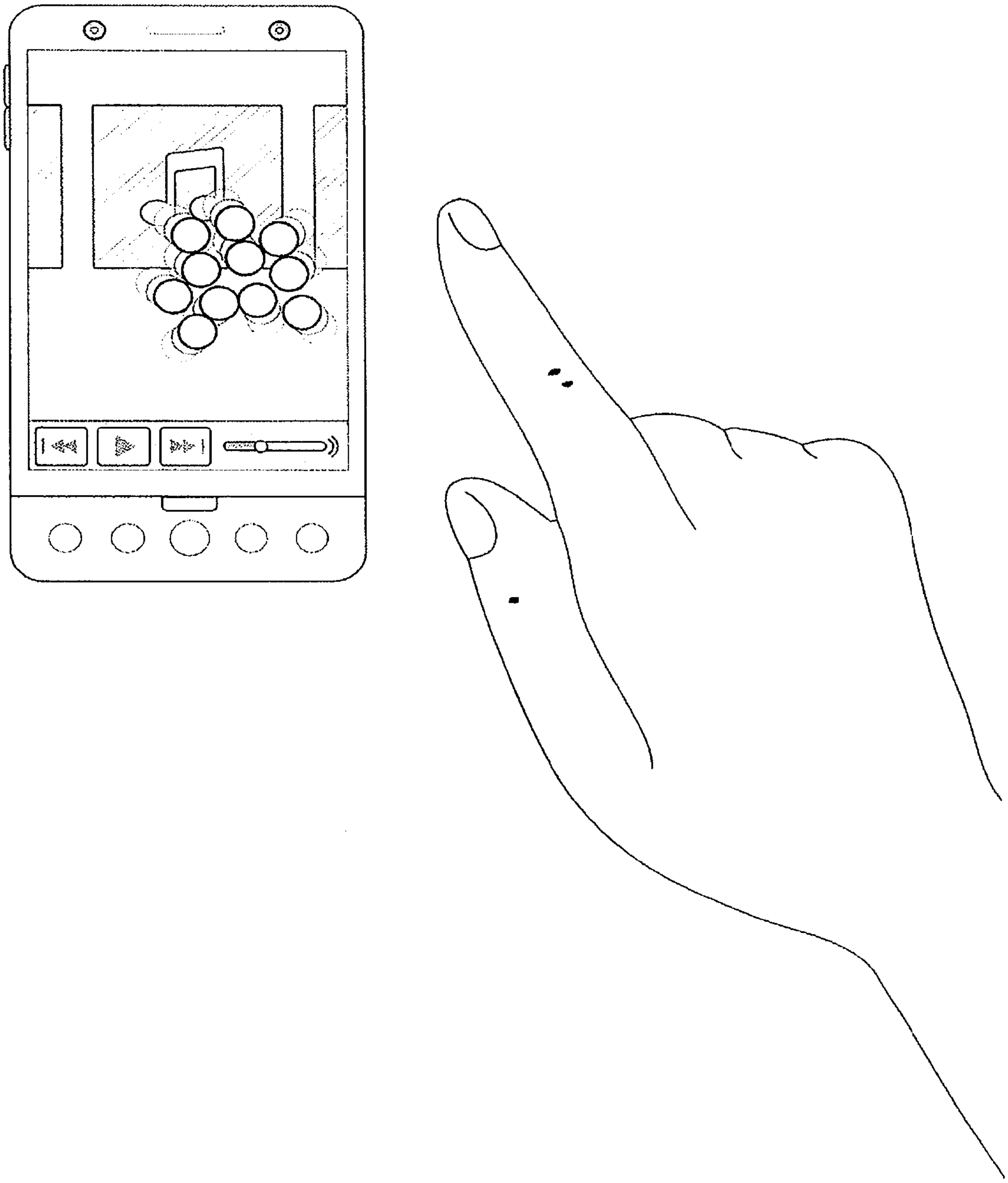


Figure 14c

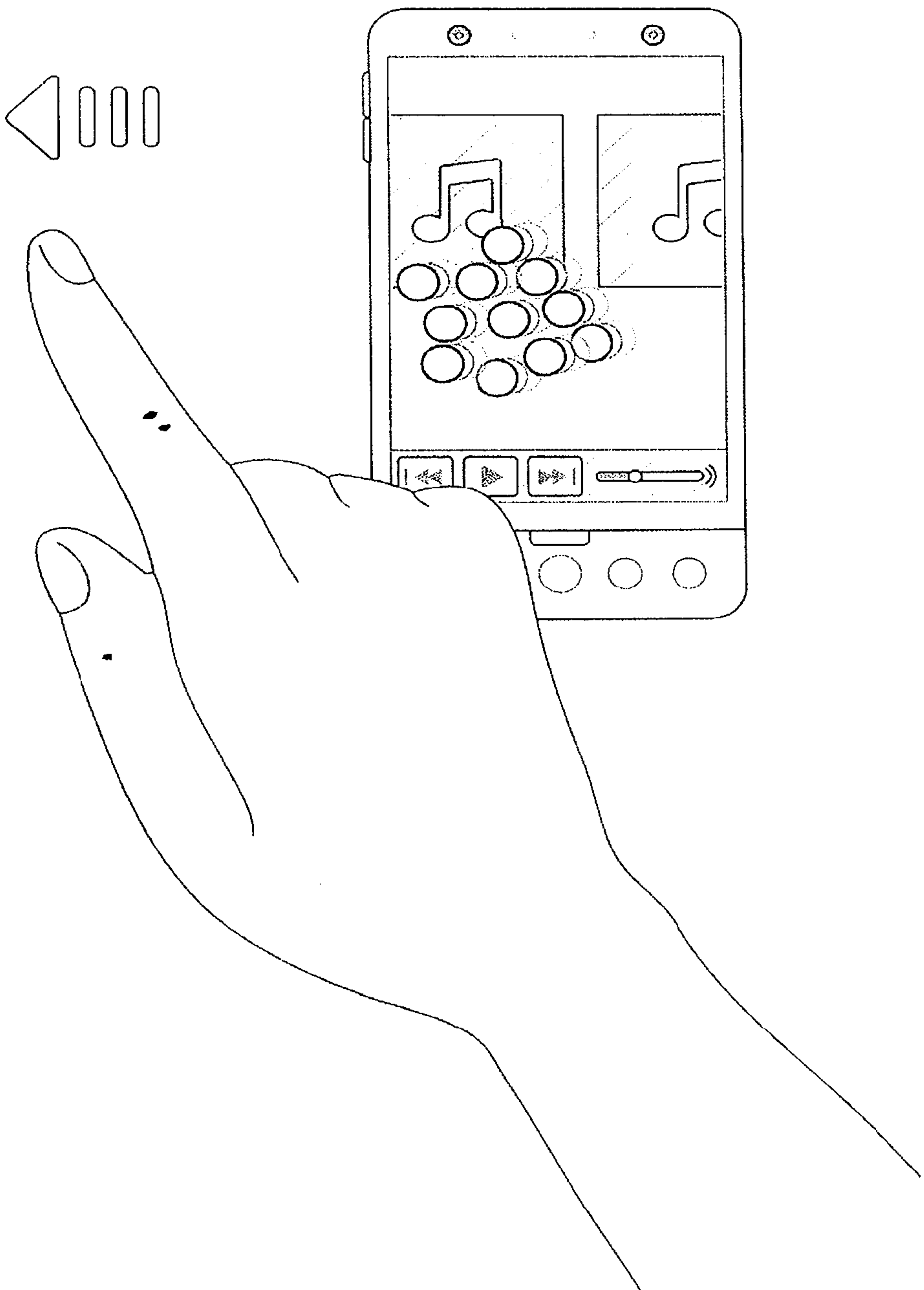


Figure 14d

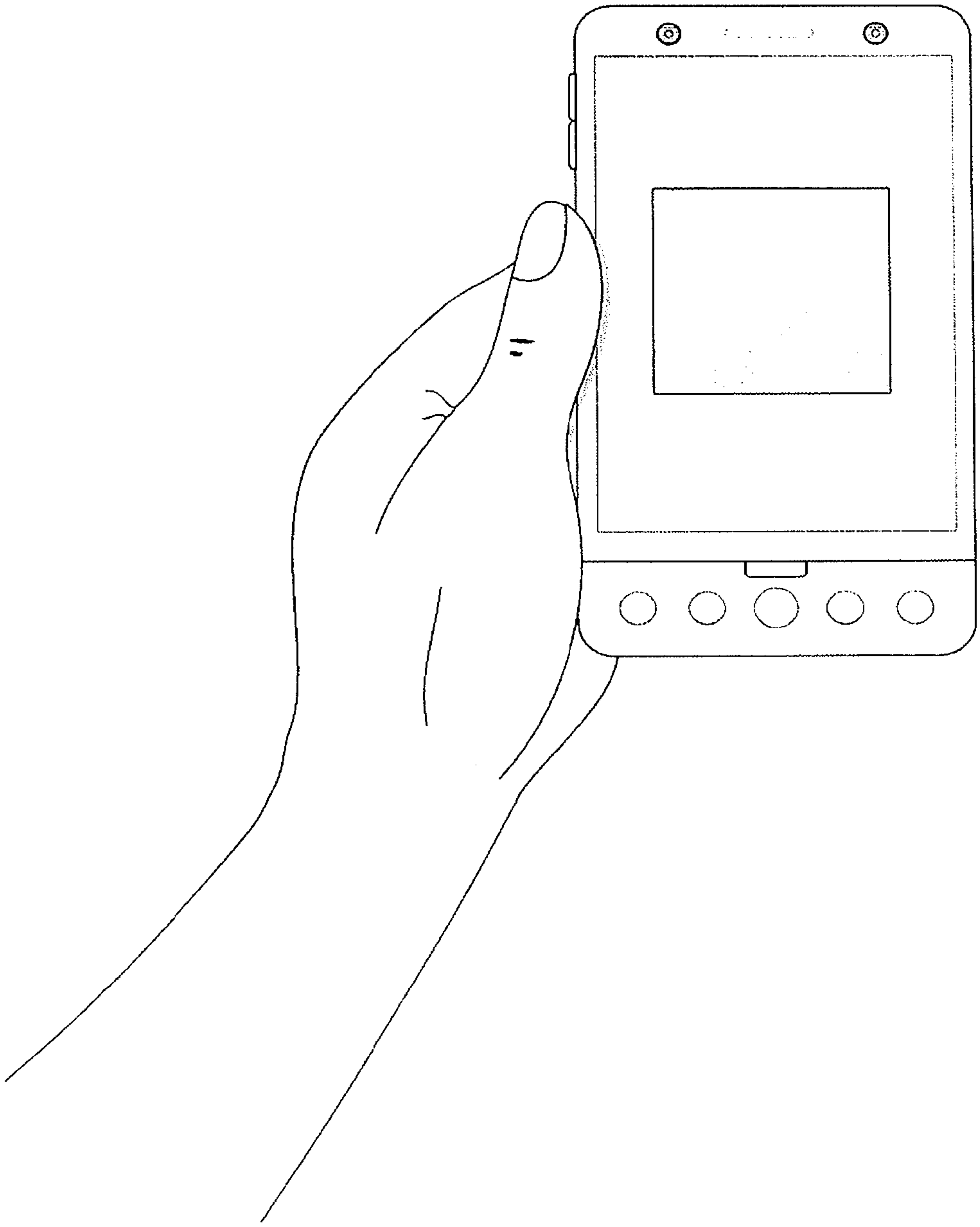


