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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
14 January 2016 (14.01.2016)

WIPO | PCT

(10) International Publication Number
WO 2016/006918 A1

(51) International Patent Classification:
G06F 3/0488 (2013.01) *G06F 3/041* (2006.01)
G06F 3/01 (2006.01) *G06F 3/044* (2006.01)

(21) International Application Number:
PCT/KR2015/007020

(22) International Filing Date:
7 July 2015 (07.07.2015)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
3353/CHE/2014 7 July 2014 (07.07.2014) IN
3353/CHE/2014 21 April 2015 (21.04.2015) IN
10-2015-0095954 6 July 2015 (06.07.2015) KR

(71) Applicant: **SAMSUNG ELECTRONICS CO., LTD.** [KR/KR]; 129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do 443-742 (KR).

(72) Inventors: **DEOTALE, Gunjan P**; 37,Hemant Nagar, 1st Main, Behind Kalamandir, Marathalli, Bangalore 560037 (IN). **VAISH, Rahul**; C1, BRS Houses, 117/4B3, 5th Cross, SGR Dental College Road, Munnekolala, Marathahalli Post Bangalore, Karnataka 560037 (IN). **BHUVANAGIRI, Sanjay Dixit**; Flat 303, MVV Plaza, Lawsons Bay colony, Visakhapatnam, Andra pradesh 530017 (IN). **SULTHAR THAJUDEEN, Niyas Ahmed**; Flat No 307, N R Springwoods Apt, 10th cross mcechs layout, Dr.shivaram karanth nagar, Bangalore, 560077 (NA). **POKHREL, Niroj**; #A2, Samhita Blossoms Apartment, Nekkundi Chinppareddy Road, Oppo. EMC2, Doddannekundi, Mandevpura Post, Bangalore 560037 (IN). **DWARAKA BHAMIDIPATI, Sreevatsa**; House no : 421, 3rd Main, B Block, AECS Layout, Kundalahalli, Bangalore 560037 (IN). **KIM, Changjin**; No. 4301-304, 12, Dongbaek 2-ro, Giheung-gu, Yongin-si, Gyeonggi-do 446-715 (KR). **KWON, Jungtae**; No. 128-1304, 498, Yeongtong-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do 443-740 (KR). **KIM, Namyun**; No. 104-306, 113, Songhyeon-ro, Dalseo-gu, Daegu 704-826 (KR).

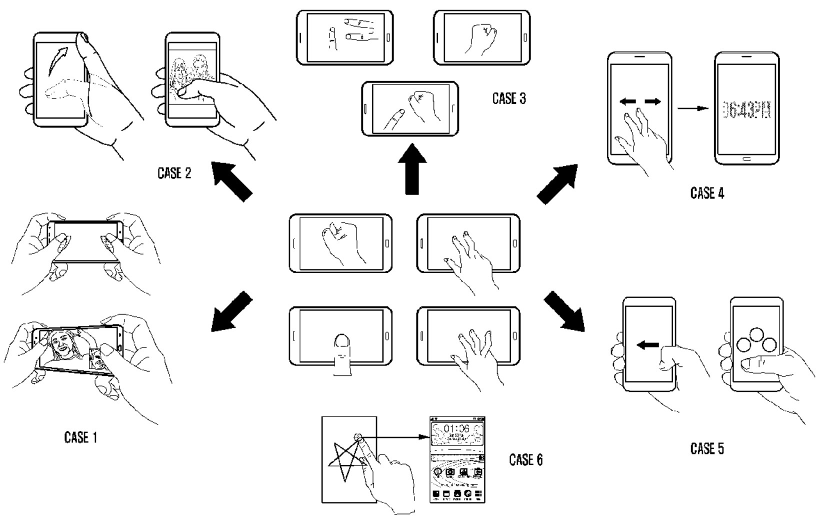
(74) Agent: **YOON, Dong Yol**; Yoon & Lee International Patent & Law Firm, 3rd Fl, Ace Highend Tower-5, 226, Gasan Digital 1-ro, Geumcheon-gu, Seoul 153-803 (KR).

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

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

[Continued on next page]

(54) Title: METHOD OF PERFORMING A TOUCH ACTION IN A TOUCH SENSITIVE DEVICE



(57) Abstract: A method of performing a touch action in a touch-sensitive device is provided. The method includes detecting a shape of a contact area associated with a touch input provided on a touch screen, determining whether the detected shape is valid based on a predetermined criteria, detecting a gesture based on a movement of a validated shape, determining whether the detected gesture is valid by matching the detected gesture with one or more predefined gestures, and performing the touch action based on a determination that the detected gesture is valid and a determination that the detected shape is valid.

WO 2016/006918 A1

Published:

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

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

Description

Title of Invention: METHOD OF PERFORMING A TOUCH ACTION IN A TOUCH SENSITIVE DEVICE

Technical Field

[1] The present disclosure relates to a touch screen. More particularly, the present disclosure relates to a method and system for identifying a gesture to enable one or more device specific actions in a touch screen enabled device.

Background Art

[2] Touch screens of different types of electronic devices enable a detection of co-ordinates of points of touch and perform a predefined action based on the detected co-ordinates. With a touch screen, a movement of an input pointer on a display can correspond to relative movements of a user's finger, as the user's finger is moved along a surface of the touch screen. Likewise, hand gestures have been implemented in touch screens, such that selections can be made when one or more taps are detected on a surface of the touch screen. In some cases, any portion of the touch screen can be tapped, and in other cases a dedicated portion of the touch screen can be tapped. In addition to making selections, scrolling can be initiated using a finger motion at an edge of the touch screen.

Disclosure of Invention

Technical Problem

[3] Recently, more advanced gestures have been implemented. For example, scrolling can be initiated by placing four fingers on the touch screen so that the scrolling gesture is recognized, and thereafter moving the four fingers on the touch screen performs scrolling events. Methods for implementing these advanced gestures, however, can be limited and in many instances counterintuitive. In certain applications, it can be beneficial to enable a user to use “real-world” gestures such as hand movements and/or finger orientations that can be generally recognized to mean certain things, so as to more efficiently and accurately perform intended operations. However, there are requirements of pre-learning or training of a model for different types of inputs in order to enable specific shape and gesture interactions. This pre-learning or training occupies storage of the device that includes the touchscreen.

[4] Therefore, there is a need for an efficient and faster method and system for identifying touch inputs on a touch screen. More specifically, there is a need for a method and a system for recognizing touch inputs received when a touch enabled screen is in a switched off state.

[5] The above information is presented as background information only to assist with an

understanding of the present disclosure. No determination has been made, and no assertion is made, as to whether any of the above might be applicable as prior art with regard to the present disclosure.

Solution to Problem

[6] Aspects of the present disclosure are to address at least the above-mentioned problems and/or disadvantages and to provide at least the advantages described below. Accordingly, an aspect of the present disclosure is to provide a method of identifying gesture to enable device specific actions in touch screen enabled devices using one or more parameters. The gesture can be a mere hand shape or a touch pattern.

[7] Another aspect of the present disclosure is to enable a device application when the screen of the touch sensitive device is in a screen-off state. Consider that a user swipes across the screen in the screen-off state to open front camera for a shot. When a user performs a touch gesture using two thumbs in the screen-off state, the camera application is automatically launched.

[8] In accordance with an aspect of the present disclosure, a method of performing a touch action in a touch sensitive device is provided. The method includes identifying a shape of a contact area associated with a touch input provided on a touch screen of the touch sensitive device based on one or more predefined parameters, determining an orientation of the identified shape, determining whether the identified shape is valid based on a predetermined criteria, and performing the touch action based on a determination that the identified shape is valid..

[9] In accordance with another aspect of the present disclosure, a method of performing a touch action in a touch-sensitive device is provided. The method includes detecting a shape of a contact area associated with a touch input provided on a touch screen of the touch-sensitive device, determining whether the determined shape is valid based on a predetermined criteria, detecting a gesture based on a movement of a validated shape having been validated by the determining of whether the detected shape is valid, determining whether the detected gesture is valid by matching the detected gesture with one or more predefined gestures, and performing the touch action based on a determination that the detected gesture is valid and a determination that the detected shape is valid.

[10] Other aspects, advantages, and salient features of the disclosure will become apparent to those skilled in the art from the following detailed description, which, taken in conjunction with the annexed drawings, discloses various embodiments of the present disclosure.

[11]

Advantageous Effects of Invention

[12] The present disclosure can identify gesture to enable device specific actions in touch screen enabled devices using one or more parameters. The gesture can be a mere hand shape or a touch pattern. The present disclosure can perform a function based on the gesture. So, the present disclosure can increase user conveniences.

Brief Description of Drawings

[13] The above and other aspects. features, and advantages of certain embodiments of the present disclosure will be more apparent from the following description taken in conjunction with the accompanying drawings, in which:

[14] FIG. 1 is a flow chart illustrating a method of recognizing a hand shape for performing a touch action in a touch sensitive device according to an embodiment of the present disclosure;

[15] FIG. 2 is a flow chart illustrating a method of processing mutual capacitance data and computing parameters according to an embodiment of the present disclosure;

[16] FIG. 3 is a flow chart illustrating a method of data binarization according to an embodiment of the present disclosure;

[17] FIG. 4A is a flow chart illustrating a method of region identification according to an embodiment of the present disclosure;

[18] FIG. 4B is a flow chart illustrating a method of modifying wrongly interpreted region values in a process of region identification according to an embodiment of the present disclosure;

[19] FIG. 5 is a schematic representation of determining an orientation of a shape using an average angle method according to an embodiment of the present disclosure;

[20] FIG. 6 is a flow chart illustrating a method of identifying various parts of a hand and separating joined fingers according to an embodiment of the present disclosure;

[21] FIG. 7A is a flow chart illustrating a method of identifying a fist according to an embodiment of the present disclosure;

[22] FIG. 7B is a schematic representation of a shape of a fist along with a few parameters such as height, width, left to right length and right to left length according to an embodiment of the present disclosure;

[23] FIG. 8A is a flow chart illustrating a method of identifying a finger according to an embodiment of the present disclosure;

[24] FIG. 8B is a schematic representation of various parameters such as height, width, left to right diagonal length and right to left diagonal length corresponding to a shape of a finger according to an embodiment of the present disclosure;

[25] FIG. 9A is a flow chart illustrating a method of separating/differentiating a finger from a palm according to an embodiment of the present disclosure;

[26] FIG. 9B is a schematic representation of various operations present in separating a

finger from a palm to an embodiment of the present disclosure;

[27] FIG. 10 is a flow chart illustrating a method of storing shapes according to an embodiment of the present;

[28] FIG. 11 is a flow chart illustrating a method of shape matching using a calculated parameter and recorded parameters according to an embodiment of the present disclosure;

[29] FIG. 12A is a schematic representation of launching a camera using a single finger shape identification according to an embodiment of the present disclosure;

[30] FIG. 12B is a flow chart illustrating a method of launching a camera using a single finger shape identification according to an embodiment of the present disclosure;

[31] FIG. 13 is a schematic representation of launching a camera using double finger shape identification according to an embodiment of the present disclosure;

[32] FIG. 14 is a flow chart illustrating a method of performing a touch action in a touch-sensitive device according to an embodiment of the present disclosure;

[33] FIG. 15 is a flow chart illustrating a method of computing parameters and linearizing a gesture performed by a user according to an embodiment of the present disclosure;

[34] FIGS. 16A and 16B are schematic representations of computing directions according to various embodiments of the present disclosure;

[35] FIG. 17A is a flow chart illustrating a method of computing one or more parameters for matching according to an embodiment of the present disclosure;

[36] FIG. 17B is a flow chart illustrating a method of processing one or more coordinates collected in a touch sensor panel integrated circuit (TSP IC) according to an embodiment of the present disclosure;

[37] FIG. 17C is a flow chart illustrating a method of matching a gesture performed by a user with a registered gesture according to an embodiment of the present disclosure;

[38] FIG. 18A is a flow chart illustrating a method of computing a pressure vertex and a direction vertex for determining a gesture performed by a user according to an embodiment of the present disclosure;

[39] FIG. 18B is a flow chart illustrating a method of matching a gesture performed by a user with a registered gesture according to an embodiment of the present disclosure;

[40] FIG. 19 is a flow chart illustrating a method of registering one or more gestures with a touch sensitive device according to an embodiment of the present disclosure;

[41] FIG. 20 is a schematic representation of launching a camera using a single finger gesture identification according to an embodiment of the present disclosure;

[42] FIG. 21A is a schematic representation of launching a camera using double finger gesture identification according to an embodiment of the present disclosure;

[43] FIG. 21B is a flow chart illustrating a method of launching a camera using double finger gesture identification according to an embodiment of the present disclosure; and

[44] FIG. 22 is a schematic representation of launching various touch actions based on shape identification according to an embodiment of the present disclosure.

[45] The same reference numerals are used to represent the same elements throughout the drawings.

Mode for the Invention

[46] The following description with reference to the accompanying drawings is provided to assist in a comprehensive understanding of various embodiments of the present disclosure as defined by the claims and their equivalents. It includes various specific details to assist in that understanding but these are to be regarded as merely exemplary. Accordingly, those of ordinary skill in the art will recognize that various changes and modifications of the various embodiments described herein can be made without departing from the scope and spirit of the present disclosure. In addition, descriptions of well-known functions and constructions may be omitted for clarity and conciseness.

[47] The terms and words used in the following description and claims are not limited to the bibliographical meanings, but, are merely used by the inventor to enable a clear and consistent understanding of the present disclosure. Accordingly, it should be apparent to those skilled in the art that the following description of various embodiments of the present disclosure is provided for illustration purpose only and not for the purpose of limiting the present disclosure as defined by the appended claims and their equivalents.

[48] It is to be understood that the singular forms “a”, “an” and “the” include the plural referents unless the context clearly dictates otherwise. Thus, for example, reference to “a component surface” includes reference to one or more of such surfaces.

[49] It will be further understood that the terms “includes,” “comprises,” “including” and/or “comprising,” when used in this specification, specify the presence of stated features, integers, operations, elements and/or components, but do not preclude the presence or addition of one or more other features, integers, operations, elements, components, and/or groups thereof. It will be understood that when an element is referred to as being “connected” or “coupled” to another element, it can be directly connected or coupled to the other element or intervening elements may be present. Furthermore, “connected” or “coupled” as used herein may include operatively connected or coupled. As used herein, the term “and/or” includes any and all combinations and arrangements of one or more of the associated listed items.

[50] Unless otherwise defined, all terms (including technical and scientific terms) used herein have the same meaning as commonly understood by one of ordinary skill in the art to which this disclosure pertains. It will be further understood that terms, such as those defined in commonly used dictionaries, should be interpreted as having a meaning that is consistent with their meaning in the context of the relevant art and will

not be interpreted in an idealized or overly formal sense unless expressly so defined herein.

[51] FIGS. 1 through 22, discussed below, and the various embodiments used to describe the principles of the present disclosure in this patent document are by way of illustration only and should not be construed in any way that would limit the scope of the disclosure. Those skilled in the art will understand that the principles of the present disclosure may be implemented in any suitably arranged communications system. The terms used to describe various embodiments are exemplary. It should be understood that these are provided to merely aid the understanding of the description, and that their use and definitions in no way limit the scope of the present disclosure. Terms first, second, and the like are used to differentiate between objects having the same terminology and are in no way intended to represent a chronological order, unless where explicitly stated otherwise. A set is defined as a non-empty set including at least one element

[52] FIG. 1 is a flow chart illustrating a method of recognizing a hand shape for performing a touch action in a touch sensitive device according to an embodiment of the present disclosure.

[53] Referring to FIG. 1, when a user touches a screen at operation 101, mutual capacitance changes and the change in the mutual capacitance is detected by a touch sensor panel integrated circuit (TSP IC) (i.e. a touch controller) at operation 102. A shape of a touch area is considered as a region.

[54] The detected mutual capacitance data for the region is processed at operation 103 and set of parameters are calculated.

[55] Based on the calculated parameters, the shape is identified at operation 104.

[56] If the identified shape matches with predefined or prerecorded shapes in the touch sensitive device at operation 105, then the shape pattern can have combination of multiple shapes, and at operation 106, a touch action corresponding to the identified shape is enabled in the touch sensitive device. In an embodiment of the present disclosure, the touch action is enabled even when the display panel of the touch sensitive device is in a screen-off state. If the identified shape does not match the predefined or prerecorded shapes in operation 105, then the method returns to operation 101.

[57] The set of parameters according to an embodiment of the present disclosure comprises one or more of an area of a region, a width of a region, a height of a region, a right-left slant length of a region, a left-right slant length of a region, a number of touch nodes enclosed in a region, a hypotenuse of a region, a rectangularity of a region, an elongatedness of a region and an average angle of a region.

[58] In an embodiment of the present disclosure, the region is identified based on

detecting a change in the mutual capacitance in the touch screen. The mutual capacitance data is processed and the one or more parameters are defined based on the identified one or more touch regions. The method of processing the mutual capacitance data is explained in detail in FIG. 2. Once the parameters are identified, the value of each of the defined parameters are determined for the identified one or more touch regions. The shape of the region is identified based on the determined values of the defined parameters.

[59] In an embodiment of the present disclosure, the touch action is enabled even when the display panel of the touch sensitive device is in the screen-off state. For instance, consider that a user swipes across the screen when the screen is in the screen-off state, to open front camera for a selfie shot. When a user performs a touch gesture using two thumbs in the screen-off state, a camera application is automatically launched. In the screen-off state of the touch screen device, the user positions their fingers to make different shapes to trigger a corresponding operation. The touch screen device recognizes the touch gestures in the screen-off state by identifying hand/finger shapes using a non-model based approach. Likewise, when a user performs a pinch out gesture in the screen-off state, a pinched-out portion of the screen displays the time and missed call notifications. When a user performs a small swipe gesture using one finger on any portion of the screen in screen-off state without lifting their finger, shortcuts for different applications will be displayed. In the screen-off state, the TSP IC may be operated in a low power mode. In an example, the low power mode may be achieved by decreasing a frequency of operation of the TSP IC.