Figure 15a

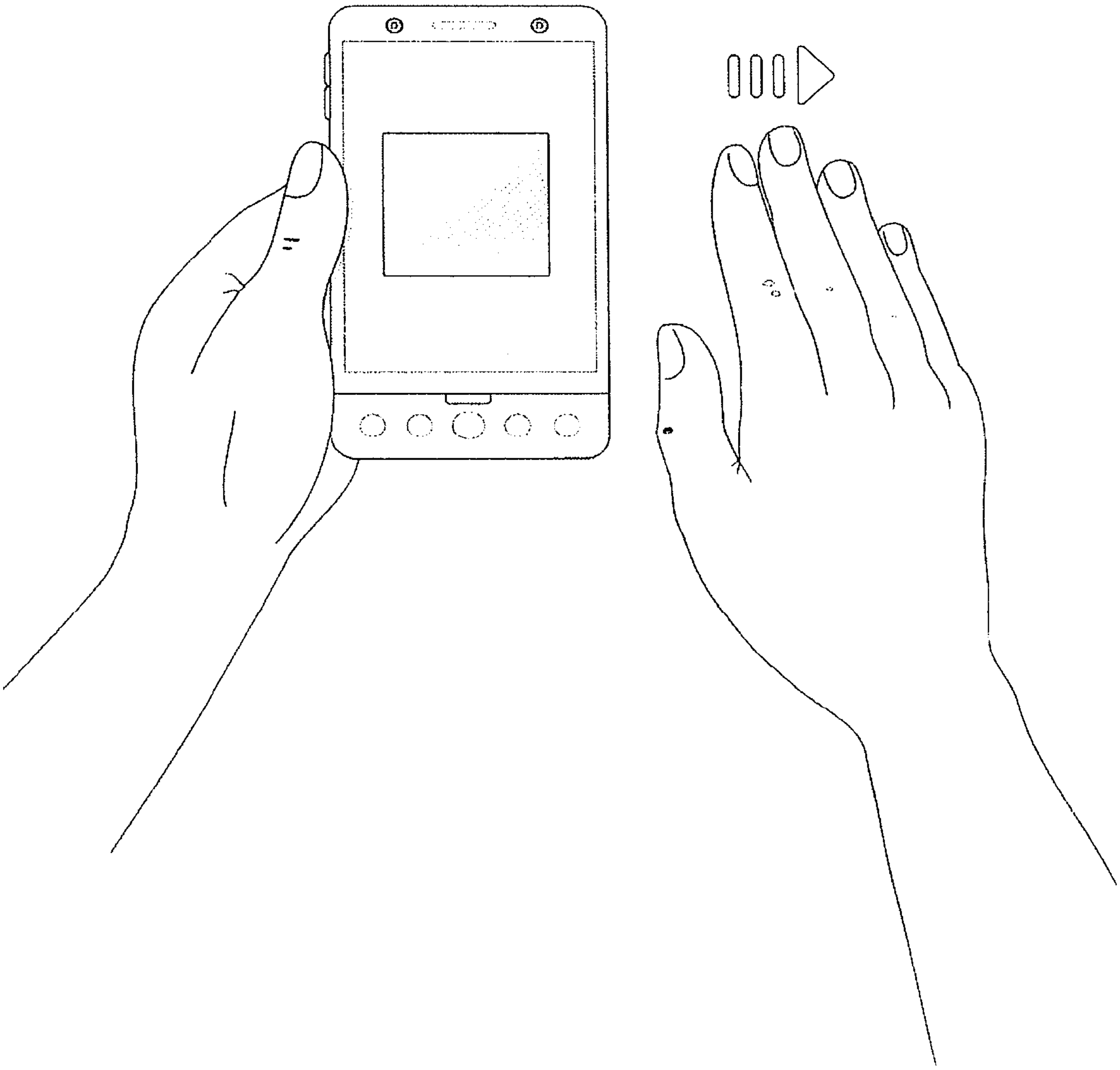


Figure 15b

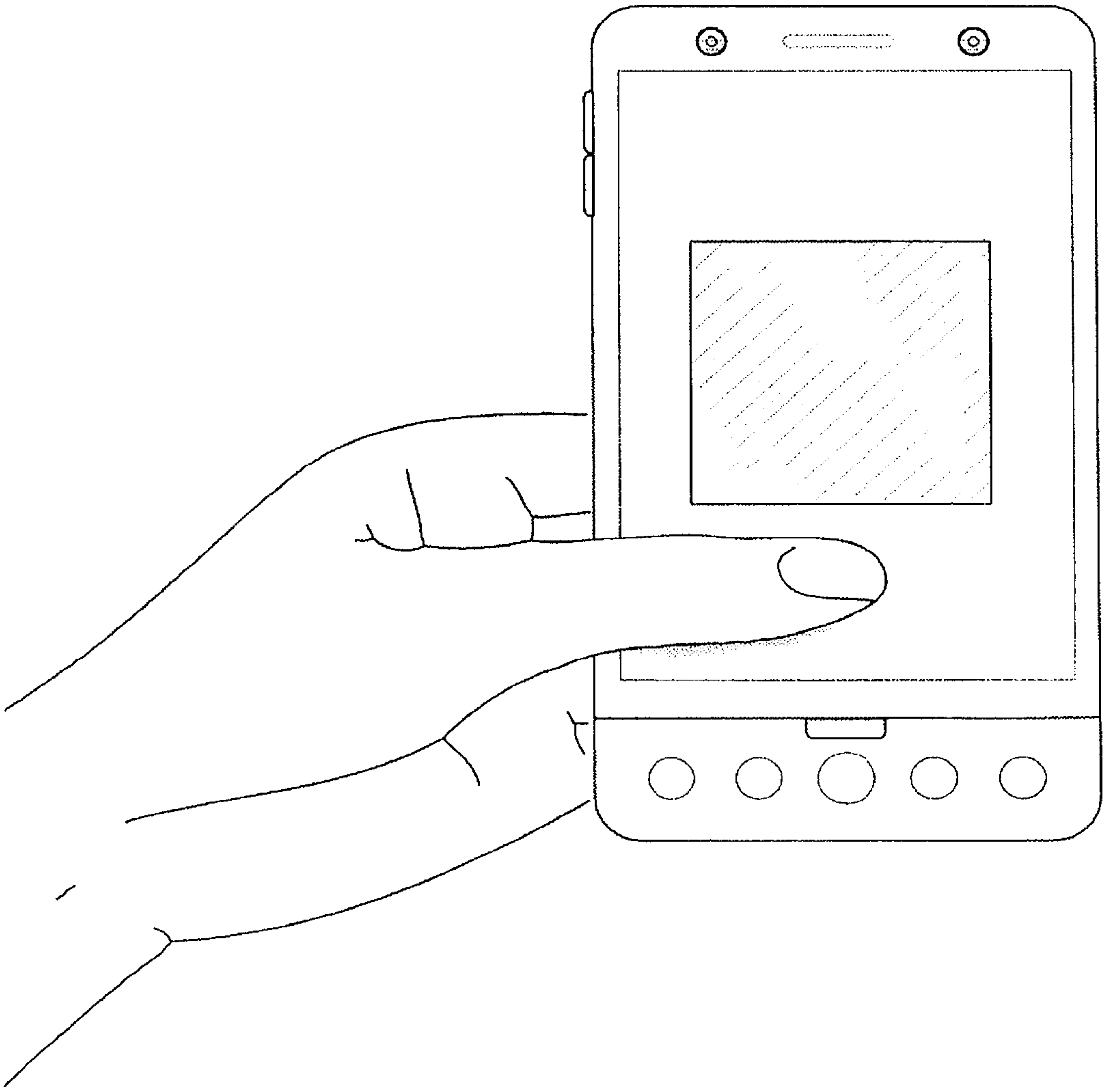


Figure 15c

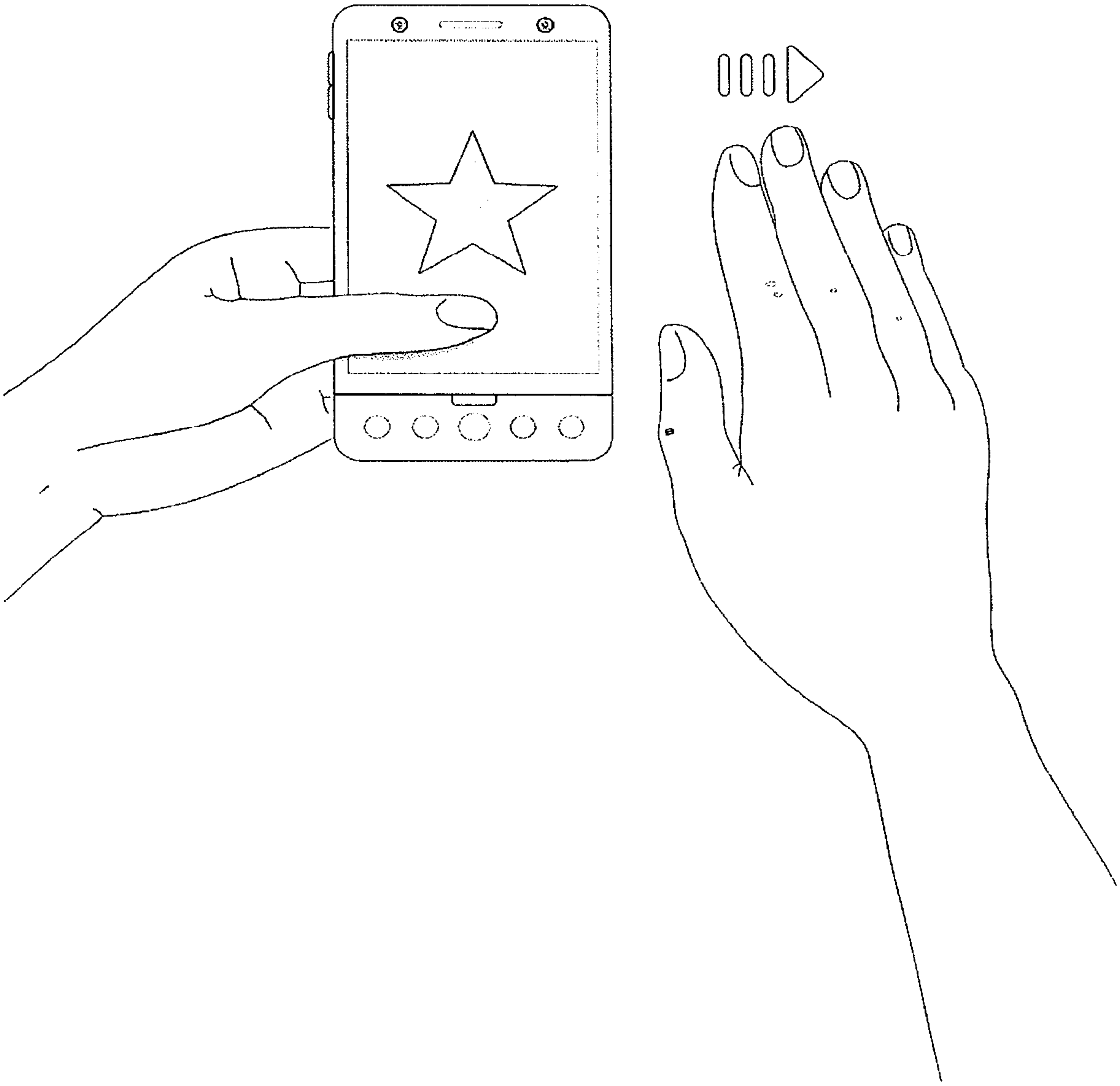


Figure 15d

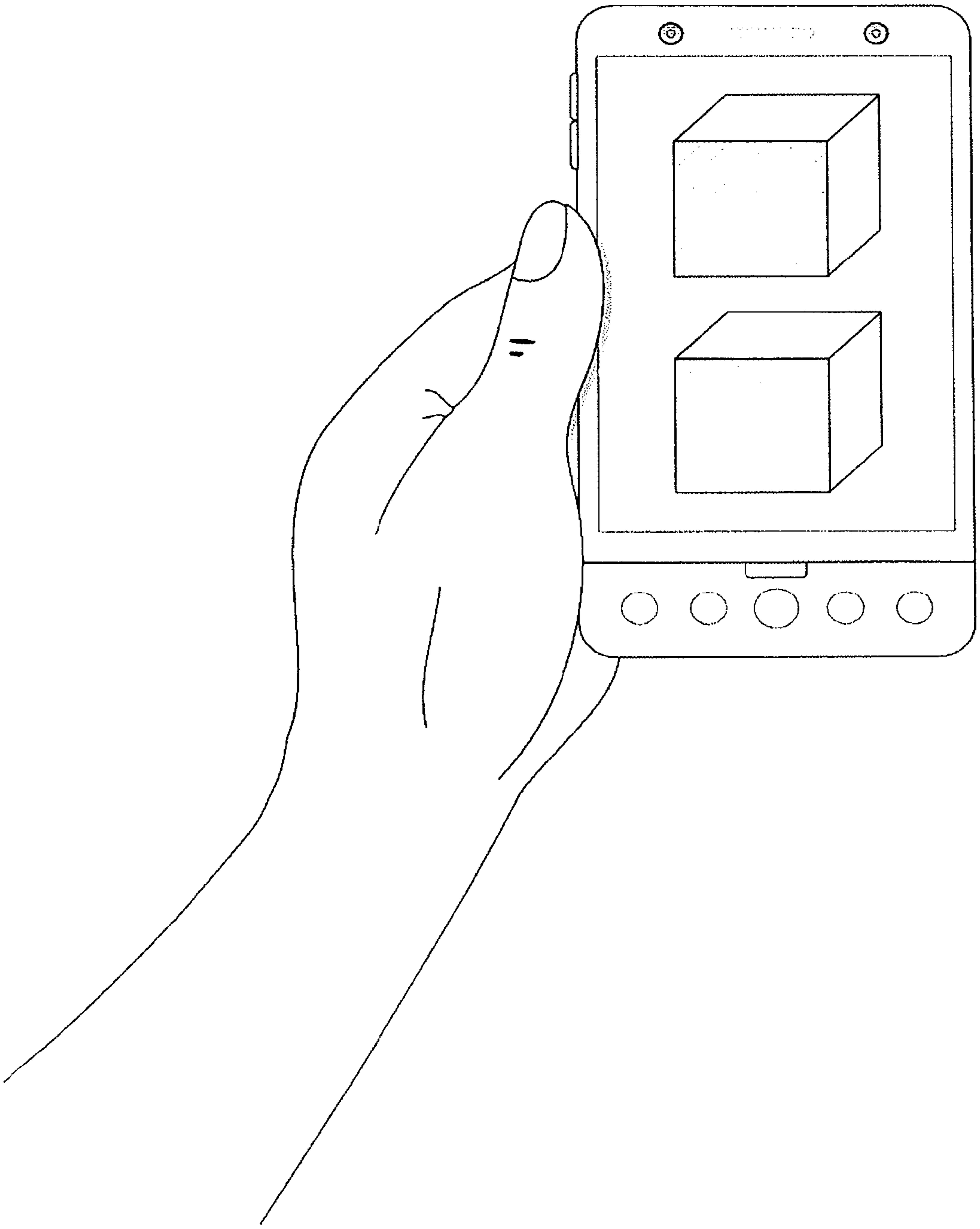


Figure 16a

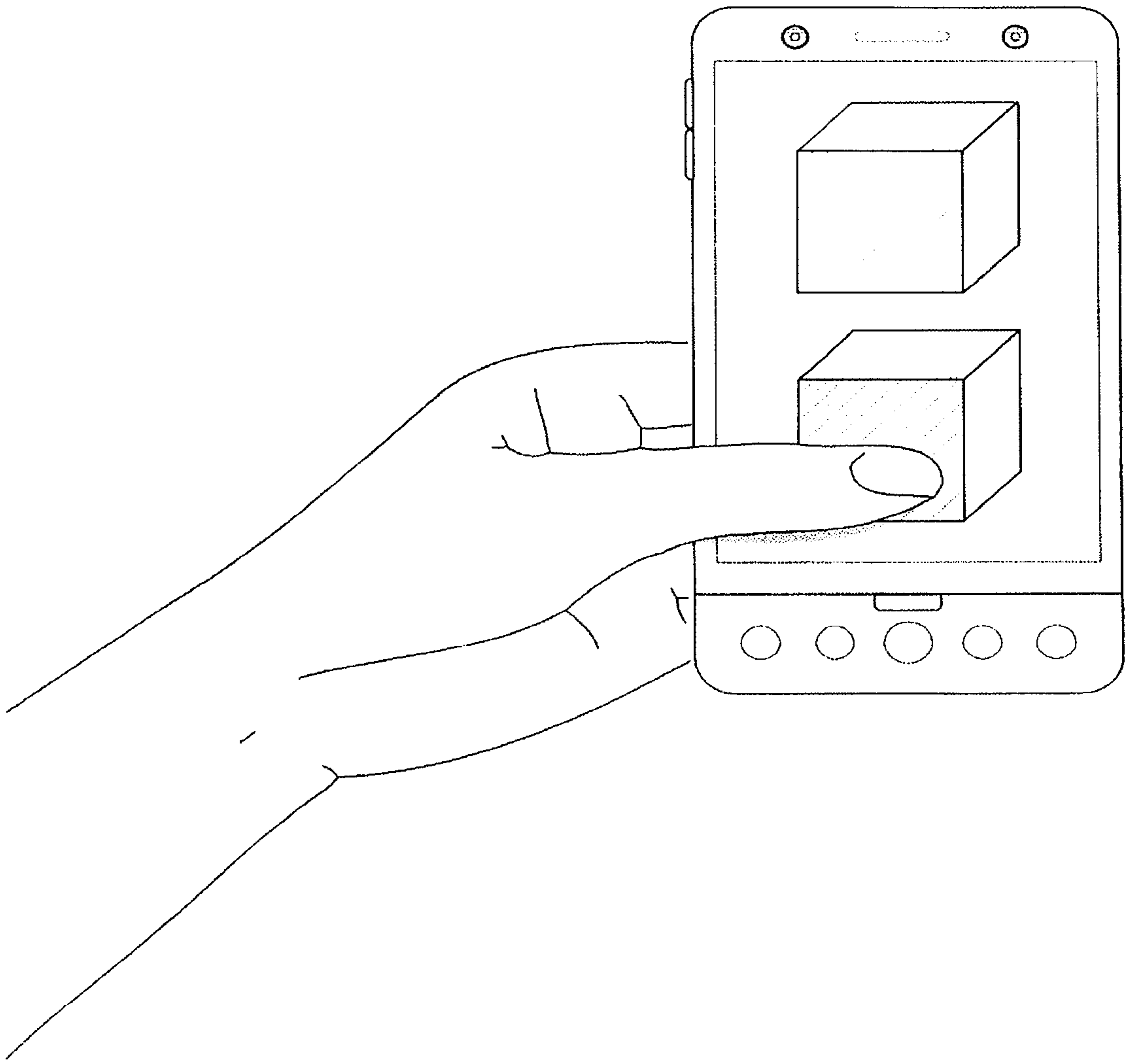