[60] FIG. 2 is a flow chart illustrating a method of processing mutual capacitance data and computing of parameters according to an embodiment of the present disclosure.

[61] Referring to FIG. 2, in order to identify a shape of a contact area, mutual capacitance data is processed. The processing includes filtering of mutual capacitance data using a binarization method as indicated in operation 201. The method of data binarization is explained in detail in FIG. 3.

[62] Further, one or more touch regions on the touch screen are identified using a region identification method at operation 202.

[63] Additionally, the parameters are computed for each identified region in operation 203.

[64] FIG. 3 is a flow chart illustrating a method of data binarization according to an embodiment of the present disclosure.

[65] Referring to FIG. 3, according to an embodiment of the present disclosure, a coupling capacity of two-crossing conductor/electrodes of a path of touch sensor panel is considered as node. A change in mutual capacitance at each of the nodes is calculated. A predefined threshold value is selected for performing data binarization.

[66] Further, as indicated in operation 301, the change in mutual capacitance at the current node is compared with the predefined threshold value of the mutual capacitance. If the mutual capacitance value of the current node is greater than the predefined threshold, the mutual capacitance value of the current node is set as 1 at operation 303. If the mutual capacitance value of the current node is less than the predefined threshold, the mutual capacitance value of the current node is set as 0 at operation 302.

[67] FIG. 4A is a flow chart illustrating a method of region identification according to an embodiment of the present disclosure.

[68] Referring to FIG. 4A, in the present disclosure, an identification of a touch region is performed using a mutual capacitance touch data array. The mutual capacitance touch data array is formed using binarized data.

[69] In order to identify the region of touch, the following initial conditions are given at operation 401.

[70] Initialize region value = 2

[71] Connected region value = 0

[72] Go to start of mutual capacitance touch data array.

[73] At operation 402, it is determined whether all the nodes in the mutual capacitance touch data array are checked. The nodes are also referred as points.

[74] If all of the nodes in the mutual capacitance touch data array are checked in operation 403, then the process of identifying the region terminates as shown in operation 403, and control flow is transferred to the flow chart illustrated in FIG. 4B.

[75] If all of the nodes in the mutual capacitance touch data array are not checked, then at operation 404, it is determined whether a current value of the node is 1. If the current value of mutual capacitance of the node is not 1, then the method proceeds to operation 410, which is described below in more detail.

[76] However, if the current value of mutual capacitance of the node is 1, then it is determined whether at least one value around the current node in all eight directions is greater than one, at operation 405.

[77] If there exists one or more nodes having a mutual capacitance value greater than one in operation 405, then the connected region value is updated with the greatest node value obtained from a surrounding current node at operation 406.

[78] Likewise, at operation 407, the current node value and all the nodes surrounding the current node are updated by the updated connected region value.

[79] If, in operation 405, there are no surrounding nodes having a mutual capacitance value greater than 1, then the connected region value is updated with the initial region value as indicated in operation 407.

[80] Further, the region value is incremented in operation 408 and the control flow is directed towards operation 407. Upon completion of updating the connected region

value in operation 407, a next node in the mutual capacitance data array is selected at operation 410.

[81] FIG. 4B is a flow chart illustrating a method of modifying wrongly interpreted region values in a process of region identification according to an embodiment of the present disclosure.

[82] Referring to FIG. 4B, in order to rectify a wrong interpretation of region values, an initial value of a current region is set as 2 at operation 411.

[83] Further, if it is determined that the current region value is greater than the final region value at operation 412, then the process is terminated as indicated in operation 413.

[84] If the current region value is less than the final region value, then the control flow is transferred to the beginning of the mutual capacitance data array at operation 414.

[85] Likewise, at operation 415, it is determined whether all nodes in the mutual capacitance data array are checked. If all nodes in the mutual capacitance data array are checked, then at operation 416, the current region value is incremented and the process returns to operation 412.

[86] If all nodes in the mutual capacitance data array are not checked in operation 415, then it is determined whether the value of the mutual capacitance at the current node is equal to the current region value at operation 418. If the value of the mutual capacitance at the current node is equal to the current region, then it is determined whether the value around current node in surrounding 8 directions is greater than zero and not equal to current region value in operation 419.

[87] If yes, in operation 419, then an error region value is set as a value obtained in the previous processing at operation 420. Consequently, at operation 421, the all nodes in the mutual capacitance data array that are equal to error region value are set as current region value.

[88] If the value of the mutual capacitance at the current node is not equal to the current region value at operation 418 and if no at operation 419, then the process proceeds to operation 417 to get a next point of the mutual capacitance data array. Further, after operation 421, the process proceeds to operation 417.

[89] FIG. 5 illustrates a schematic representation of determining an orientation of a shape using an average angle method according to an embodiment of the present disclosure.

[90] Referring to FIG. 5, the various parameters to be calculated in order to determine a shape of a touch region are one or more of an area of a region, a width of a region, height of a region, a right-left slant length of a region, a left-right slant length of a region, a number of touch nodes enclosed in a region, a hypotenuse of a region, a rectangularity of a region, an elongatedness of a region and an average angle of a region.

[91] In order to find a minimum bounding rectangle, a method for calculating a bounding

box is used.

[92] The area of the bounding box = (Width X Height) of the bounding box.

[93] Perimeter of the bounding box = 2X(Width + Height) of the bounding box.

[94] Centroid of shape(x,y) =($\sum$xi /n, $\sum$yi /n), where i $\in$ [1,n], n is total number of points detected.

[95] xi: x coordinates of points inside the shape.

[96] yi: y coordinates of points inside the shape

[97] Area of shape = Total number of points touched inside the shape.

[98] Perimeter of shape = Total number of points covered on border of shape.

[99] In order to identify the orientation of the shape average angle method is used. According to an embodiment of the present disclosure, straight lines are drawn inside the shape at different angles and each line above threshold and then find average of all the angles.

[100] The average angle method (the angle of line)

[101]

[102] $$\alpha i \tan^{-1}\left(\frac{y2-y1}{x2-x1}\right)$$ ...Equation 1

[103] Average angle of shape = $\sum\alpha$i /n, where i $\in$ [1,n], n is total number of points detected

[104] The operations involved in determining left to right width:

[105] Operation 1: find the point of shape closest to left-top point of the bounding box

[106] Operation 2: find the point of shape closest to right-bottom point of the bounding box

[107] Operation 3: Count all the points lying on the line joining the points mentioned in operations 1 and 2.

[108] The operations involved in determining right to left width.

[109] Operation 1: find the point of shape closest to right-top point of the bounding box

[110] Operation 2: find the point of shape closest to left-bottom point of the bounding box

[111] Operation 3: Count all the points lying on the line joining the points mentioned in operations 1 and 2.

[112] FIG. 6 is a flow chart illustrating a method of identifying various parts of a hand and separating joined fingers according to an embodiment of the present disclosure.

[113] Referring to FIG. 6, in an embodiment of the present disclosure, a fist identification is performed at operation 601. A method of identifying the fist is explained in detail in FIG. 7A.

[114] Further, at operation 602 it is determined that whether the fist is present in the touch region. If the fist is present in the touch region, then the fist count is increased and all points in identified region are reset to 0 at operation 603.

[115] If the fist is not identified in the touch region in operation 602, then finger identi-

fication is performed at operation 604. The method of identifying a finger is explained in detail in FIG. 8A.

[116] If the finger is present in the touch region in operation 605, then a finger count is incremented and all the point present in the touch region are rest to 0 at operation 606.

[117] If the finger is not present in the touch region in operation 605, then it is determined whether any undefined region is present in the touch region in operation 607.

[118] If no undefined region is present in the touch region, then the process terminates at operation 608.

[119] If the at least one undefined region is detected in operation 607, then it is determined whether the current unidentified shape is different from an unidentified shape of a previous iteration at operation 609.

[120] If the current unidentified shape is not matching the unidentified shape of the previous iteration, then the palm count is increased and the process terminated as indicated in operation 610.

[121] If the current in the identified shape matches the unidentified shape of the previous iteration at operation 609, then at operation 611, the finger identification is performed.

[122] FIG. 7A is a flow chart illustrating a method of identifying a fist according to an embodiment of the present disclosure.

[123] Referring to FIG. 7A, in order to identify a fist, parameters such as a height to width ratio, left to right length, and a perimeter of a bounding box are used. A predefined range is set for each of the parameters.

[124] According to an embodiment of the present disclosure, at operation 701, a height to width ratio is calculated. In order to identify a shape of a fist, the height to width ratio must be within a constant range. In an embodiment of the present disclosure, the ideal height to width ratio is approximately 3.

[125] If the height to width ratio falls in a predefined ratio, then a left to right ratio and a right to left ratio are calculated at operation 702. Further it is determined that the left to right length and right to left length fall within a predefined range (approximately 1.3 if LR>RL or 0.75 if RL>LR) at operation 702.

[126] Then the perimeter of shape or perimeter of bounding box is determined at operation 703. If the perimeter of shape or perimeter of bounding box falls in a constant range fixed for fist (approximately 0.75) in operation 703, then the area of the bounding box and the area cover by the hand shape are computed at operation 704.

[127] Further, at operation 704, a ratio of the area of the bounding box and the area covered by the hand shape is computed. If the computed ratio falls within a predefined range, then the fist is identified at operation 705. The fist is not identified in operation 706, if any of the above calculated parameters do not fall in the predefined range (i.e., no at operations 701, 702, 703 or 704). The predefined ranges are set based on a study

conducted on hands of various people to identify possible measures of fist.

[128] FIG. 7B is a schematic representation of a shape of a fist along with a few parameters such as height, width, left to right length and right to left length according to an embodiment of the present disclosure.

[129] Referring to FIG. 7B, a height (H) of a fist, a width (W) of the fist, a left to right length (LR) of the fist and a right to left length (RL) of the fist are illustrated.

[130] FIG. 8A is a flow chart illustrating a method of identifying a finger according to an embodiment of the present disclosure.

[131] Referring to FIG. 8A, in an embodiment of the present disclosure, a right to left diagonal length, and a left to right diagonal length of a hand shape are computed to identify a finger.

[132] At operation 801, it is determined whether a maximum of the left to right diagonal length and the right to left diagonal length falls within a predefined range that is fixed for a finger. In an embodiment of the present disclosure, the predefined range is 2.5.

[133] If the maximum of the left to right diagonal length and the right to left diagonal length falls within the predefined range at operation 802, then at operation 803, a perimeter of a shape or a perimeter of a bounding box is computed and it is determined whether the perimeter of the shape or the perimeter of the bounding box lies within a range fixed for a finger. In an embodiment of the present disclosure, the approximate range is set as approximately 1.

[134] Further, at operation 804, a ratio between the area of bounding box and the area covered by the finger region is calculated. If the ratio falls between predefined ranges, then the finger is identified at operation 805. In an embodiment of the present disclosure, the predefined range of ratio between the area of bounding box and area covered by the finger region is set approximately as 0.7 to 0.9. In case any of the above mentioned ratio of parameters do not fall within the corresponding predefined ratio (i.e., no at operations 810, 802, 803 or 804), then the process terminates without identifying the finger, as indicated in operation 806.

[135] FIG. 8B is a schematic representation of various parameters such as height, width, left to right diagonal length and right to left diagonal length corresponding to a shape of a finger according to an embodiment of the present disclosure.

[136] Referring to FIG. 8B, a height (H) of a shape of a finger, a width (W) of the shape of the finger, a left to right diagonal length (LR) of the shape of the finder and a right to left diagonal length (RL) of the shape of the finger are illustrated.

[137] FIG. 9A illustrates a flow chart of a method of separating/differentiating a finger from a palm according to an embodiment of the present disclosure.

[138] Referring to FIG. 9A, in an embodiment of the present disclosure, at operation 901, a bounding box for the touch region is obtained.

[139] At operation 902, a first encountered region from top left is obtained. The first encountered region in view of the present embodiment may be a left most finger.

[140] Further, traversing all points of the encountered region along a width of the bounding box and storing its length are performed at operation 903.

[141] Then at operation 904, a next row of the bounding box is selected.

[142] Further, a count of length of a previously encountered region in a current row is performed at operation 905.

[143] Then an absolute value of a difference between the previously stored length and the length of the current region is calculated, at operation 906, where the difference is referred as delta.

[144] At operation 907, it is determined whether the delta value is greater than a predefined threshold value. In an embodiment of the present disclosure, the predefined threshold is set as 4. In parallel, the length of encountered region in the current row is set as zero. If the value of delta is less than the predefined threshold, then control flow transfers to operation 904, for selecting the next row of the bounding box. If the value of delta is higher than the predefined threshold value, then at operation 908, all values in the current row are reset to zero and the process terminates.

[145] FIG. 9B is a schematic representation of various operations present in separating a finger from ta palm according to an embodiment of the present disclosure.

[146] Referring to FIG. 9B, at operation 1, a finger is separated from a palm. Likewise, a bounding box is created for the rest of the touch region at operations 2 and 3 and each of the finger and palm are identified at operation 4.

[147] FIG. 10 is a flow chart illustrating a method of storing shapes according to an embodiment of the present disclosure.

[148] Referring to FIG. 10, a method to record one or more shapes in a touch sensitive device for reference a user's needs to select an option to record a shape is illustrated. According to an embodiment of the present disclosure, a blank area is provided in the touch sensitive device for providing an input for recording a gesture using a hand shape at operation 1001. The user needs to repeat the same gesture multiple times for recording minor deviations present in the hand shape while providing the touch input. Each of the touch input frames are stored in the touch sensitive device memory.

[149] If data for a consecutive predefined number of frames is similar at operation 1002, then at operation 1003, mutual capacitance of the data is processed. Further, in operation 1003, the parameters are computed for the shape.

[150] If the data for the consecutive predefined number of frames is not similar in operation 1002, then the method returns to operation 1001.

[151] Once the hand shape is validated and parameters are computed, the parameters and corresponding shapes are stored in the touch sensitive device in operation 1004.

[152] FIG. 11 is a flow chart illustrating a method of shape matching using a calculated parameter and recorded parameters according to an embodiment of the present disclosure.

[153] Referring to FIG. 11, in order to perform shape matching, at operation 1101, it is determined whether a number of regions detected is equal to a number of recorded regions.

[154] If the number of region detected is equal to the number of recorded region, then, in operation 1102 the following absolute values are computed and compared with a predefined ratio (Ω).

[155] ABS(W/LR-RECORDED(W/LR))<Ω

[156] ABS(H/LR-RECORDED(H/LR))< Ω

[157] ABS(W/RL-RECORDED(W/RL))< Ω

[158] ABS(H/RL-RECORDED(H/RL))< Ω

[159] where, W is the width of the shape, H is the height of the shape, LR is the left to right length of the shape and RL is the right to left length of the shape.

[160] According to an embodiment of the present disclosure, an approximate value of the predefined value (Ω) falls within 0.25. If the value of Ω is within the range of 0.25 for at least three conditions as mentioned above, the further comparisons are performed to determine the match from the recorded shapes.

[161] At operation 1103, a difference between a value of a perimeter of the touch shape and the recorded perimeter is computed. Further, an absolute value of the computed difference is compared with a predefined value for corresponding shape. In an embodiment of the present disclosure, the predefined value is approximately 10.

[162] If the computed value is less than the predefined value, then at operation 1104 a difference between a recorded angle and a current angle is determined. Then an absolute of the difference is compared with a predefined angle. According to an embodiment of the present disclosure, the predefined angle is approximately 20.

[163] If the determined difference is less than the predefined angle, then the centroid gap is considered as another parameter for matching touch shape. At operation 1105, a difference between the centroid gap in the touch shape and a recorded centroid gap is calculated. Then the calculated difference is compared with the predefined value.

[164] If the difference in centroid gap is less than the predefined value at operation 1105, then at operation 1106, the matched shape is obtained. If any of the comparisons mentioned in operations 1101, 1102, 1103, 1104, and 1105 fail, then the shape is not matched with any of the recorded shapes in operation 1107.

[165] FIG. 12A is a schematic representation of launching a camera using a single finger shape identification according to an embodiment of the present disclosure. FIG. 12A illustrates a use case of a shape based a touch action launch.

[166] Referring to FIG. 12A, according to an embodiment of the present disclosure, a camera is launched if a single finger is detected at a certain orientation over a screen, even if a display panel of a touch screen device is in a screen-off state. The touch screen device recognizes touch shapes in the screen-off state by identifying hand/finger shapes using a non-model based approach. FIG. 12A illustrates that N seconds after the performance of a gesture, the camera is launched.

[167] FIG. 12B is a flow chart illustrating a method of launching a camera using a single finger shape identification according to an embodiment of the present disclosure.

[168] Referring to FIG. 12B, at operation 1201, a single finger is identified.

[169] Further, a centroid of a left half and a bottom half of a finger is checked at operation 1202.

[170] Then an orientation is defined as 3, which corresponds to 270 degrees (landscape mode), in operation 1203.

[171] Then it is determined that whether the left to right diagonal of the finger is greater than the right to left diagonal of the finger at operation 1204. If the left to right diagonal is greater than the right to left diagonal, then the average angle method is implemented to identify the orientation of the shape.

[172] The average angle is determined and compare with 30 degrees and 0 degrees at operation 1205. If the average angle is between 30 degrees and 0 degrees, then it is determined whether the centroid of the finger is less than one third of a width of the finger in operation 1206.

[173] If the above condition of operation 1206 is satisfied, then a shape corresponding to a camera is identified and the camera is launched in the touch screen device at operation 1207.

[174] In an embodiment of the present disclosure, after operation 1202, the centroid of the finger, the right half and the bottom half of the finger is checked to identify the shape in operation 1208.

[175] In such cases the orientation is defined as 1, which corresponds to 90 degrees (portrait mode) as indicated in operation 1209.