Figure 16b

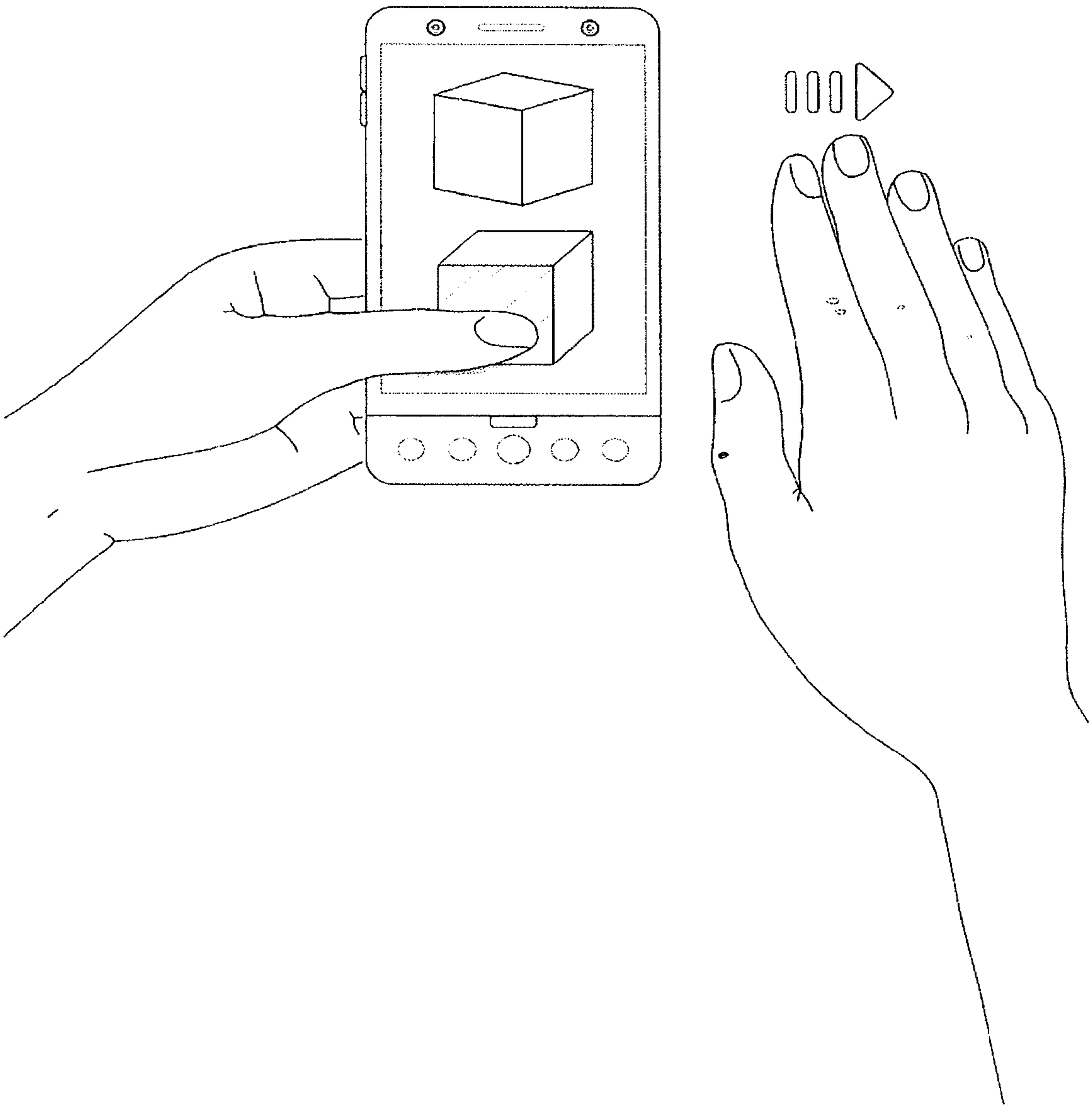


Figure 16c

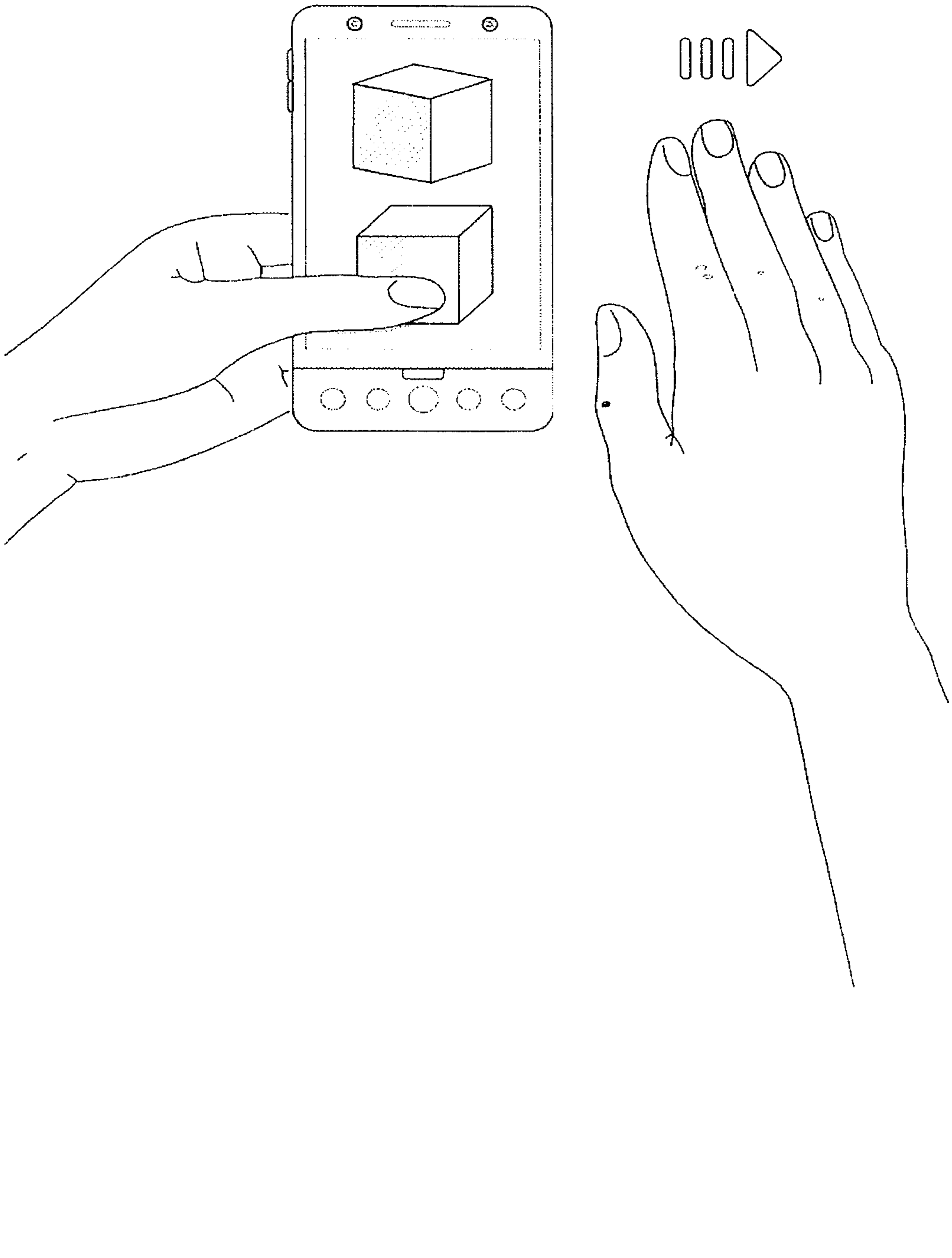


Figure 16d

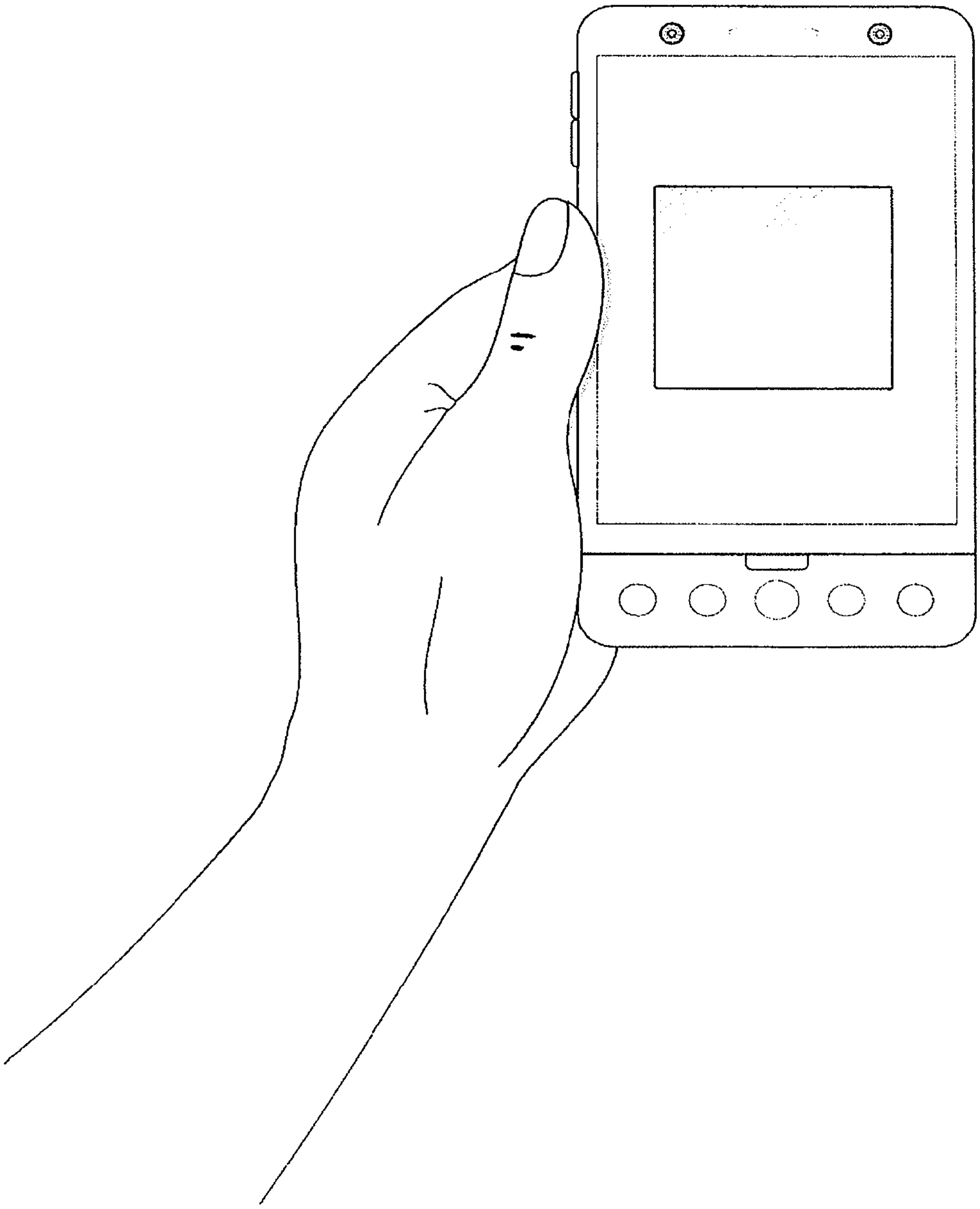


Figure 17a

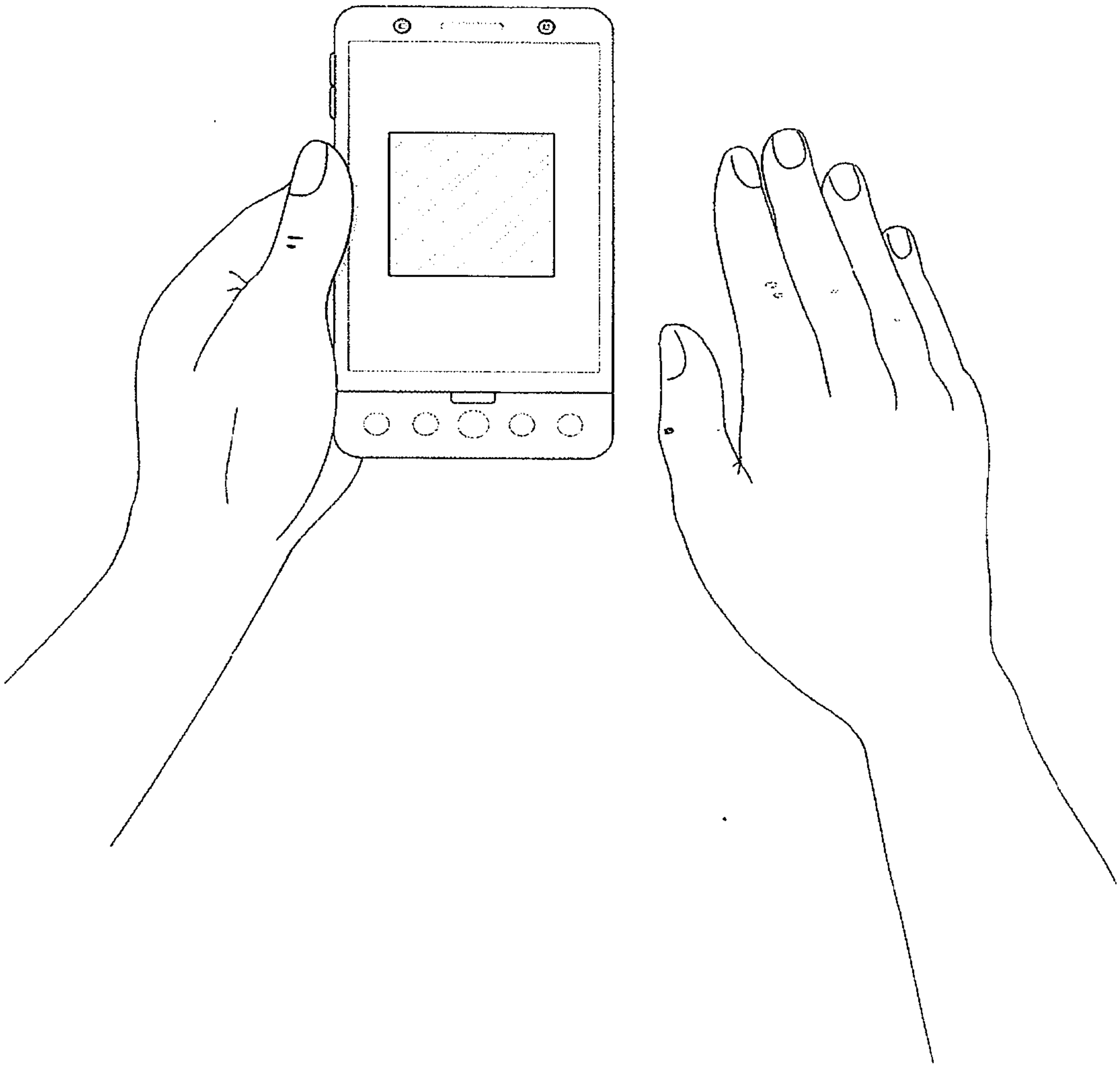


Figure 17b

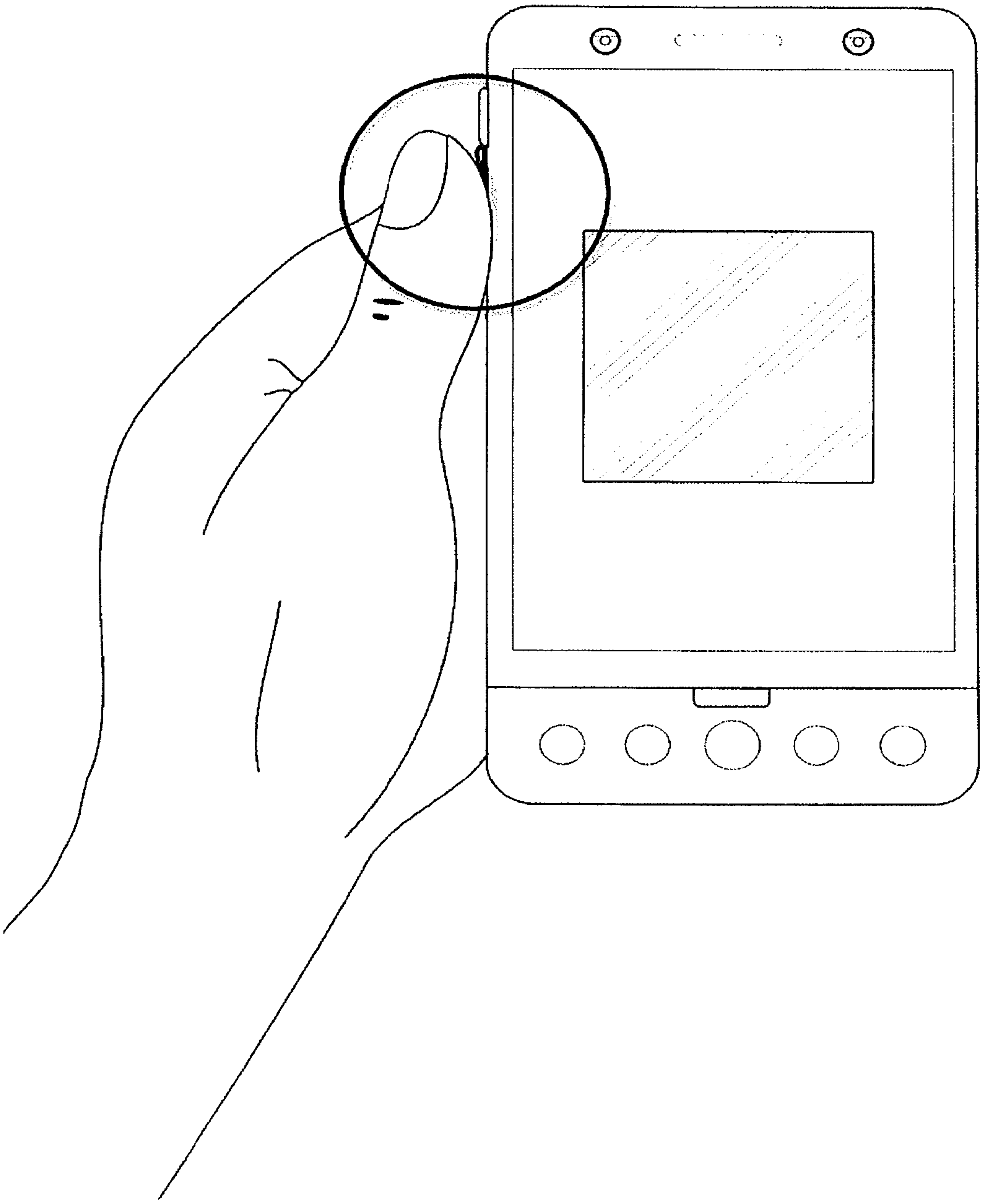