[176] Further, it is determined whether right to left diagonal is greater than the left to right diagonal at operation 1210. If the right to left diagonal is greater than the left to right diagonal, then the average angle method is applied to define the shape.

[177] The average angle is determined and compared with 30 degrees and 0 degrees at operation 12011. If the average angle is between 30 degrees and 0 degrees, then the following condition is determined at the operation 1212:

[178] centroid > 2* width/3.

[179] If the above condition is satisfied, then the shape corresponding to the camera is identified and the camera is launched in the touch screen device at operation 1207.

[180] If at least one of the conditions of operations 1204, 1205, 1206, 1210, 1211, and 1212 is not satisfied, then the touch sensitive device waits until user lifts the finger as indicated at operation 1213.

[181] FIG. 13 is a schematic representation of launching a camera using double finger shape identification according to an embodiment of the present disclosure.

[182] Referring to FIG. 13, when a user places two fingers at a certain orientation on a screen of a touch sensitive device for a certain amount of time (e.g., N seconds), a camera launch is performed.

[183] FIG. 14 is a flow chart illustrating a method of performing a touch action in a touch-sensitive device according to an embodiment of the present disclosure.

[184] Referring to FIG. 14, a user touches a touch sensitive device at operation 1401.

[185] Further, at operation 1402, the touch sensitive device detects a validity of the touch. In an embodiment of the present disclosure, the TSP IC provides mutual capacitance data or touch coordinate. The touch coordinate data is processed data. The validation of touch data is different for mutual capacitance data and touch coordinate data. The mutual capacitance data is a two-dimensional (2D) array input data. Each cell of the array indicates the mutual capacitance value or coupling capacity of two-crossing conductor/electrodes paths in the touch screen panel. The first operation involved in validating the input data is binarization of the input array with respect to a predetermined threshold value. This binarized data is used to identify the region of touch. The region of touch is also referred as cluster. The mutual capacitance data is considered as valid if it contains one region indicating one touch point. If the input data is touch coordinate data, then the data received is the set of parameters for all the touch locations or pointers. The set of parameters may include the two (or three) dimensional location parameters of the pointer as well as other parameters such as the area of the pointer, length and width of the pointer with respect to boundaries of the screen. The number of touch regions is determined using the parameters. The touch coordinate data is considered as valid if it contains one cluster/region indicating one touch point.

[186] If the touch is not determined to be valid at operation 1402, then the touch-sensitive device proceeds to wait for a predefined number of frames in operation 1403 and then proceeds to operation 1401 after the predefined number of frames.

[187] If the touch is determined to be valid at operation 1402, then at operation 1404, the validated input data is stored. In order to store the input data for usage in further computation operations, additional computation operations may be involved. If the input data is mutual capacitance data, the identified region is a set of adjacent node values. This node values are converted to a point by computing a statistical parameter such as mean of various coordinate values of the points. Further, optionally convert the node values to the touch coordinate value by multiplying with the appropriate factor. The

appropriate factor is a scaling factor based on the resolution and the number of electrodes present in the touch sensitive device. If the input data is touch coordinate data, then no additional computation is required. The necessary parameters obtained from the input data are stored in the touch sensitive device.

[188] At operation 1405, it is determined whether a gesture is completed or still in progress, (e.g., the user is still drawing the gesture). The assertion to check if the gesture is completed or not is done based on the number of empty frames (no-touch frames) received for both input data types (i.e., mutual capacitance data and touch coordinate data). The threshold for number of empty frames is the threshold time to be waited to confirm the end of gesture. The detection of end of gesture is important for gestures with multiple strokes. If multi-strokes gestures are not supported by the touch sensitive device, the detection of end of gesture fixes to handle erroneous empty data frames.

[189] For mutual capacitance data, if the number of consecutive empty frames exceeds the threshold number, the gesture has ended. If any non-empty frame (touch frame) is received in between, then the algorithm resumes normal operation and continues normally for multi-strokes supported algorithms. In a case of single stroke algorithms this threshold is set to be very low only to compensate for erroneous empty frames. For touch coordinate data, the check for gesture completed is triggered by a pointer up event. The check for a non-empty frame is the occurrence of a pointer move or down event.

[190] If the gesture is determined to not be completed in operation 1405, then the touch-sensitive device returns to operation 1402.

[191] After the gesture is completed in operation 1405, then at operation 1406, the parameters are computed either using the mutual capacitance or the touch coordinate data. In an embodiment of the present disclosure, there are two approaches present in order to compute the parameters. The first method is based on directions of edges assuming the gesture to be linear and linearizing the gesture if it is not linear. The second method is based on cosine distance between the performed and every recorded gesture. All these computations may be performed in the IC or the application processor. The computation of parameters according to one embodiment of the present disclosure is explained in detail in FIGS. 13, 14, and 15.

[192] Further, at operation 1407, the computed parameters are validated.

[193] If the computed parameters are determined to not be valid at operation 1407, then the touch-sensitive device proceeds to operation 1403.

[194] After the computed parameters are determined to be valid at operation 1407, then at operation 1408, the corresponding match is found from the one or more registered or predefined matches. The method operations involved in finding corresponding match

from the one or more registered gesture is explained in detail in FIG. 17.

[195] After the match is found at operation 1408, at operation 1409, the corresponding touch action is enabled in the touch sensitive device.

[196] FIG. 15 is a flow chart illustrating a method of computing parameters and linearizing a gesture performed by a user according to an embodiment of the present disclosure.

[197] Referring to FIG. 15, in an embodiment of the present disclosure, a gesture is linearized to obtain endpoints. These endpoints form edges, and direction indices of these edges are parameters used to compare and match gestures performed by the user to one or more registered gestures.

[198] The first two points of the touch input provided by the user are considered as the first edge. Then, at operation 1501, the next point is given as the input. The next point is checked to find, whether the next point is collinear with the first edge or not.

[199] Further, at operation 1502, it is determined whether the number of points already processed is equal to zero.

[200] If the number of points processed is zero, then the next point is taken as the first point and the edge is started from the next point as indicated in operation 1503. If the number of points processed are not equal to zero, then it is determined that whether the previously processed points are collinear or not at operation 1504.

[201] If the next point is collinear in operation 1504, then it is combined with the current edge, the new endpoints being one of the initial endpoints and the new point. If it is not collinear in operation 1504, then the latest endpoint of the current edge and the new point are considered as endpoints of the new current edge, so as to update the previous points at operation 1505.

[202] If the next point is collinear in operation 1504, then at operation 1506, the angle between the previous two edges formed by current point and previous two vertices is computed.

[203] Next, it is determined whether the angle computed in operation 1504 is less than a threshold value in operation 1507. If the computed angle is less than the predefined threshold, then a length of the edge is determined and compared to a threshold at operation 1509. The predefined threshold is set based on a number of directions approach that has been selected to compute direction. For example, a 4 direction or an 8 direction approach is used to compute the direction. If the angle is more than the predefined threshold at operation 1507 then the previous point is replaced with the current point at operation 1508. If its length is not less than a certain predefined threshold, as determined at operation 1509, then the previous point is replaced with the current point at operation 1508.

[204] Further, the previous edge is discarded if its length is less than a certain predefined threshold value at operation 1509 and the two edges are merged as indicated in

operation 1510, with the final edge having the first point of previous edge and latest point of current edge as the new current edge.

[205] The remaining points to be processed are determined at operation 1511. If there are points to be processed, then a direction is computed for all the remaining points by returning to operation 1501. If there are no more points to be processed, then the direction is computed at operation 1512.

[206] Further, after operations 1503, 1505 and 1508, the process returns to operation 1501.

[207] FIGS. 16A and 16B are schematic representations of computing directions according to various embodiments of the present disclosure.

[208] After linearizing the entire gesture using the method explained in FIG. 13, the directions of each edge are computed.

[209] Referring to FIG. 16A, an illustration is provided in which directions of the edges are computed within various ranges varying from 4 directions, 8 directions to 360 directions. These directions are computed based on the vectors obtained by considering consecutive endpoints of each edge starting from the initial point. FIG. 16A depicts the boundaries for various direction values assigned to an edge for an 8 direction. All the regions need not be of the same size necessarily. The region around each of the direction is considered as zone corresponding to that direction. For instance, the direction assigned to any edge within the zone 1 is 0 and the zone 2 is 4.

[210] Referring to FIG. 16B, an example gesture with two edges having directions 4 and 7 is illustrated. The example gesture contains two edges. Based on the inclination of the edge, directions have been computed as 1 and 7 for each of the two edges respectively.

[211] FIG. 17A is a flow chart illustrating a method of computing one or more parameters for matching according to an embodiment of the present disclosure.

[212] Referring to FIG. 17A, gesture coordinates are collected in the TSP IC. While collecting coordinates of a gesture, additional operations may be performed in the interval between two strokes such as calculating and storing stroke lengths. When a user touches a screen, a touch sensitive device detects whether the user starts making a gesture it as indicated in operation 1701.

[213] Further, it is determined whether the user is moving their finger to provide the gesture at operation 1702. If the user moves their finger in order to make the gesture, then it determines whether the current coordinates are the beginning of the gesture at operation 1703.

[214] If the current coordinates are the beginning of the gesture, then the current coordinates are added to an array of coordinates as shown in operation 1707 and then the method returns to operation 1702. In an embodiment of the present disclosure, a completion of the gesture is iteratively determined.

[215] If the current coordinates are not the beginning of the gesture in operation 1703, then

a distance between the current coordinates and the last coordinates is calculated in operation 1708. Further, in operation 1708 it is determined whether the distance between the current coordinates and the last coordinates is greater than or equal to a predefined threshold distance. The predefined threshold distance is set based on sampling rate.

[216] If the distance between the current coordinates and the last coordinates is greater than or equal to predefined threshold distance, then the coordinates of current points are added to coordinate array in operation 1707. If the distance between the current coordinates and the last coordinates is less than the predefined threshold distance, then control flow transfers to operation 1702.

[217] In case, the user has lifted the finger in operation 1702 (e.g., the user is no longer moving their finger on the screen), then a stoke length is calculated and stored at operation 1704 in the touch sensitive device.

[218] Then it is determined whether a time limit has expired at operation 1705. The gesture may contain multiple strokes; hence an empty data frame does not necessarily indicate the end of the gesture. So the time limit is the time interval between one or more strokes of a gesture.

[219] If the time limit has not expired in operation 1705, then at operation 1706, it is determined whether the user makes a new stroke or not.

[220] If the user makes the new stoke corresponding to the gesture, then the coordinate of the new stroke is added to the array of coordinates in operation 1707.

[221] If the user does not make a new stroke in operation 1706, then the method returns to operation 1705. In operation 1705, if the time limit has expired, then the process of collecting coordinates stops and the processing of the collected coordinates starts as indicated at operation 1709.

[222] FIG. 17B is a flow chart illustrating a method of processing one or more coordinates collected using the method explained in FIG. 17A according to an embodiment of the present disclosure.

[223] Referring to FIG. 17B, collected coordinates are processed in a TSP IC. The computation required to convert the input data into parameters is done in the IC itself or partially done in another processor.

[224] Once the user completes the gesture and the coordinates are collected at operation 1710, then collected coordinates are sampled at operation 1711.

[225] The centroid of the sampled coordinates is calculated at operation 1712. The gesture performed on the touch sensitive device depends on the orientation of the touch sensitive device.

[226] Hence, at operation 1713, it is determined whether the gesture matches an orientation invariant. If the gesture matches the orientation invariant, an orientation angle of the

sampled coordinates is calculated at operation 1714.

[227] Further, the sampled coordinates are translated by keeping the centroid as origin at operation 1715.

[228] Further, at operation 1716, the translated coordinates are rotated around the origin by the calculated orientation angle.

[229] If the gesture does not match the orientation invariant in operation 1713, then the sampled coordinates are merely translated by keeping the centroid as the origin as shown in operation 1717. Further, after operations 1716 and 1717, the processed coordinates are sent to an application processor for storing at operation 1718.

[230] FIG. 17C is a flow chart illustrating a method of matching a gesture performed by a user with a registered gesture according to an embodiment of the present disclosure.

[231] Referring to FIG. 17C, in an embodiment of the present disclosure, a registered gesture and a user performed gesture are compared based on a cosine distance between them. The processed coordinates are used for matching the user performed gesture at operation 1721.

[232] At operation 1722 the processed coordinates are normalized and used for matching against recorded or predefined gestures, specified by the user. Scores are generated based on cosine distance computed between the performed and each recorded or predefined gesture individually.

[233] The number of strokes of the performed gesture is compared with the number of strokes of the gesture it is being matched with to generate the score, such that the score is not generated if there are no strokes at operation 1723.

[234] If the score is generated in operation 1723, then a best score obtained for the performed gesture is compared with a predefined minimum expected value, determined based on the recorded instances of the gesture, and a determination is made as to whether the best score is greater or equal to the predefined minimum expected value at operation 1724.

[235] If the best score is greater than or equal to the predefined minimum expected value, then an order of points of touch and strokes of the performed gesture is compared with the order of points of touch and strokes of the gesture with the best score, such that a determination is made as to whether the gesture contains dots and whether the dots are drawn in a correct order at operation 1725.

[236] If operation 1725 determines that the gesture contains the dots and that the dots are drawn in the correct order, then a ratio of a perimeter of the minimum bounding box of the performed gesture to a length of the performed gesture is compared with a ratio of a perimeter of the minimum bounding box of the gesture with the best score to a length of the gesture with the best score and a determination is made as to whether differences between the ratio of the performed gesture and the ratio of the gesture with the best

score is within a limit at operation 1726.

[237] If operation 1726 determines that differences between the ratios is within the limit, then, a determination is made that the match is found at operation 1727.

[238] If any of the comparisons in operations 1723, 1724, 1725 and 1726 fails, then a determination is made that the match for the user performed gesture is not found is operation 1728 for the user performed gesture. Otherwise, a match is found for the performed gesture and the corresponding touch action is enabled in the touch sensitive device.

[239] FIG. 18A is a flow chart illustrating a method of computing a pressure vertex and a direction vertex for determining a gesture performed by a user according to an embodiment of the present disclosure. In this approach, pressure data is used along with touch data to characterize a gesture. Similar shapes performed on a screen are considered as different gestures based on a number of times pressure is applied and released at each vertex of the gesture.

[240] Referring to FIG. 18A, at operation 1801, touch data and pressure data are extracted from the TSP IC.

[241] Based on the extracted touch data, a gesture is linearized and directions of each edge forming the gesture are computed at operation 1802.

[242] Then a location of a pressure point, a number of times the pressure is applied at that location and a pressure vertex are stored at operation 1803.

[243] Then pressure peaks are found using the stored pressure points and the stored pressure points are validated based on a predefined pressure peak threshold at operation 1804.

[244] If a user lifts their finger from a screen of a touch sensitive devices at operation 1805, then a direction vertex is calculated at operation 1806 and the direction vertex and the pressure vertex are compared and if they match the user performed gesture with the registered gestures then a pressure count is saved at operation 1807.

[245] If the finger of the user is not lifted from the screen of the touch sensitive device, then control flow transfers to operation 1801, for iteratively checking the pressure and touch data for the next coordinates.

[246] FIG. 18B is a flow chart illustrating a method of matching a gesture performed by a user with a registered gesture according to an embodiment of the present disclosure.

[247] Referring to FIG. 18B, in an embodiment of the present disclosure, a registered gesture and a user performed gesture are compared in terms of direction of touch data and pressure count.

[248] At operation 1808, a number of directions of touch data of the user performed gesture is compared with a number of directions present in a registered gesture.

[249] If the number of directions of the touch data is equal to the number of directions

present in the registered gesture, then direction values of the touch data and the registered gesture are compared at operation 1809.

[250] If the direction values of the touch data of the user performed gesture are equal to the direction values of the registered gesture, then a pressure counts of the touch data and the registered gesture are compared at operation 1810.

[251] If the pressure count of the touch data of the user performed gesture is the same as the pressure count of the registered gesture, then the match is a success at operation 1811. If any of the above comparisons in operations 1808, 1809 and 1810 fails, then the match for the user performed gesture is not found for the user performed gesture as shown in operation 1812.

[252] FIG. 19 is a flow chart illustrating a method of registering one or more gestures with a touch sensitive device according to an embodiment of the present disclosure.

[253] Referring to FIG. 19, in an embodiment of the present disclosure, a set of predefined trails gesture patterns are provided as reference to create a new gesture. In order to register one or more gesture with a touch sensitive device, mutual capacitance data and touch coordinate data of a touch input by a user from the TSP IC are processed and sent to the application processor at operation 1901.

[254] Further, the touch sensitive device provides an option to the user to enter the same gesture multiple times in order to detect the minute variations in the gesture performed at each time at operation 1902. An average score is computed by taking average of all scores generated during performing gesture at each time. The scores are generated by matching gesture with stored or predefined gestures.

[255] At operation 1903, it is determined whether a current gesture matches an already recorded gesture or whether the current gesture is different than previous trials of same gesture.

[256] If any of the above conditions is satisfied, then the user performed gesture is discarded at operation 1905.

[257] If the user performed gesture does not satisfy any of the conditions of operation 1903, then the gesture is assigned an identifier (ID) and name as shown in operation 1904. Further, the gesture is stored in a persistent storage such as a database or a file, along with ID, name, list of processed coordinates, and average minimum score in operation 1904.

[258] FIG. 20 is a schematic representation of launching a camera using a single finger gesture identification according to an embodiment of the present operation. The figure illustrates a use case of gesture based touch action.

[259] Referring to FIG. 20, according to an embodiment of the present disclosure, a camera is launched if a single finger gesture at a certain orientation is detected over a screen even if a display panel of a touch screen device is in a screen-off state. The touch

screen device recognizes touch gestures in the screen-off state by identifying hand/finger shapes at certain orientation using a non-model based approach. FIG. 20 illustrates that N seconds after the performance of gesture, the camera is launched.

[260] FIG. 21A is a schematic representation of launching a camera using double finger gesture identification according to an embodiment of the present disclosure.

[261] Referring to FIG. 21A, when a user swipes two fingers at a certain orientation on a screen of a touch sensitive device, ta camera launch is performed after N seconds.

[262] FIG. 21B is a flow chart illustrating a method of launching a camera using double finger gesture identification according to an embodiment of the present disclosure.

[263] Referring to FIG. 21B, a touch screen identifies a shape in a screen-off state. In the present use case, a camera is launched using two fingers. The shape is determined using a centroid. At operation 2101, a centroid C1 of a first finger is stored if the camera shape is identified.

[264] Further, a centroid of a second finger shape is tracked and updated at operation 2102. Then the difference between the centroid of the first finger (C1) and the centroid of the second finger (C2) is computed.

[265] If the computed difference is greater than 3 electrodes, then a time counter for holding is increased as indicated in operation 2103, and then the centroid of the second finger is compared with an angle at operation 2104.

[266] Further, in operation 2104, a determination is made as to whether C2 > width/2 and as to whether a final angle > 60. If the determinations are negative, then the method returns to operation 2102. If the determinations are positive, then the method proceeds to operation 2105.

[267] Further, at operation 2105, it is determined whether the counter for holding is greater than a predefined number (e.g., 10). The predefined number is set based on the sampling frequency. If the above condition fails, then at operation 2106, it is declared that no shape is identified, C1 is erased and the camera grip detection starts again.

[268] If the counter for holding is greater than the predefined number at operation 2105, then a sliding time is determined at operation 2107. If the sliding time is less than 2 seconds in operation 2107, then the camera is launched in the touch sensitive device at operation 2108.

[269] If the sliding time is not less than 2 seconds in operation 2107, then the method returns to operation 2101.

[270] FIG. 22 is a schematic representation of launching various touch actions based on shape identification according to an embodiment of the present disclosure.

[271] Referring to FIG. 22 six use cases are illustrated, where a touch input is provided on a touch screen of a touch sensitive device that is in a screen-off state.

[272] The first use case 1 illustrates that a user performs a touch restore using two thumbs

in the screen-off state for launching a camera. Case 1 is explained in detail in FIGS. 12A and 20 of the present disclosure.

[273] Likewise, the second use case 2 illustrates a launch of a front camera when the user swipes the screen, as explained in detail in FIGS. 12B and 21B.

[274] In the third use case 3, the gestures are designed in a form of a letter. The user is allowed to give touch input in the form of a specific letter or a pattern. For instance, the touch input in the form of a first letter of an application performs launching of the application. Consider that the touch input in the shape of C, then the chat on application is launched in the touch sensitive device.

[275] According to fourth use case 4 as detailed in FIG. 20, the touch screen of the touch sensitive device shows a time and a missed call notification when the user performs a gesture such as pinch out when the touch screen of the touch sensitive device is in the screen-off state.

[276] The fifth use case 5 depicts a small swipe gesture. When the user provides the touch input as a small swipe, then according an embodiment of the present disclosure, the touch sensitive device displays a short cut for various applications. The short cut to be displayed may be configured by the user according to preferences.

[277] The sixth use case 6 illustrates a gesture used for unlocking the screen. The user may draw any preconfigured shape on the touch screen of the touch sensitive device, when it is in the screen-off state to unlock the touch screen. For instance, the user may configure a star shape for unlocking the screen. Once the same is performed, the screen unlocks.

[278] Various aspects of the present disclosure can also be embodied as computer readable code on a non-transitory computer readable recording medium. A non-transitory computer readable recording medium is any data storage device that can store data which can be thereafter read by a computer system. Examples of the non-transitory computer readable recording medium include Read-Only Memory (ROM), Random-Access Memory (RAM), CD-ROMs, magnetic tapes, floppy disks, and optical data storage devices. The non-transitory computer readable recording medium can also be distributed over network coupled computer systems so that the computer readable code is stored and executed in a distributed fashion. Also, functional programs, code, and code segments for accomplishing the present disclosure can be easily construed by programmers skilled in the art to which the present disclosure pertains.

[279] At this point it should be noted that various embodiments of the present disclosure as described above typically involve the processing of input data and the generation of output data to some extent. This input data processing and output data generation may be implemented in hardware or software in combination with hardware. For example, specific electronic components may be employed in a mobile device or similar or

related circuitry for implementing the functions associated with the various embodiments of the present disclosure as described above. Alternatively, one or more processors operating in accordance with stored instructions may implement the functions associated with the various embodiments of the present disclosure as described above. If such is the case, it is within the scope of the present disclosure that such instructions may be stored on one or more non-transitory processor readable mediums. Examples of the processor readable mediums include Read-Only Memory (ROM), Random-Access Memory (RAM), CD-ROMs, magnetic tapes, floppy disks, and optical data storage devices. The processor readable mediums can also be distributed over network coupled computer systems so that the instructions are stored and executed in a distributed fashion. Also, functional computer programs, instructions, and instruction segments for accomplishing the present disclosure can be easily construed by programmers skilled in the art to which the present disclosure pertains.

[280] While the present disclosure has been shown and described with reference to various embodiments thereof, it will be understood by those skilled in the art that various changes in form and details may be made therein without departing from the spirit and scope of the present disclosure as defined by the appended claims and their equivalents.

Claims

[Claim 1] A method of performing a touch action in a touch sensitive device, the method comprising:
sensing a touch input provided on a touch screen of the touch sensitive device;
identifying a shape of a contact area associated with the sensed touch input provided on the touch screen based on one or more predefined parameters;
determining an orientation of the identified shape; and
performing the touch action corresponding to the sensed touch input based on the identified shape.

[Claim 2] The method as claimed in claim 1, wherein sensing the touch input is sensed when a display panel of the touch sensitive device is in a screen-off state.

[Claim 3] The method as claimed in claim 1, wherein the identifying of the shape of the contact area comprises:
detecting a change in a mutual capacitance of the touch screen, by a touch sensor panel integrated circuit (TSP IC) embedded in the touch sensitive device as a result of the touch input,
filtering mutual capacitance data using a binarization method,
identifying one or more touch regions on the touch screen using a region identification method,
defining one or more parameters based on the identified one or more touch regions,
determining a value of each of the one or more predefined parameters for the identified one or more touch regions, and
identifying the shape based on the values determined by the determining of the value of each of the one or more predefined parameters.

[Claim 4] The method as claimed in claim 3, wherein the binarization method comprises:
filtering the mutual capacitance data using a predefined threshold value, and
removing noise using a predefined upper limit.

[Claim 5] The method as claimed in claim 3, wherein the region identification method comprises:
processing the filtered mutual capacitance data to segment the one or

more touch regions based on one or more discontinuities in the mutual capacitance data, each discontinuity, from the one or more discontinuities, marking a beginning of a new region, and
labeling the one or more touch regions based on an order of the identification of the one or more touch regions.

[Claim 6] The method as claimed in claim 3, wherein the one or more parameters comprise one or more of an area of a region, a width of a region, a height of a region, a right-left slant length of a region, a left-right slant length of a region, a number of touch points enclosed in a region, a hypotenuse of a region, a rectangularity of a region, an elongatedness of a region and an average angle of a region.

[Claim 7] The method as claimed in 6, wherein the orientation of the identified shape is determined according to an average angle method comprising:
computing a set of a plurality of angles formed by lines joining boundary points to other boundary points where a distance between the joined boundary points is greater than or equal to a certain computed length, and
determining an average of the computed set of the plurality of angles to determine the orientation of the identified shape.

[Claim 8] The method as claimed in claim 1, further comprising:
determining whether the identified shape is valid based on the predetermined criteria, and
wherein the determining of whether the identified shape is valid comprises:
matching the identified shape with one or more recorded shapes stored in the touch sensitive device by comparing one or more parameters associated with the identified shape with one or more parameters associated with the one or more recorded shapes.

[Claim 9] The method as claimed in claim 8, wherein the matching of the identified shape comprises:
determining that differences between the one or more parameters associated with the identified shape and the one or more parameters associated with one or more recorded shapes lie within a predetermined threshold range.

[Claim 10] The method as claimed in claim 1, further comprising:
registering the identified shape, and
storing one or more parameters in a memory of the touch-sensitive device as the one or more predefined parameters.

[Claim 11] A non-transitory computer-readable storage medium storing instructions that, when executed, cause at least one processor to perform the method of claim 1.

[Claim 12] A method of performing a touch action in a touch-sensitive device, the method comprising:
sensing a touch input provided on a touch screen of the touch-sensitive device;
identifying a shape of a contact area associated with the sensed touch input provided on the touch screen based on one or more predefined parameters;
determining whether the identified shape is valid based on a predetermined criteria;
detecting a gesture based on a movement of a validated shape having been validated by the determining of whether the detected shape is valid;
determining whether the detected gesture is valid by matching the detected gesture with one or more predefined gestures; and
performing the touch action corresponding to the sensed touch input based on a determination that the detected gesture is valid and a determination that the detected shape is valid.

[Claim 13] The method as claimed in claim 12, wherein the detecting of the gesture based on the movement of the validated shape comprises:
computing a centroid of a touch region associated with the contact area,
storing the centroid if a distance between the computed centroid and a previously stored centroid is above a certain predefined threshold,
determining an end of the detected gesture, and
discarding the detected gesture if the detected gesture has not ended within a certain predefined time period.

[Claim 14] The method as claimed in claim 13, wherein the computing of the centroid of the touch region comprises at least one of:
determining the centroid of the touch region based on one or more node values, if an algorithm is configured to use mutual capacitance data, and
determining the centroid of the touch region by using touch coordinates obtained from a touch sensor panel integrated circuit (TSP IC), if the algorithm is not configured to the use mutual capacitance data.

[Claim 15] The method as claimed in claim 13, wherein the determining of the end of the gesture comprises:

determining that a number of nodes above a predefined threshold in a mutual capacitance data frame received from a touch sensor panel integrated circuit (TSP IC) do not lie within a predefined threshold range, if the touch sensitive device is configured to use mutual capacitance data,
determining that an event corresponding to a pointer-up event is received from the TSP IC, if the touch sensitive device is configured to use touch coordinate data, and
checking that another stroke has not started within a predefined time interval, otherwise making a current stroke a part of the detected gesture.

[Claim 16] The method as claimed in claim 12, wherein the determining of whether the detected gesture is valid comprises:
matching the detected gesture with one or more predefined gestures corresponding to the identified shape, if the touch sensitive device is configured to perform shape matching before gesture matching, and
matching the detected gesture with all registered or predefined gestures if the touch sensitive device is not configured to perform the shape matching before the gesture matching.

[Claim 17] The method as claimed in claim 16, wherein matching the detected gesture with the one or more predefined gestures comprises:
determining a match between the detected gesture and the one or more predefined gestures using one of a direction based approach, a direction and pressure based approach, and a minimum cosine distance with additional checks approach.

[Claim 18] The method as claimed in claim 17, wherein the determining of the match using the direction based approach comprises:
filtering a plurality of collected coordinates,
linearizing the filtered coordinates,
computing direction parameters of the linearized coordinates,
matching the computed direction parameters with direction patterns of the one or more predefined gestures, and
determining that differences between the computed direction parameters and the one or more predefined gestures lie within a predefined range.

[Claim 19] The method as claimed in claim 17, wherein the determining of the match using the minimum cosine distance with additional checks approach comprises:

collecting coordinates while the detected gesture is being performed,
sampling the collected coordinates,
calculating a centroid of the sampled coordinates,
calculating an orientation of the sampled coordinates if gesture matching is configured as orientation sensitive,
translating the sampled coordinates by keeping the centroid as origin,
rotating the translated coordinates around the origin by a computed angle, if the gesture matching is configured as orientation sensitive,
computing the minimum cosine distance between the detected gesture and recorded/predefined gestures using the collected coordinates,
performing a dynamic minimum score check,
performing an order of points of touch check if the detected gesture contains multiple strokes, and
performing a perimeter-length ratio check between the detected gesture and a best matched gesture.

[Claim 20] The method as claimed in claim 17, wherein the determining of the match using the direction and pressure based approach comprises:
filtering a plurality of collected coordinates,
linearizing the filtered coordinates,
computing direction parameters of the linearized coordinates,
collecting pressure coordinates at which pressure is applied and a number of times the pressure is applied at the pressure coordinates,
filtering out the collected pressure coordinates having a distance from a nearest vertex of a linearized gesture that is more than a predefined threshold,
matching the computed direction parameters using the direction based approach, and
matching the number of times and an order in which the pressure is applied at each vertex with a registered gesture, if a computed direction matches.

[Claim 21] A non-transitory computer-readable storage medium storing instructions that, when executed, cause at least one processor to perform the method of claim 12.

[Claim 22] A touch sensitive device performing a touch action comprising:
a memory configured to store one or more predefined parameters;
a touch screen configured to sense a touch input;
a processor configured to sense the touch input provided on the touch screen, identify a shape of a contact area associated with the sensed

touch input based on the one or more predefined parameters, determine an orientation of the identified shape, perform the touch action corresponding to the sensed touch input based on the identified shape.

[Claim 23] The touch sensitive device of claim 22, wherein the processor detects a change in a mutual capacitance of the touch screen, by a touch sensor panel integrated circuit (TSP IC) embedded in the touch sensitive device as a result of the touch input, filters mutual capacitance data using a binarization method, identifies one or more touch regions on the touch screen using a region identification method, defines one or more parameters based on the identified one or more touch regions, determines a value of each of the one or more predefined parameters for the identified one or more touch regions, identifies the shape based on the values determined by the determining of the value of each of the one or more predefined parameters.

[Claim 24] The touch sensitive device of claim 22, wherein the processor matches the identified shape with one or more recorded shapes stored in the touch sensitive device by comparing one or more parameters associated with the identified shape with one or more parameters associated with the one or more recorded shapes, determines whether the identified shape is valid, if the identified shape is valid, performs the touch action corresponding the touch input based on the identified shape.

[Claim 25] The touch sensitive device of claim 22, wherein the processor registers the identified shape, stores one or more parameters in a memory of the touch sensitive device as the one or more predefined parameters.

[Claim 26] The touch sensitive device of claim 22, wherein the processor senses a touch input provided on a touch screen of the touch sensitive device, identifies a shape of a contact area associated with the sensed touch input provided on the touch screen based on one or more predefined parameters, determines whether the identified shape is valid based on a predetermined criteria, detects a gesture based on a movement of a validated shape having been validated by the determining of whether the detected shape is valid, determines whether the detected gesture is valid by matching the detected gesture with one or more predefined gestures, performs the touch action corresponding to the sensed touch input based on a determination that the detected gesture is valid and a determination that the detected shape is valid.

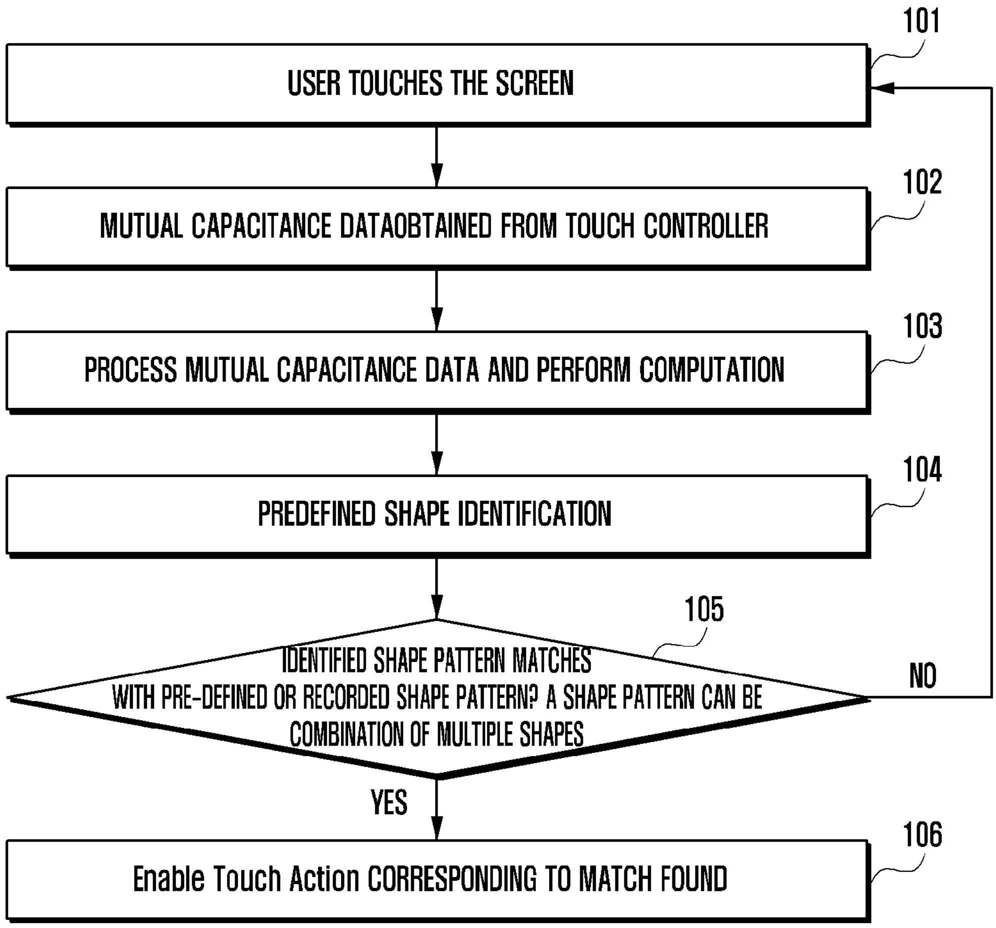
[Claim 27] The touch sensitive device of claim 26, wherein the processor computes a centroid of a touch region associated with the contact area, stores the centroid if a distance between the computed centroid and a previously stored centroid is above a certain predefined threshold, determines an end of the detected gesture, discards the detected gesture if the detected gesture has not ended within a certain predefined time period.