Figure 17c

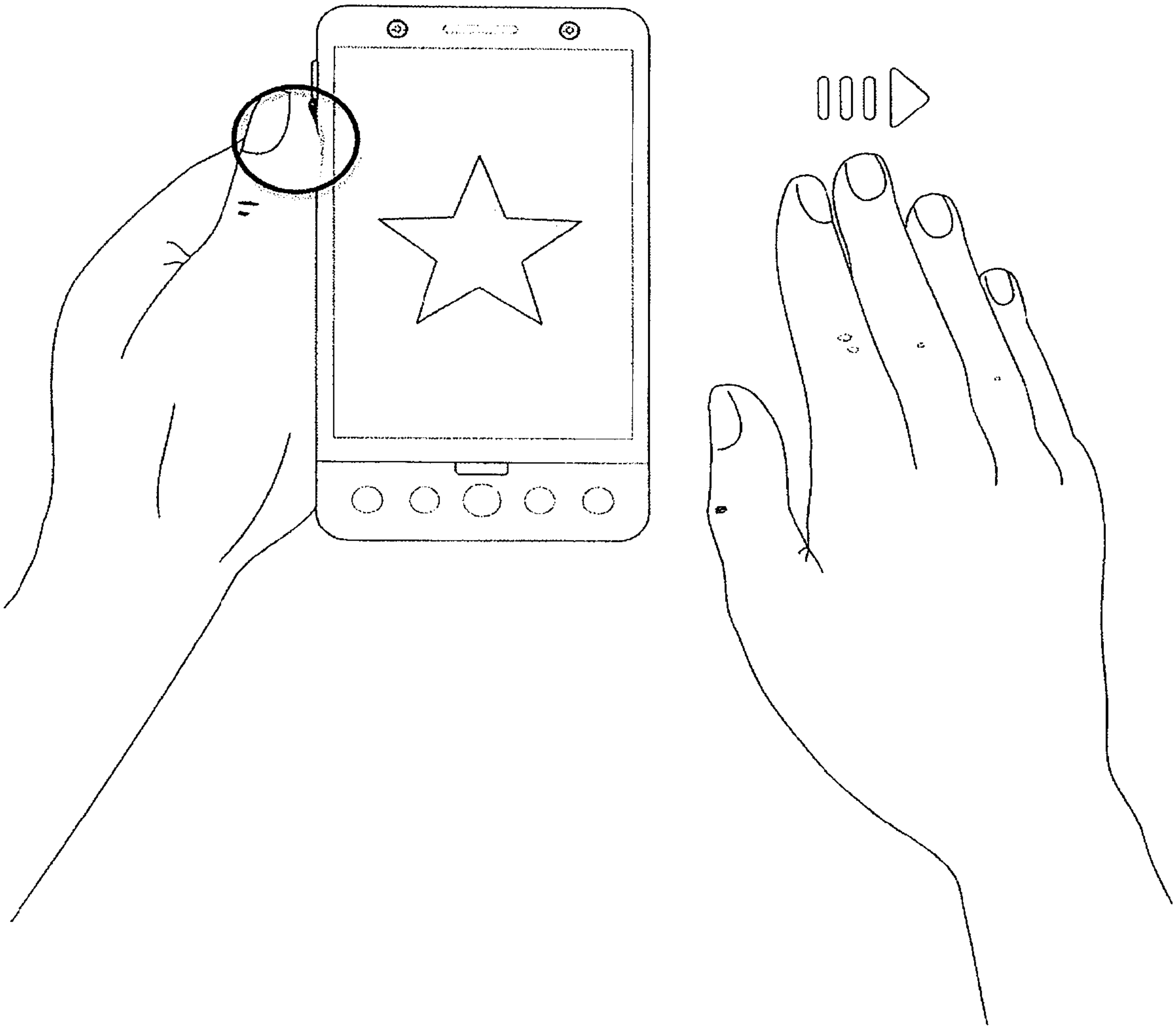


Figure 17d

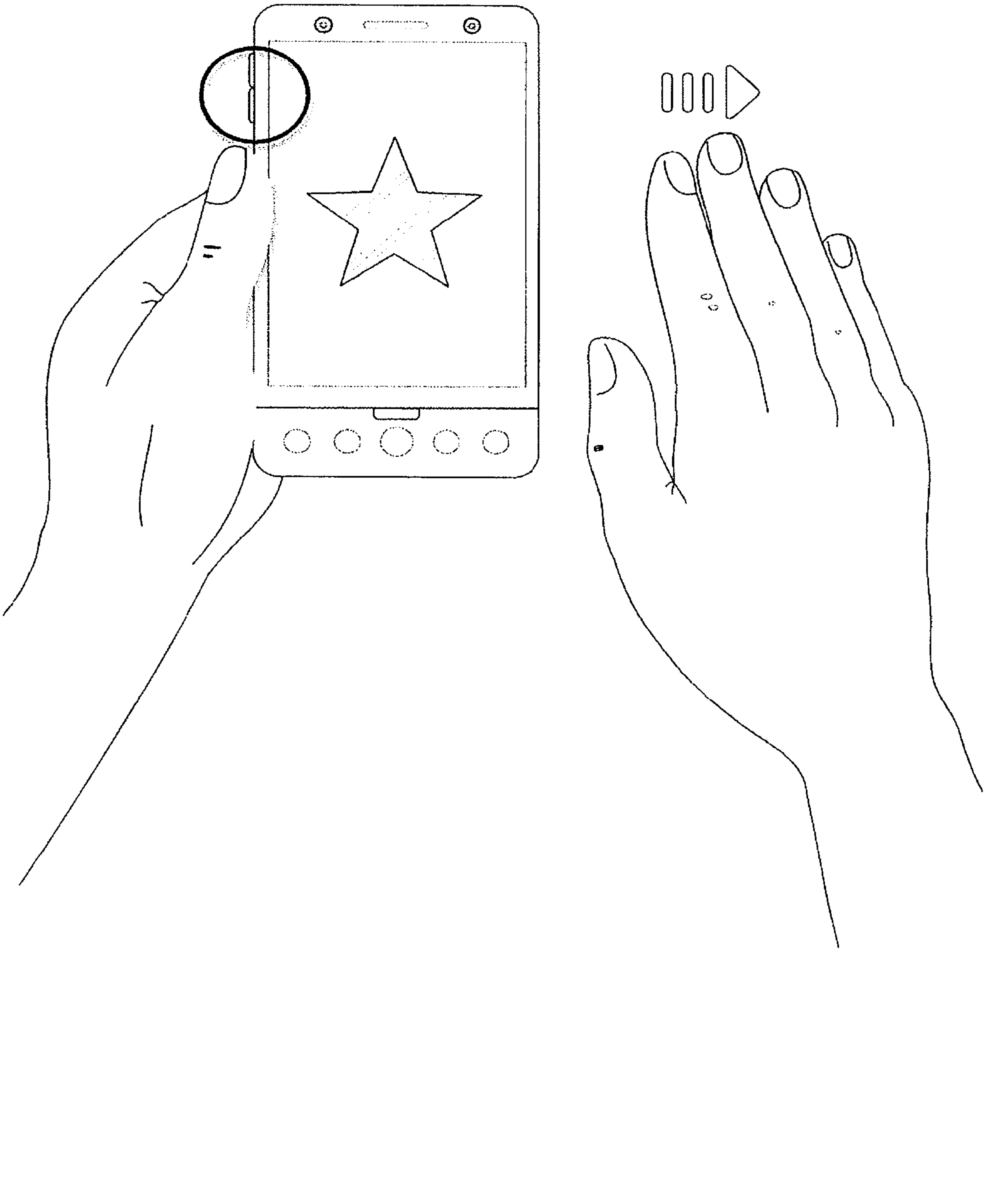


Figure 17e