[Claim 28] The touch sensitive device of claim 27, wherein the processor determines the centroid of the touch region based on one or more node values, if an algorithm is configured to use mutual capacitance data, determines the centroid of the touch region by using touch coordinates obtained from a touch sensor panel integrated circuit (TSP IC), if the algorithm is not configured to the use mutual capacitance data.

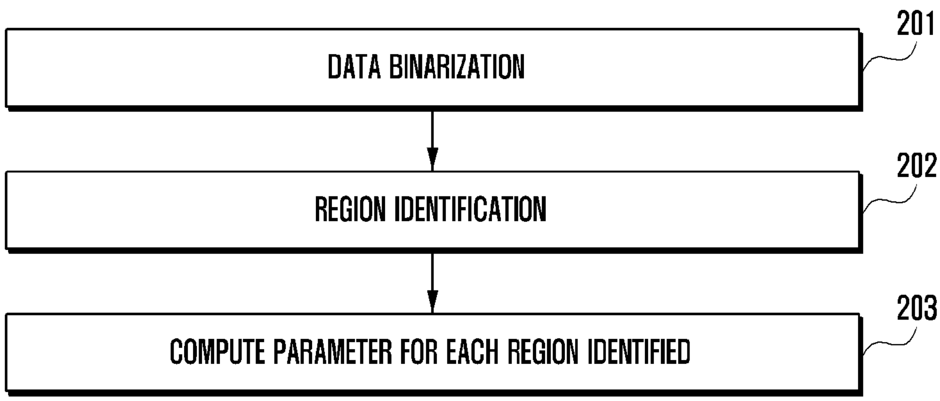
[Claim 29] The touch sensitive device of claim 27, wherein the processor determines that a number of nodes above a predefined threshold in a mutual capacitance data frame received from a touch sensor panel integrated circuit (TSP IC) do not lie within a predefined threshold range, if the touch sensitive device is configured to use mutual capacitance data, determines that an event corresponding to a pointer-up event is received from the TSP IC, if the touch sensitive device is configured to use touch coordinate data, checks that another stroke has not started within a predefined time interval, otherwise making a current stroke a part of the detected gesture.

[Claim 30] The touch sensitive device of claim 22, wherein the processor matches the detected gesture with one or more predefined gestures corresponding to the identified shape, if the touch sensitive device is configured to perform shape matching before gesture matching, matches the detected gesture with all registered or predefined gestures if the touch sensitive device is not configured to perform the shape matching before the gesture matching.

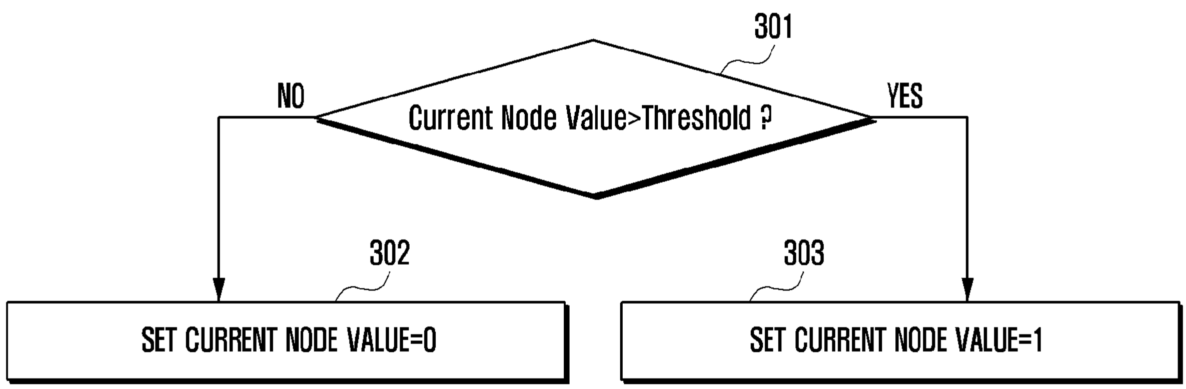
[Fig. 1]



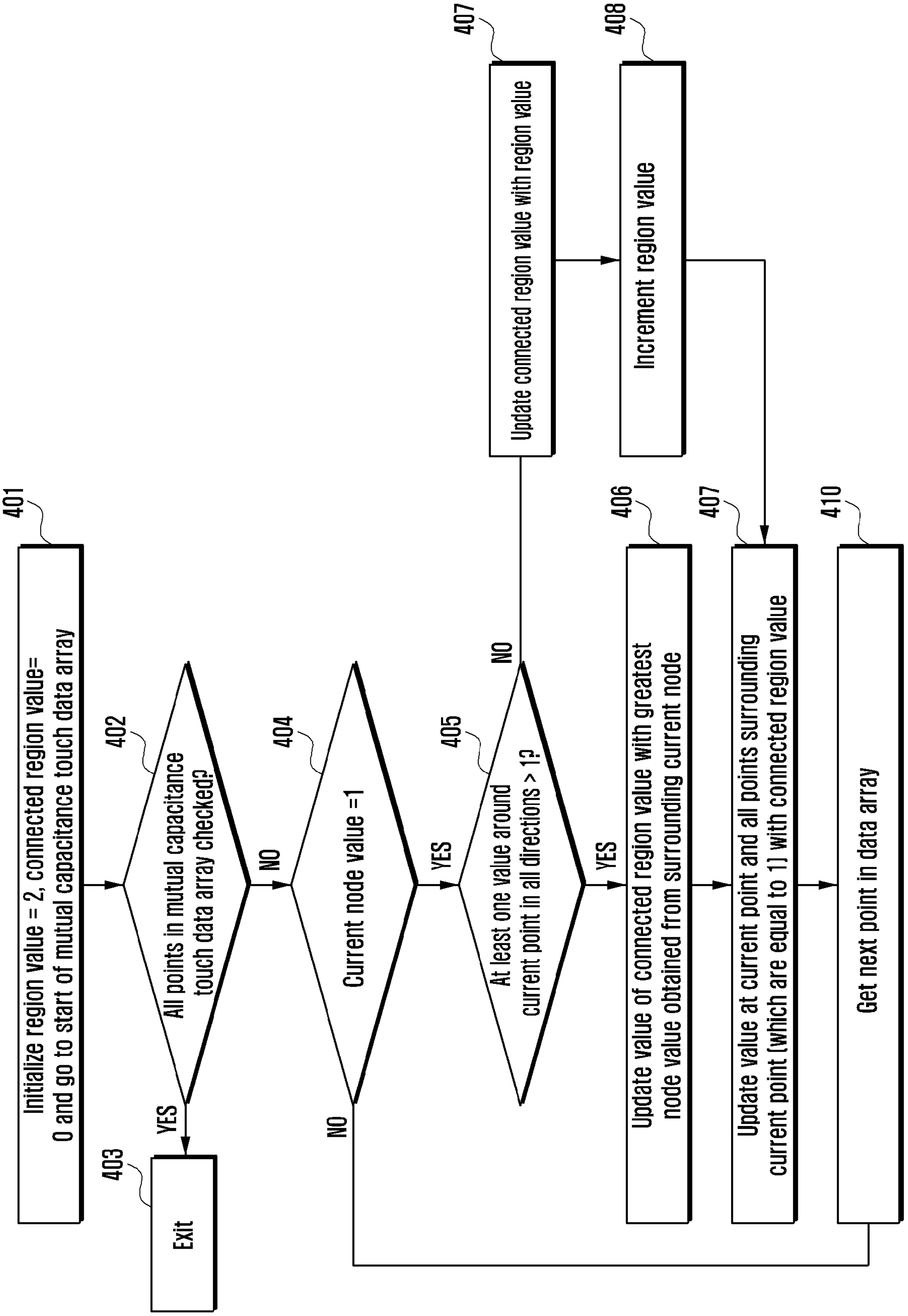
[Fig. 2]



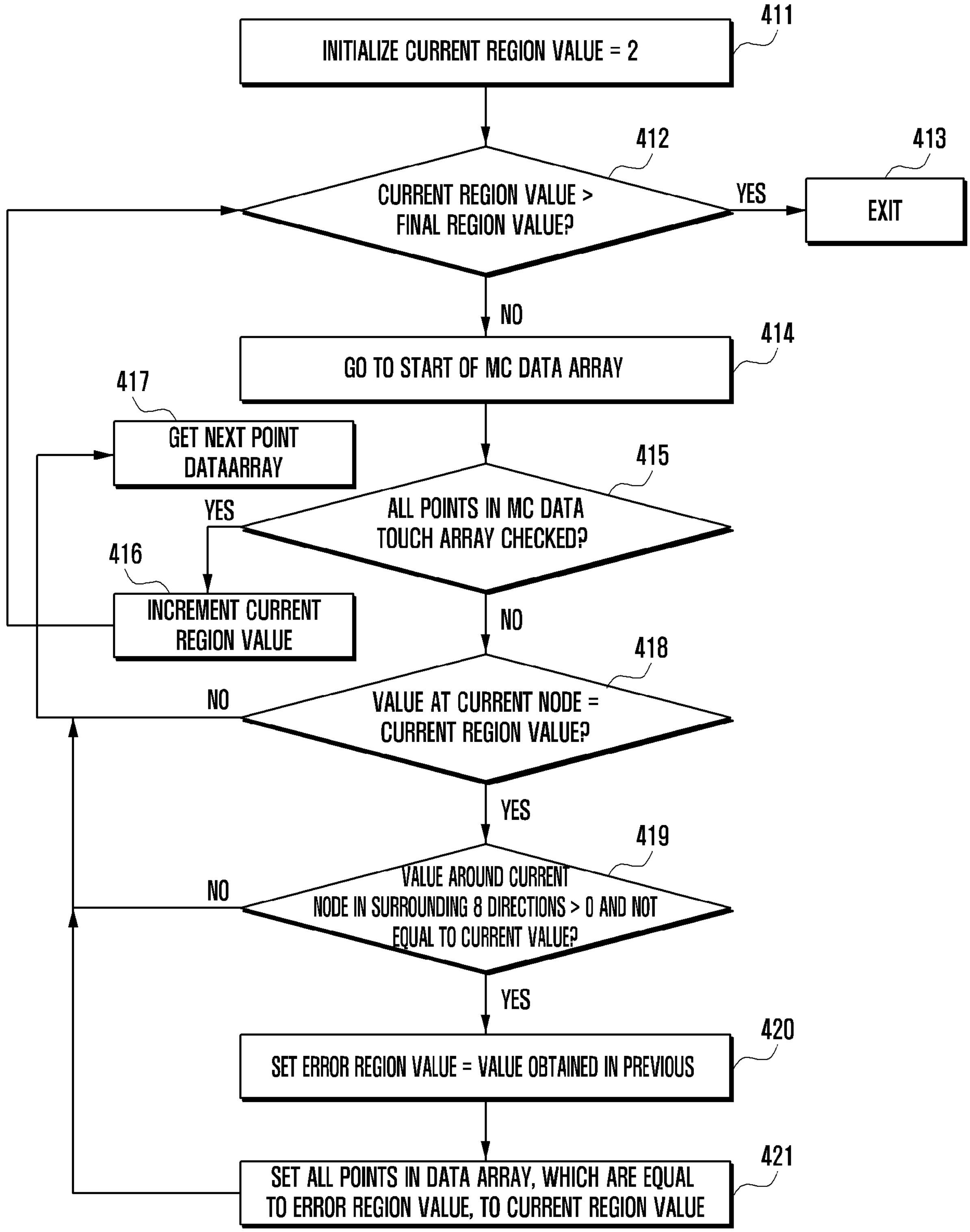
[Fig. 3]



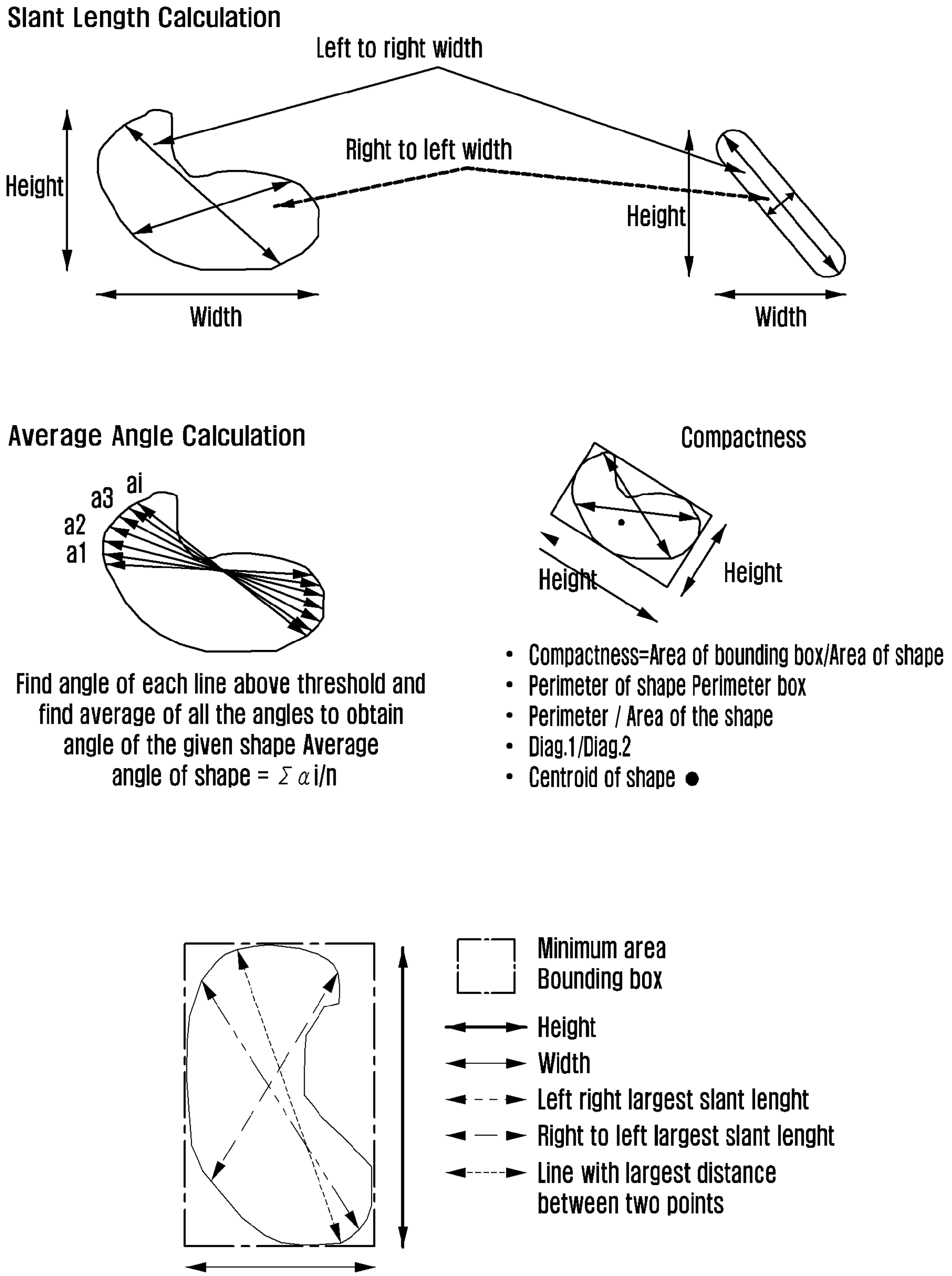
[Fig. 4a]


401
Initialize region value = 2, connected region value= 0 and go to start of mutual capacitance touch data array
402
All points in mutual capacitance touch data array checked?
YES
403
Exit
NO
404
Current node value =1
NO
YES
405
At least one value around current point in all directions > 1?
NO
407
Update connected region value with region value
408
Increment region value
YES
406
Update value of connected region value with greatest node value obtained from surrounding current node
407
Update value at current point and all points surrounding current point (which are equal to 1) with connected region value
410
Get next point in data array

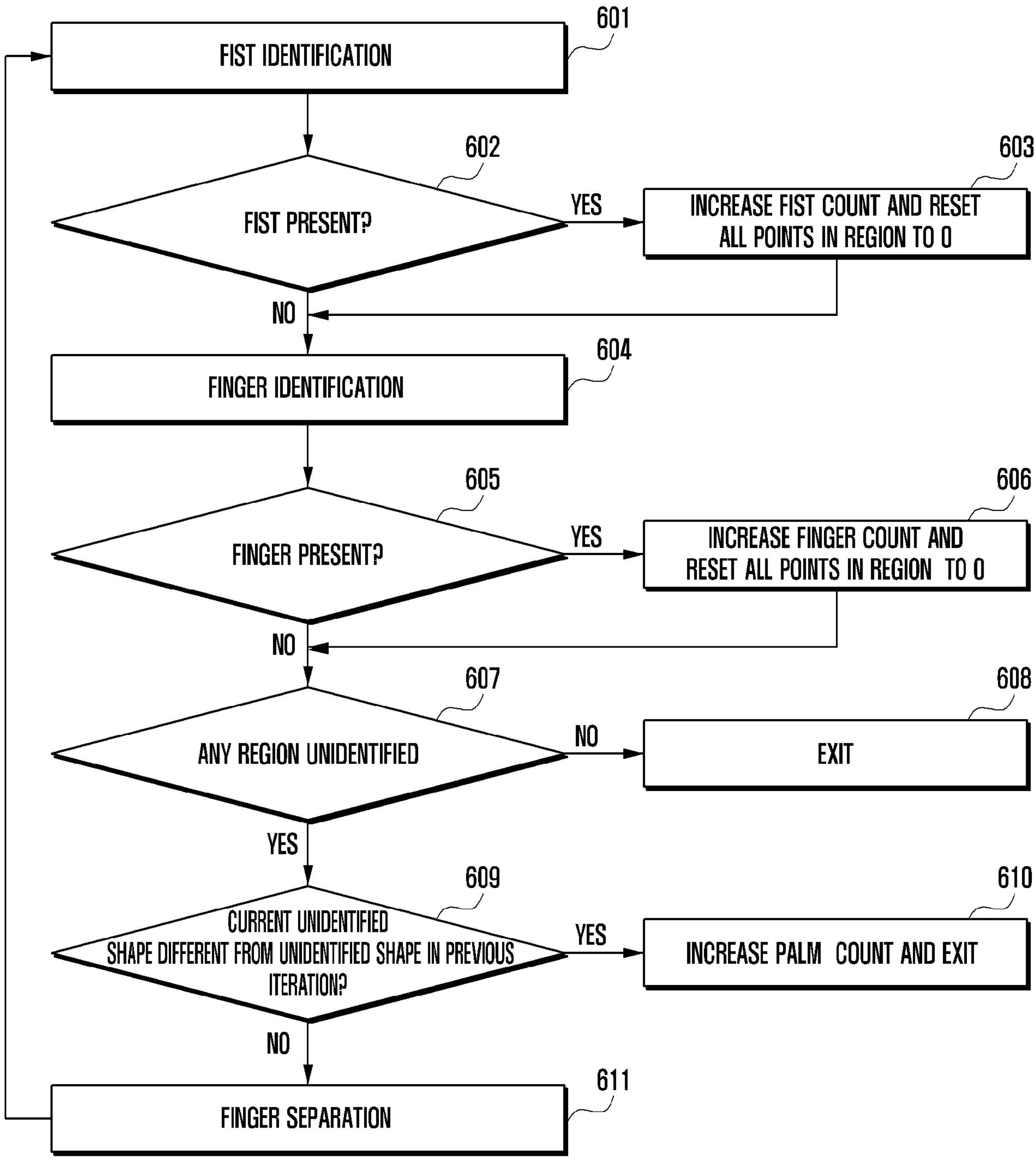
[Fig. 4b]


411
INITIALIZE CURRENT REGION VALUE = 2
412
CURRENT REGION VALUE > FINAL REGION VALUE?
YES
413
EXIT
NO
414
GO TO START OF MC DATA ARRAY
417
GET NEXT POINT DATAARRAY
415
ALL POINTS IN MC DATA TOUCH ARRAY CHECKED?
YES
416
INCREMENT CURRENT REGION VALUE
NO
418
VALUE AT CURRENT NODE = CURRENT REGION VALUE?
NO
YES
419
VALUE AROUND CURRENT NODE IN SURROUNDING 8 DIRECTIONS > 0 AND NOT EQUAL TO CURRENT VALUE?
NO
YES
420
SET ERROR REGION VALUE = VALUE OBTAINED IN PREVIOUS
421
SET ALL POINTS IN DATA ARRAY, WHICH ARE EQUAL TO ERROR REGION VALUE, TO CURRENT REGION VALUE

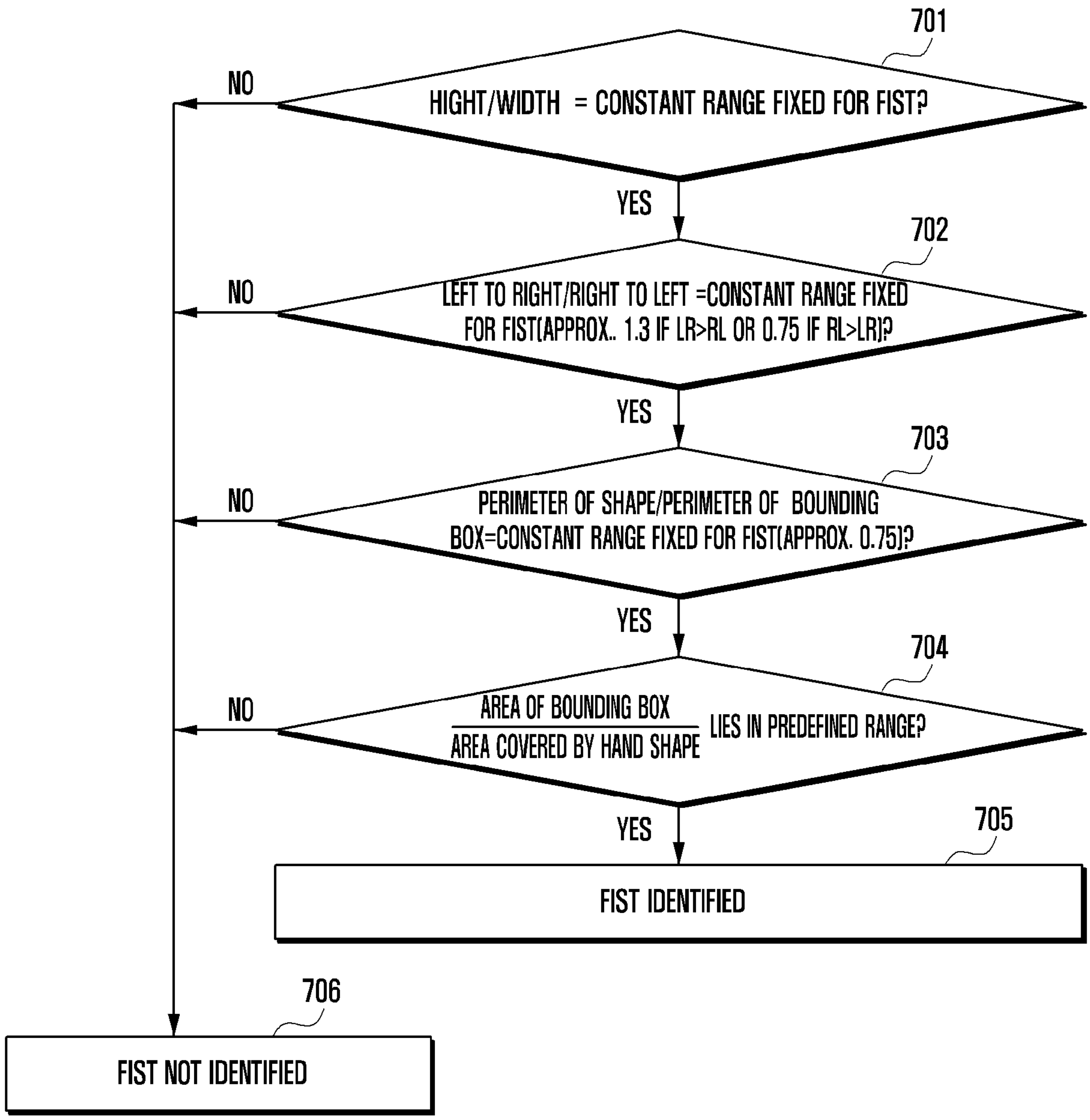
[Fig. 5]


Slant Length Calculation
Left to right width
Right to left width
Height
Width
Height
Width
Average Angle Calculation
ai
a3
a2
a1
Find angle of each line above threshold and find average of all the angles to obtain angle of the given shape Average angle of shape = Σ αi/n
Compactness
Height
Height
• Compactness=Area of bounding box/Area of shape
• Perimeter of shape Perimeter box
• Perimeter / Area of the shape
• Diag.1/Diag.2
• Centroid of shape ●
Minimum area
Bounding box
Height
Width
Left right largest slant lenght
Right to left largest slant lenght
Line with largest distance
between two points

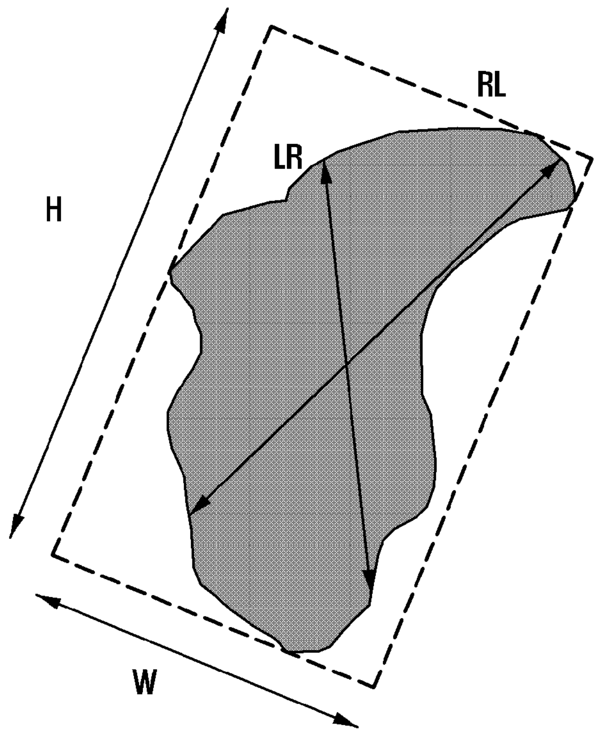
[Fig. 6]


601
FIST IDENTIFICATION
602
FIST PRESENT?
YES
603
INCREASE FIST COUNT AND RESET ALL POINTS IN REGION TO 0
NO
604
FINGER IDENTIFICATION
605
FINGER PRESENT?
YES
606
INCREASE FINGER COUNT AND RESET ALL POINTS IN REGION TO 0
NO
607
ANY REGION UNIDENTIFIED
NO
608
EXIT
YES
609
CURRENT UNIDENTIFIED SHAPE DIFFERENT FROM UNIDENTIFIED SHAPE IN PREVIOUS ITERATION?
YES
610
INCREASE PALM COUNT AND EXIT
NO
611
FINGER SEPARATION

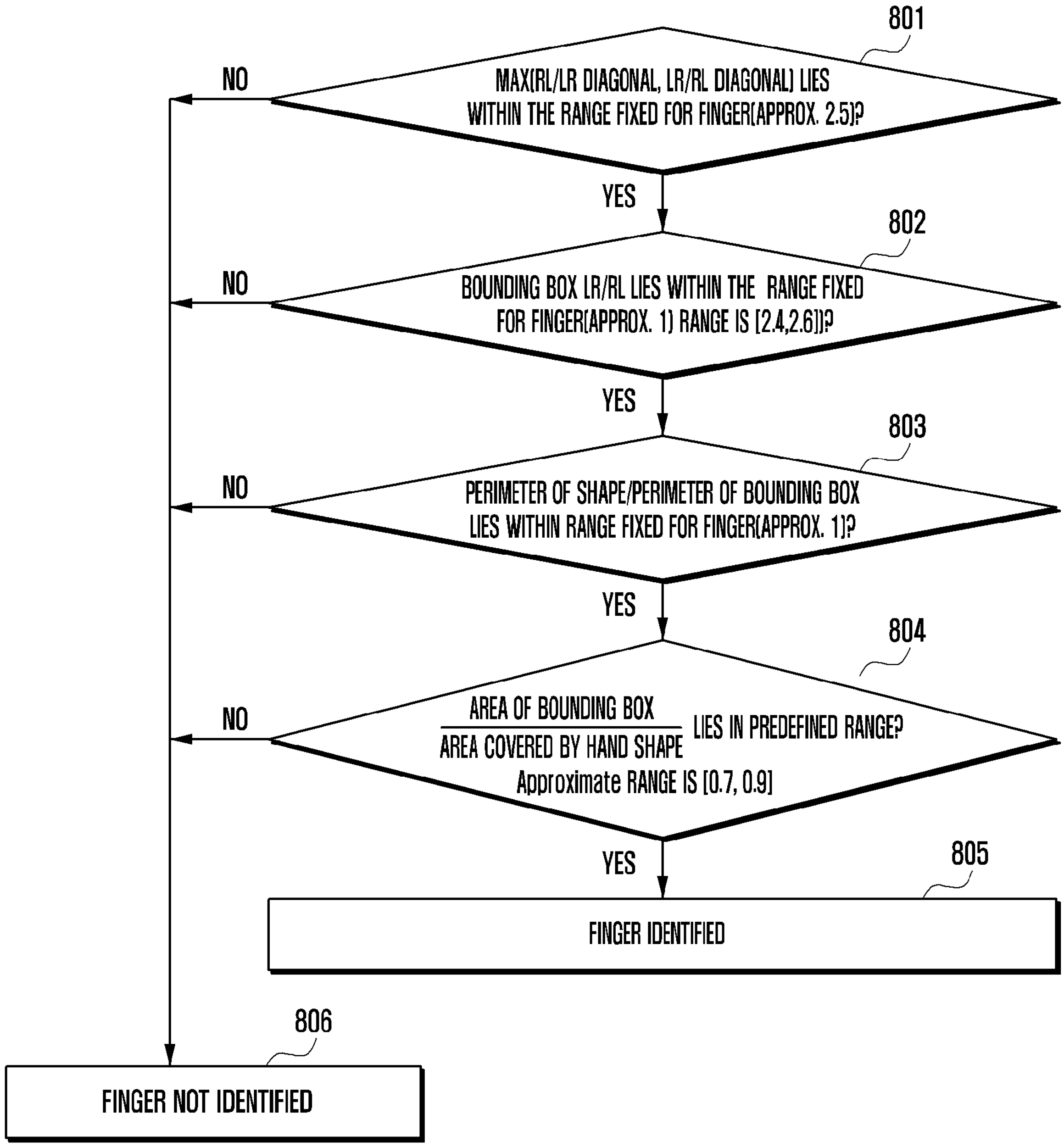
[Fig. 7a]


701
HIGHT/WIDTH = CONSTANT RANGE FIXED FOR FIST?
NO
YES
702
LEFT TO RIGHT/RIGHT TO LEFT =CONSTANT RANGE FIXED FOR FIST(APPROX.. 1.3 IF LR>RL OR 0.75 IF RL>LR)?
NO
YES
703
PERIMETER OF SHAPE/PERIMETER OF BOUNDING BOX=CONSTANT RANGE FIXED FOR FIST(APPROX. 0.75)?
NO
YES
704
AREA OF BOUNDING BOX / AREA COVERED BY HAND SHAPE LIES IN PREDEFINED RANGE?
NO
YES
705
FIST IDENTIFIED
706
FIST NOT IDENTIFIED

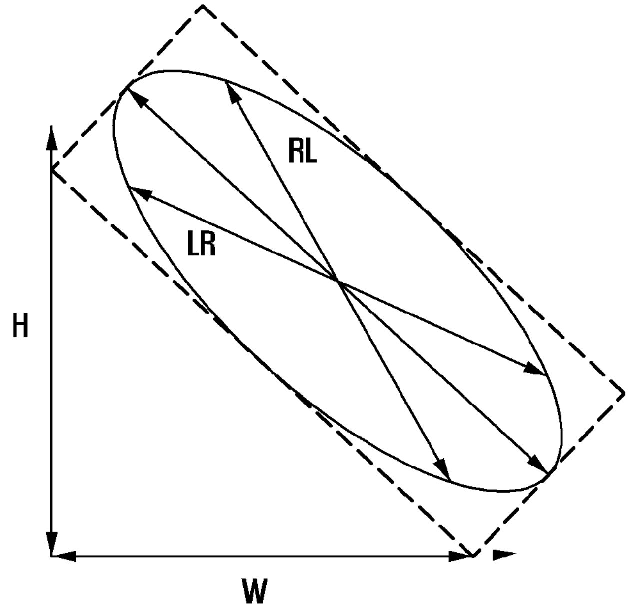
[Fig. 7b]



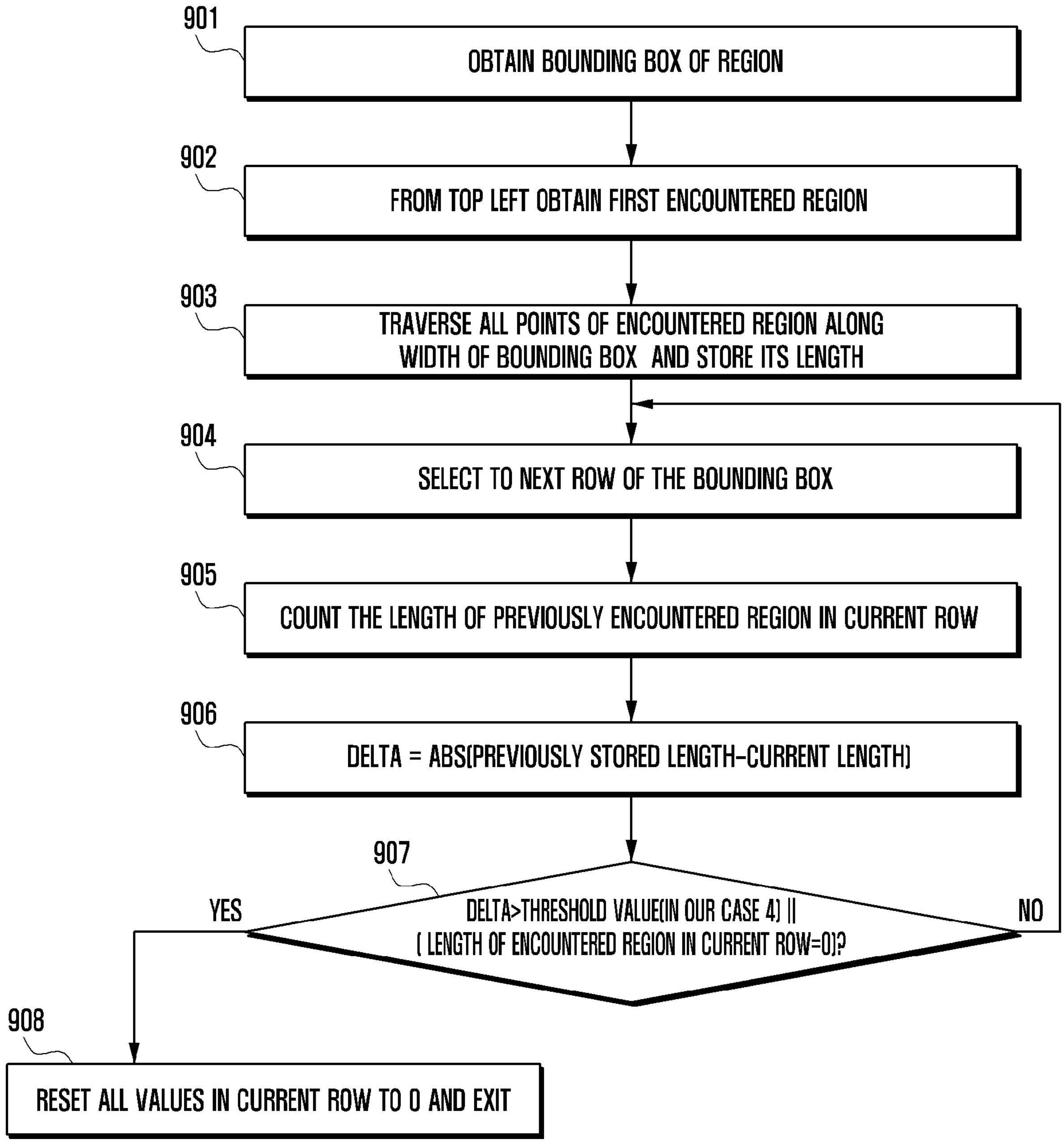
[Fig. 8a]