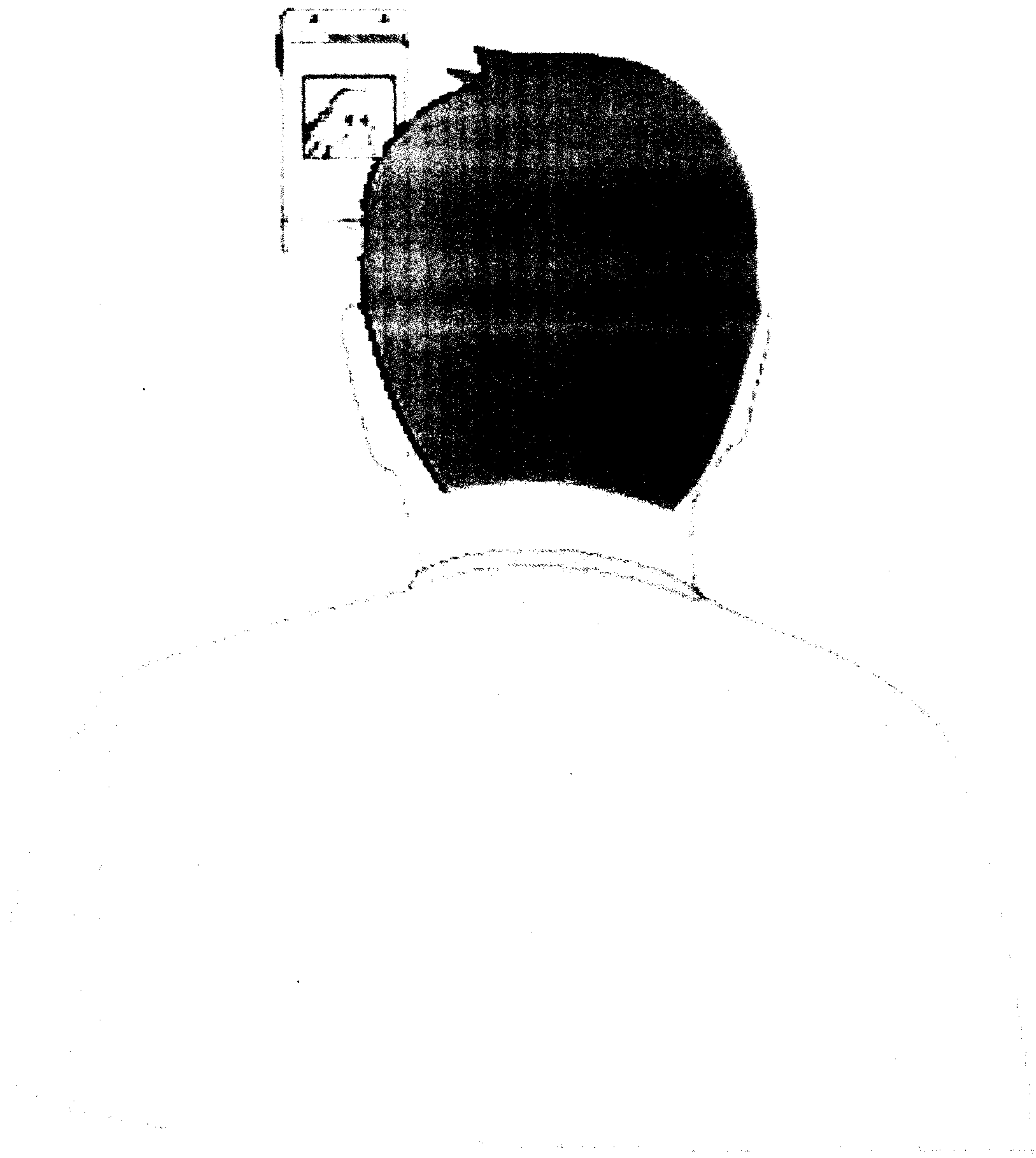


Figure 18a

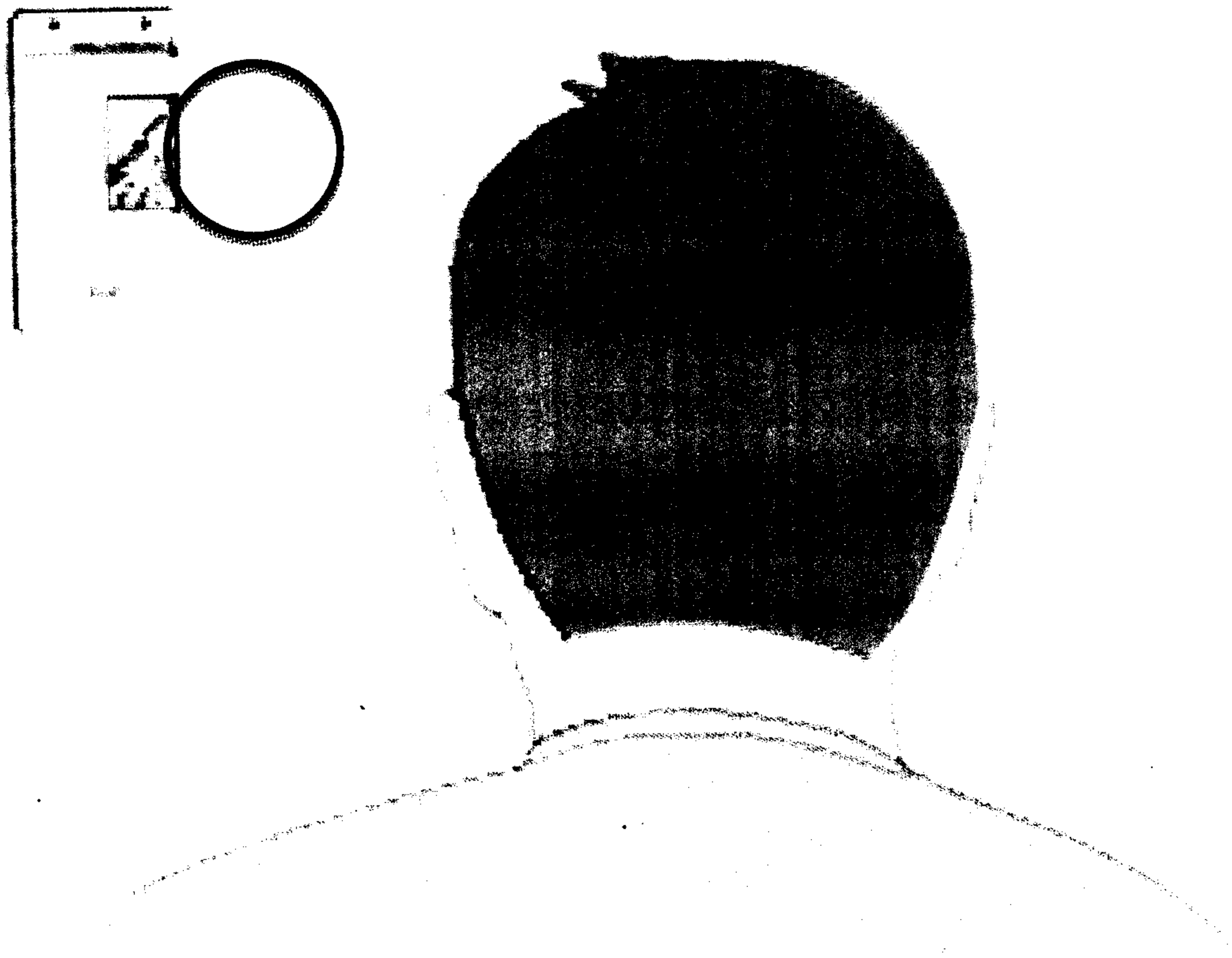
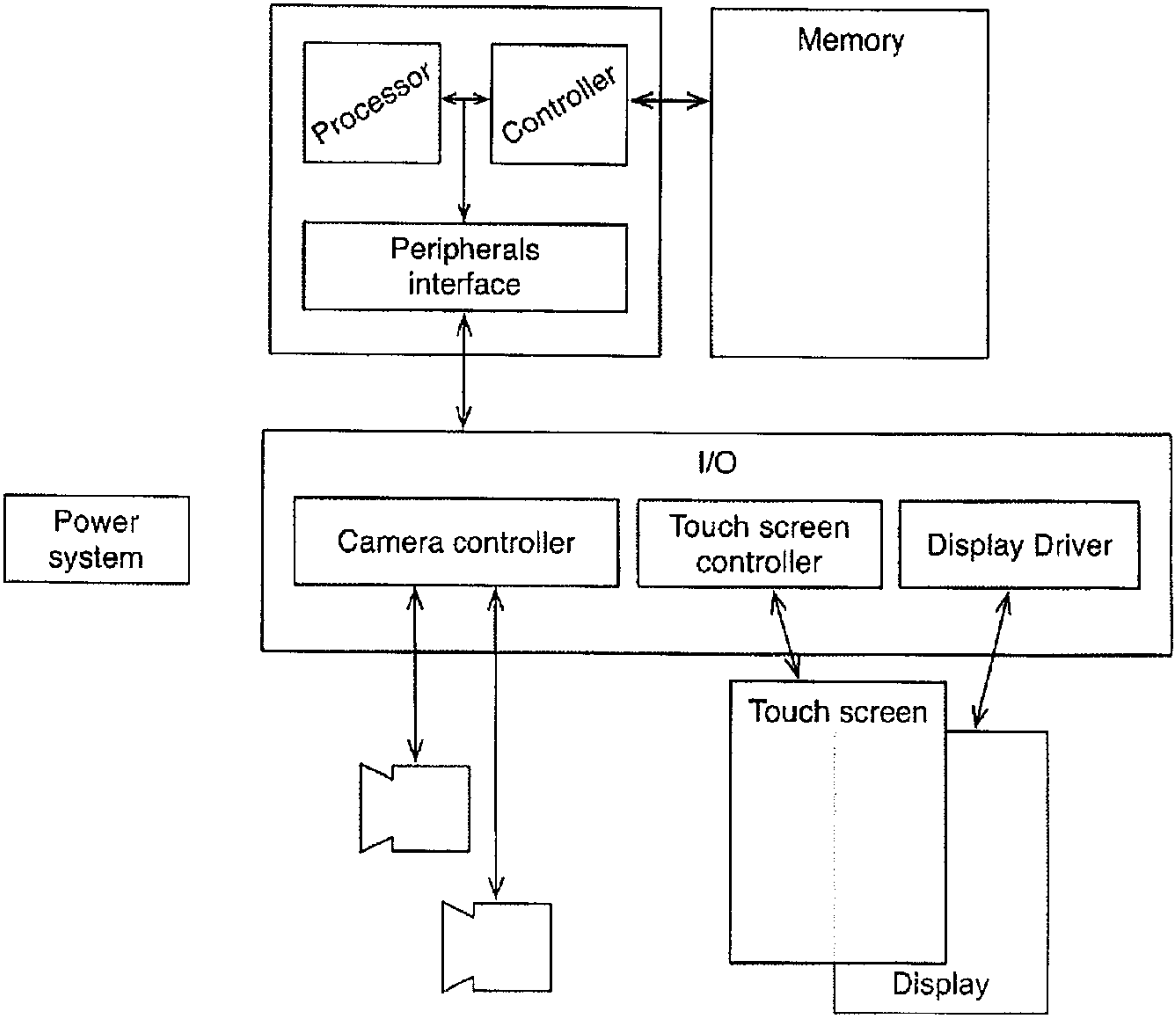


Figure 18b


Processor
Controller
Memory
Peripherals interface
I/O
Power system
Camera controller
Touch screen controller
Display Driver
Touch screen
Display