801
MAX(RL/LR DIAGONAL, LR/RL DIAGONAL) LIES WITHIN THE RANGE FIXED FOR FINGER(APPROX. 2.5)?
NO
YES
802
BOUNDING BOX LR/RL LIES WITHIN THE RANGE FIXED FOR FINGER(APPROX. 1) RANGE IS [2.4,2.6])?
NO
YES
803
PERIMETER OF SHAPE/PERIMETER OF BOUNDING BOX LIES WITHIN RANGE FIXED FOR FINGER(APPROX. 1)?
NO
YES
804
AREA OF BOUNDING BOX / AREA COVERED BY HAND SHAPE LIES IN PREDEFINED RANGE? Approximate RANGE IS [0.7, 0.9]
NO
YES
805
FINGER IDENTIFIED
806
FINGER NOT IDENTIFIED

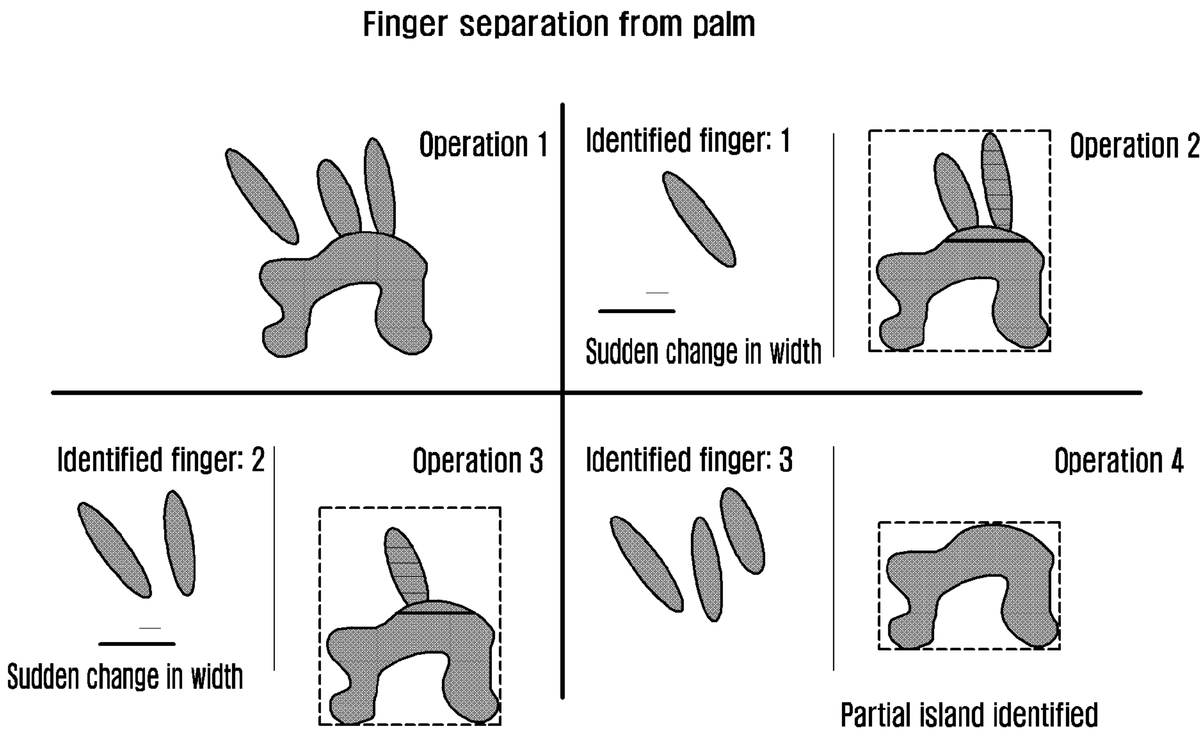
[Fig. 8b]



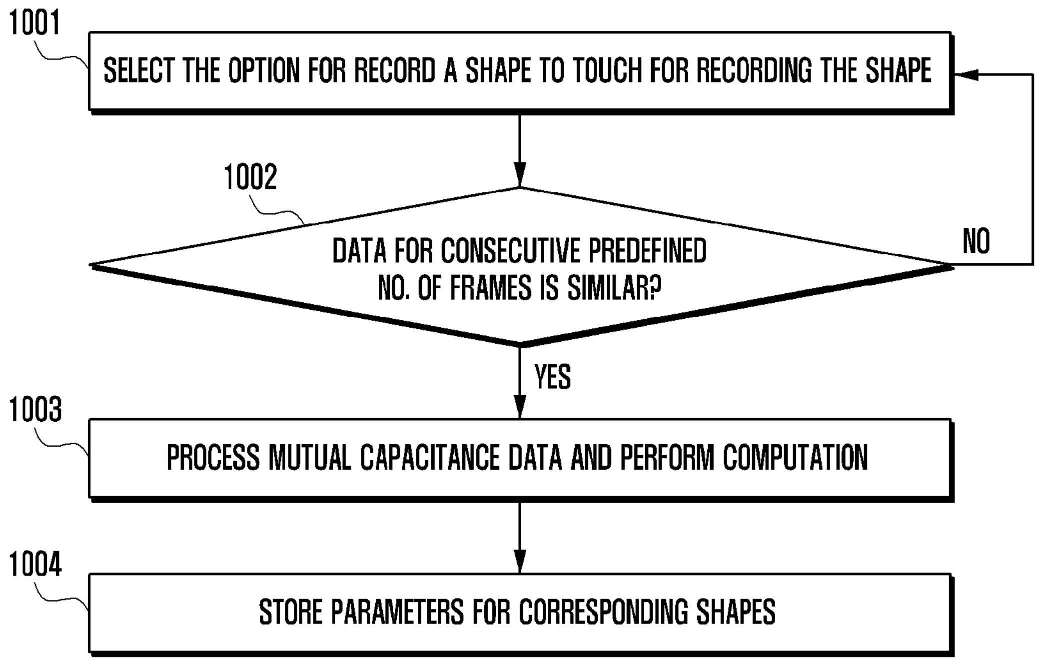
[Fig. 9a]


901
OBTAIN BOUNDING BOX OF REGION
902
FROM TOP LEFT OBTAIN FIRST ENCOUNTERED REGION
903
TRAVERSE ALL POINTS OF ENCOUNTERED REGION ALONG WIDTH OF BOUNDING BOX AND STORE ITS LENGTH
904
SELECT TO NEXT ROW OF THE BOUNDING BOX
905
COUNT THE LENGTH OF PREVIOUSLY ENCOUNTERED REGION IN CURRENT ROW
906
DELTA = ABS(PREVIOUSLY STORED LENGTH-CURRENT LENGTH)
907
DELTA>THRESHOLD VALUE(IN OUR CASE 4) ||
(LENGTH OF ENCOUNTERED REGION IN CURRENT ROW=0)?
YES
NO
908
RESET ALL VALUES IN CURRENT ROW TO 0 AND EXIT

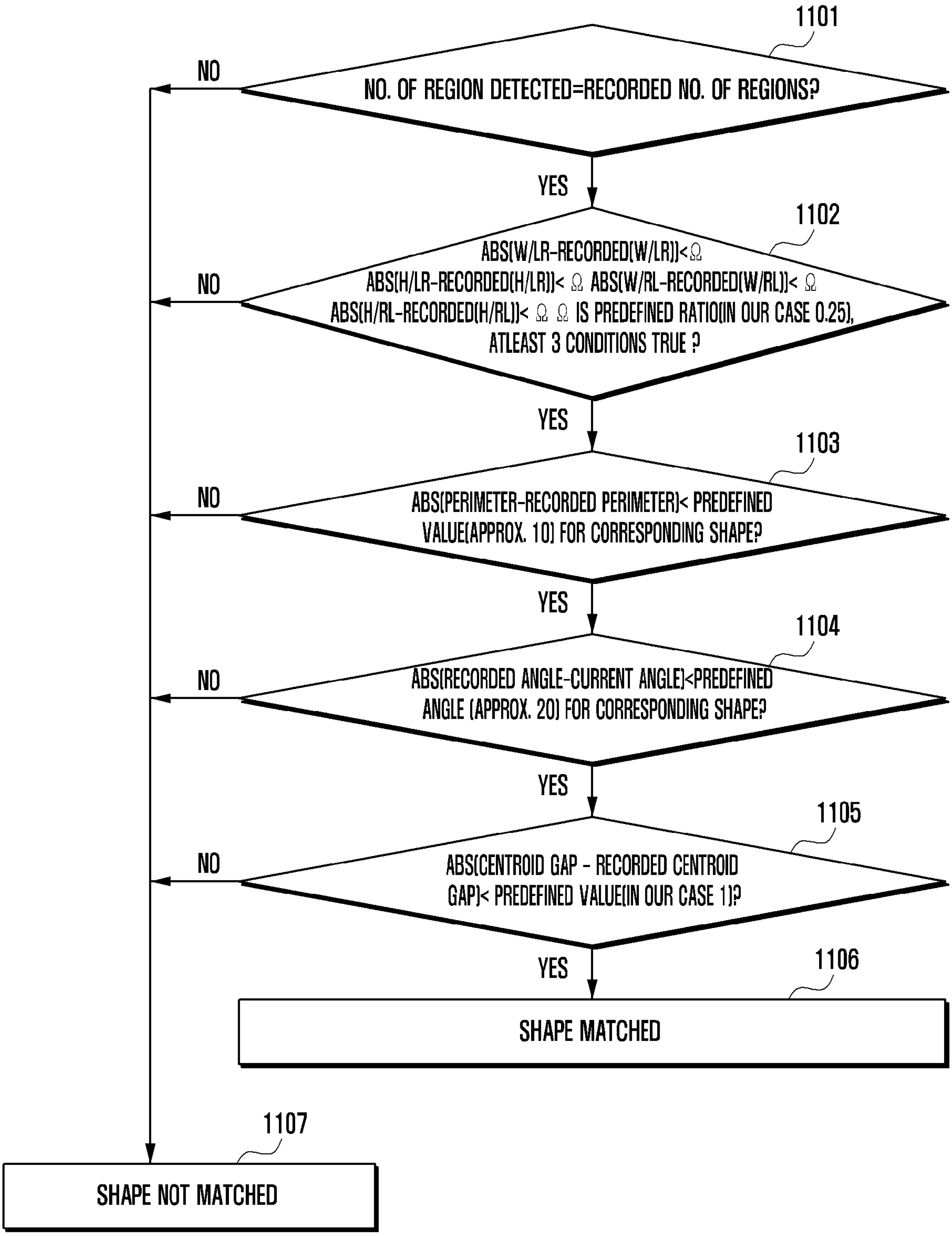
[Fig. 9b]


Finger separation from palm
Operation 1
Identified finger: 1
Sudden change in width
Operation 2
Identified finger: 2
Sudden change in width
Operation 3
Identified finger: 3
Operation 4
Partial island identified

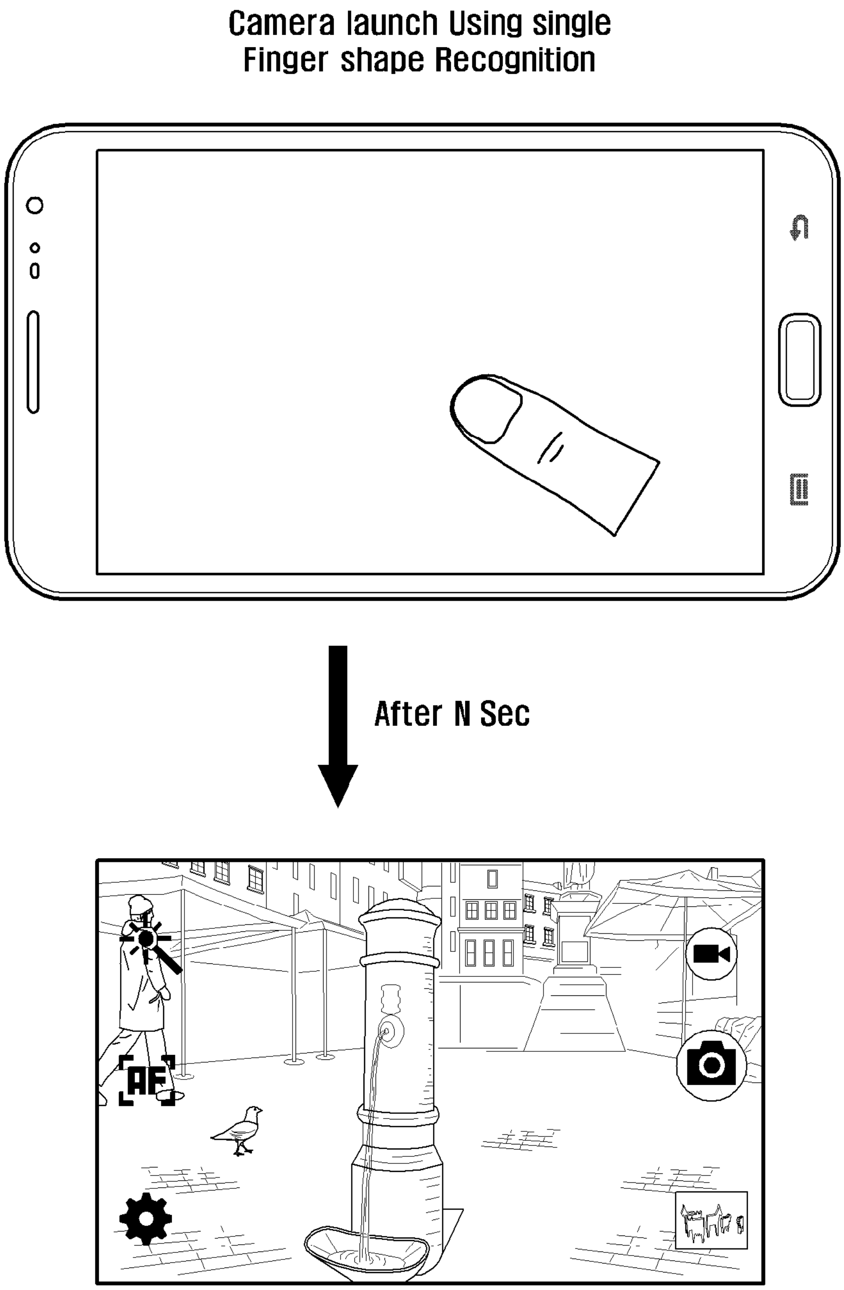
[Fig. 10]


1001
SELECT THE OPTION FOR RECORD A SHAPE TO TOUCH FOR RECORDING THE SHAPE
1002
DATA FOR CONSECUTIVE PREDEFINED
NO. OF FRAMES IS SIMILAR?
NO
YES
1003
PROCESS MUTUAL CAPACITANCE DATA AND PERFORM COMPUTATION
1004
STORE PARAMETERS FOR CORRESPONDING SHAPES

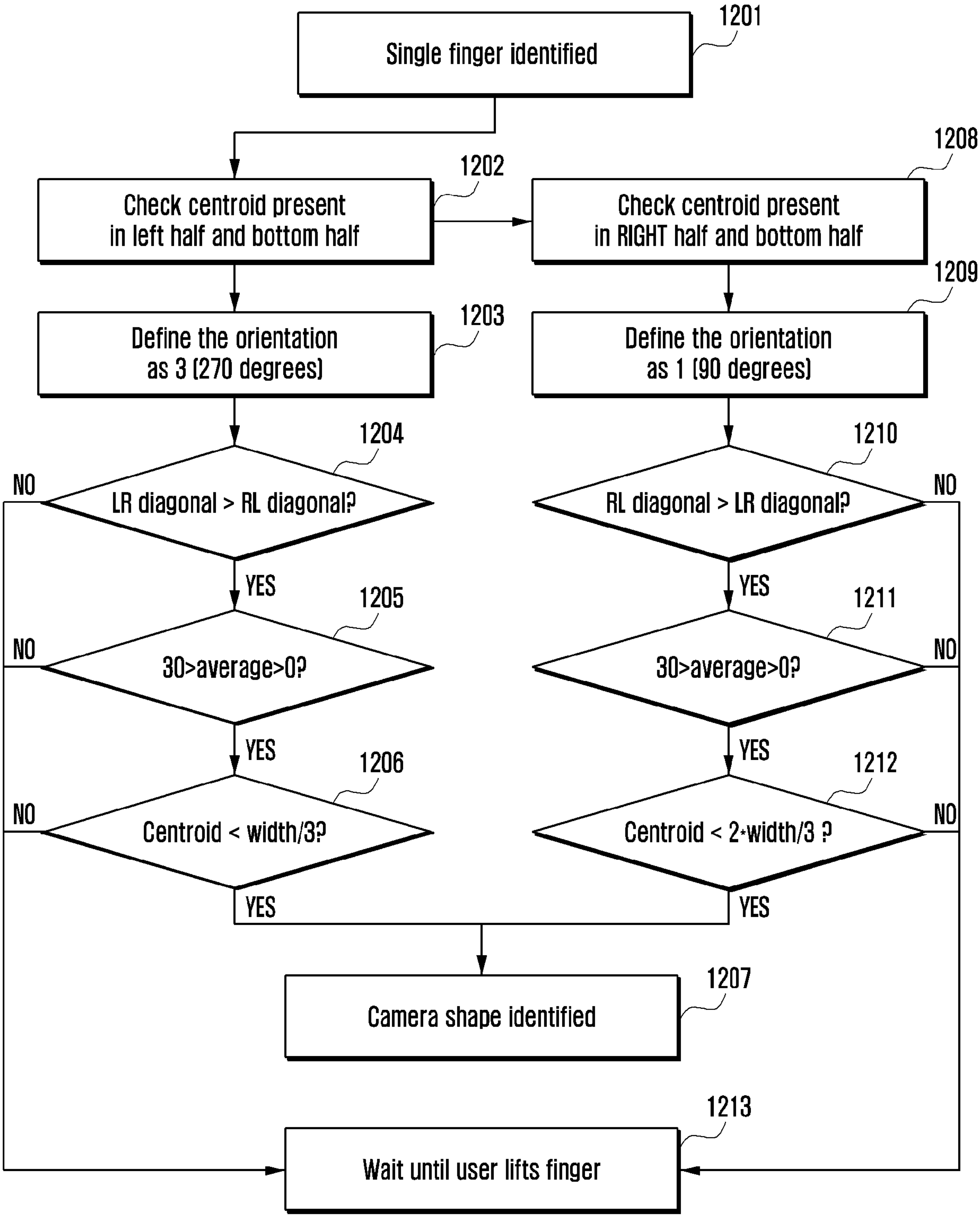
[Fig. 11]


1101
NO. OF REGION DETECTED=RECORDED NO. OF REGIONS?
NO
YES
1102
ABS(W/LR-RECORDED(W/LR))<Ω
ABS(H/LR-RECORDED(H/LR))< Ω ABS(W/RL-RECORDED(W/RL))< Ω
ABS(H/RL-RECORDED(H/RL))< Ω Ω IS PREDEFINED RATIO(IN OUR CASE 0.25),
ATLEAST 3 CONDITIONS TRUE ?
NO
YES
1103
ABS(PERIMETER-RECORDED PERIMETER)< PREDEFINED
VALUE(APPROX. 10) FOR CORRESPONDING SHAPE?
NO
YES
1104
ABS(RECORDED ANGLE-CURRENT ANGLE)<PREDEFINED
ANGLE (APPROX. 20) FOR CORRESPONDING SHAPE?
NO
YES
1105
ABS(CENTROID GAP - RECORDED CENTROID
GAP)< PREDEFINED VALUE(IN OUR CASE 1)?
NO
YES
1106
SHAPE MATCHED
1107
SHAPE NOT MATCHED

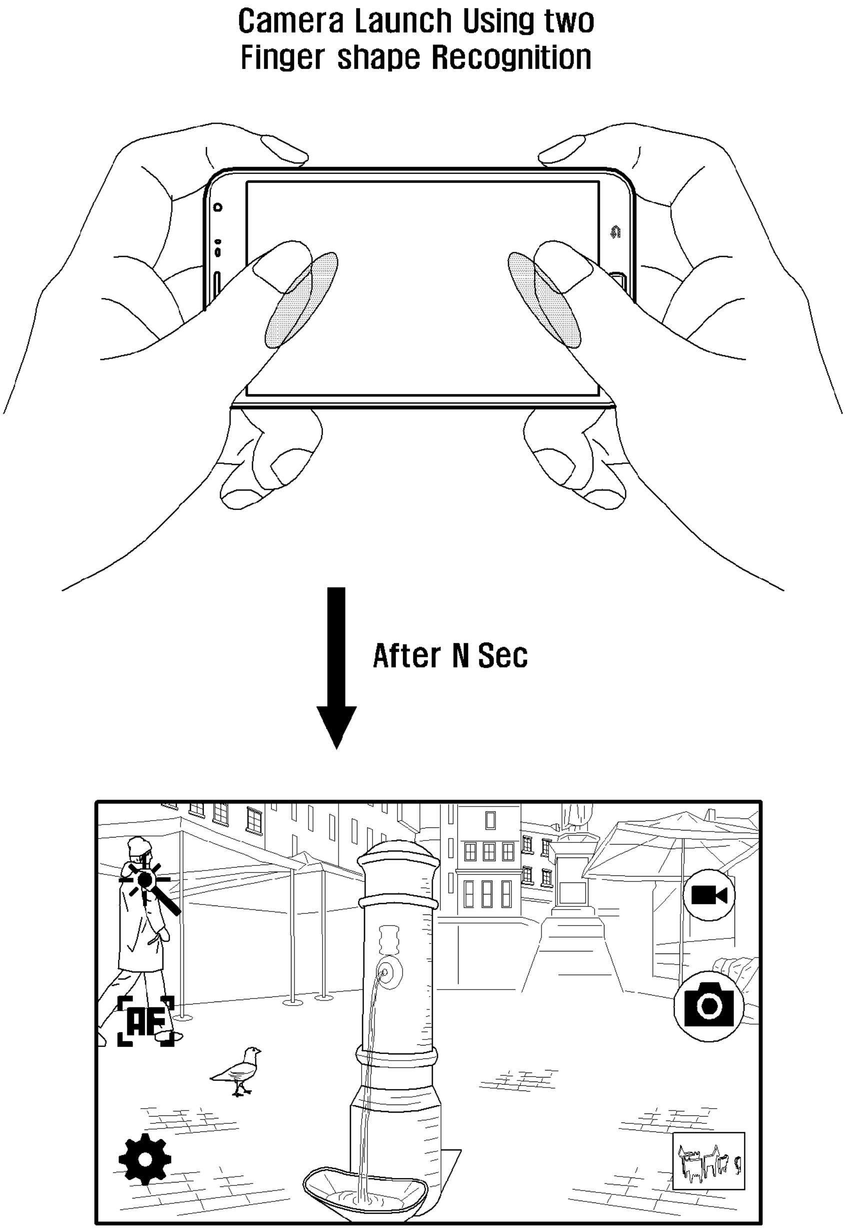
[Fig. 12a]


Camera launch Using single
Finger shape Recognition
After N Sec
AF

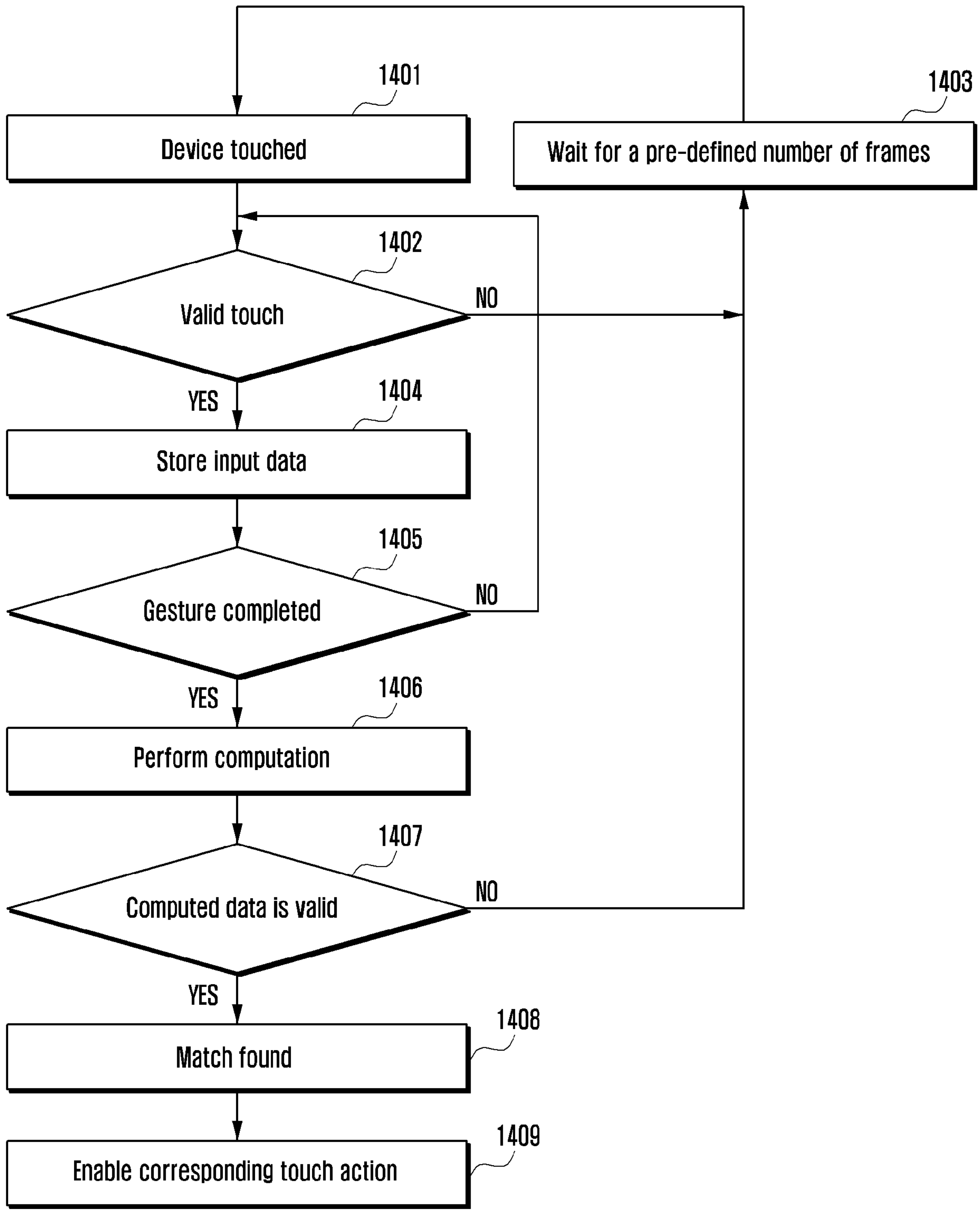
[Fig. 12b]


1201
Single finger identified
1202
Check centroid present in left half and bottom half
1208
Check centroid present in RIGHT half and bottom half
1203
Define the orientation as 3 (270 degrees)
1209
Define the orientation as 1 (90 degrees)
1204
LR diagonal > RL diagonal?
1210
RL diagonal > LR diagonal?
NO
YES
1205
30>average>0?
1211
30>average>0?
1206
Centroid < width/3?
1212
Centroid < 2*width/3 ?
1207
Camera shape identified
1213
Wait until user lifts finger

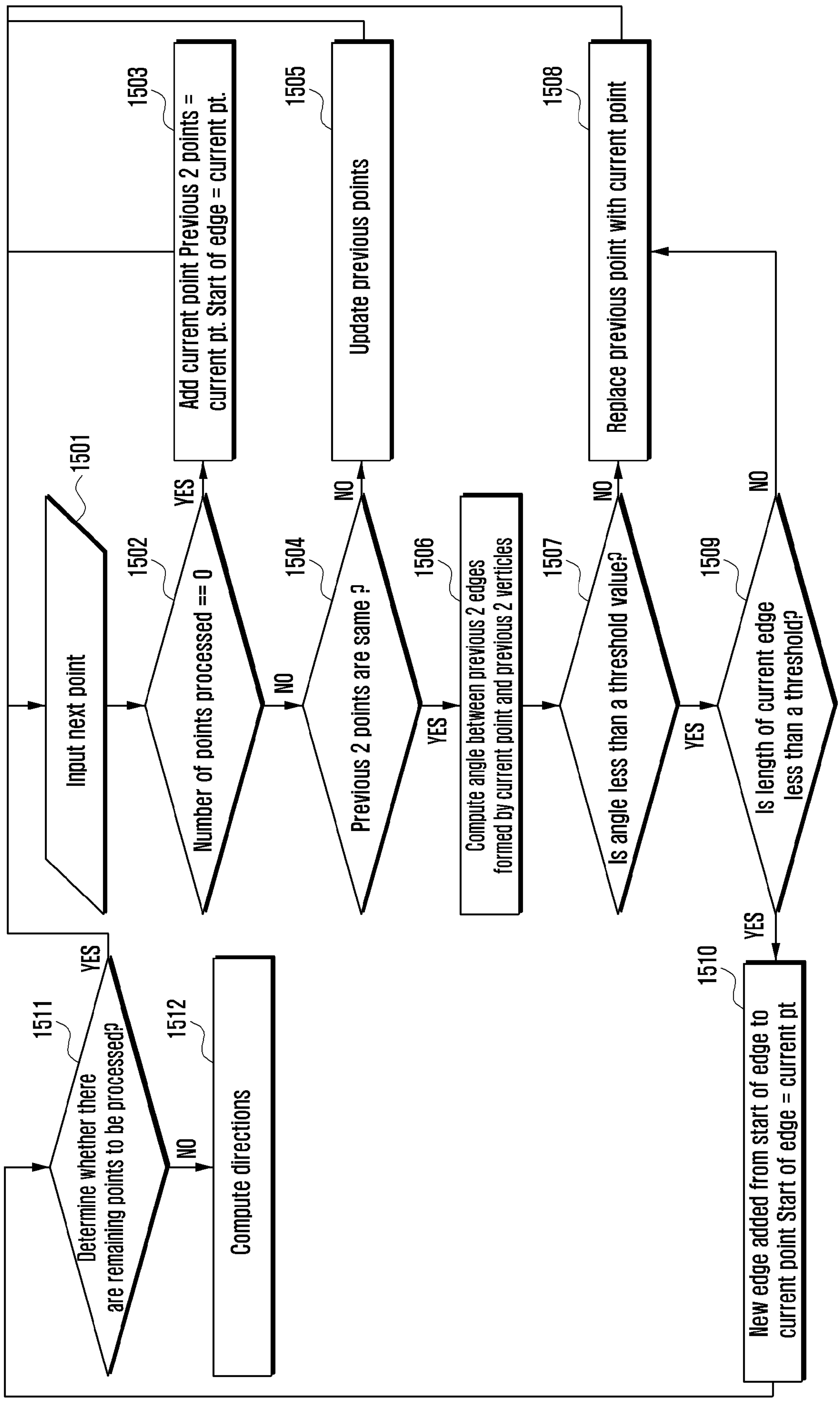
[Fig. 13]


Camera Launch Using two
Finger shape Recognition
After N Sec
AF

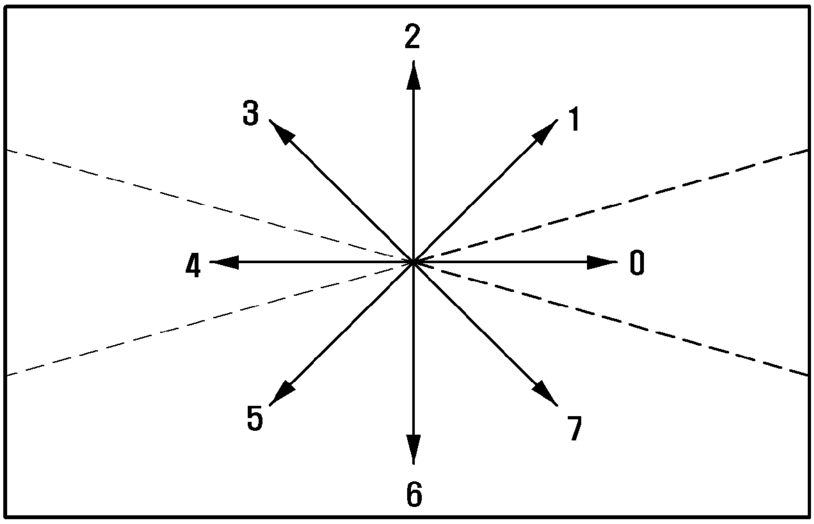
[Fig. 14]


1401
Device touched
1403
Wait for a pre-defined number of frames
1402
Valid touch
NO
YES
1404
Store input data
1405
Gesture completed
NO
YES
1406
Perform computation
1407
Computed data is valid
NO
YES
1408
Match found
1409
Enable corresponding touch action

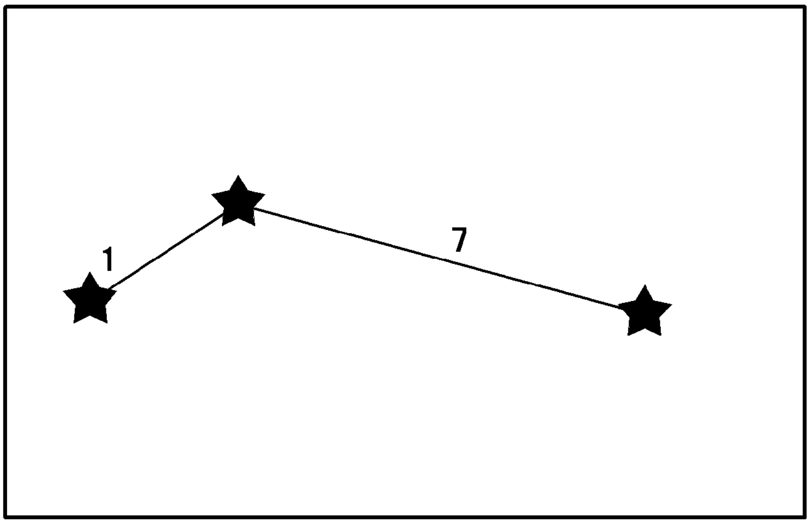
[Fig. 15]


1501
Input next point
1502
Number of points processed == 0
YES
1503
Add current point Previous 2 points = current pt. Start of edge = current pt.
NO
1504
Previous 2 points are same ?
NO
1505
Update previous points
YES
1506
Compute angle between previous 2 edges formed by current point and previous 2 verticles
1507
Is angle less than a threshold value?
NO
1508
Replace previous point with current point
YES
1509
Is length of current edge less than a threshold?
NO
YES
1510
New edge added from start of edge to current point Start of edge = current pt
1511
Determine whether there are remaining points to be processed?
YES
NO
1512
Compute directions

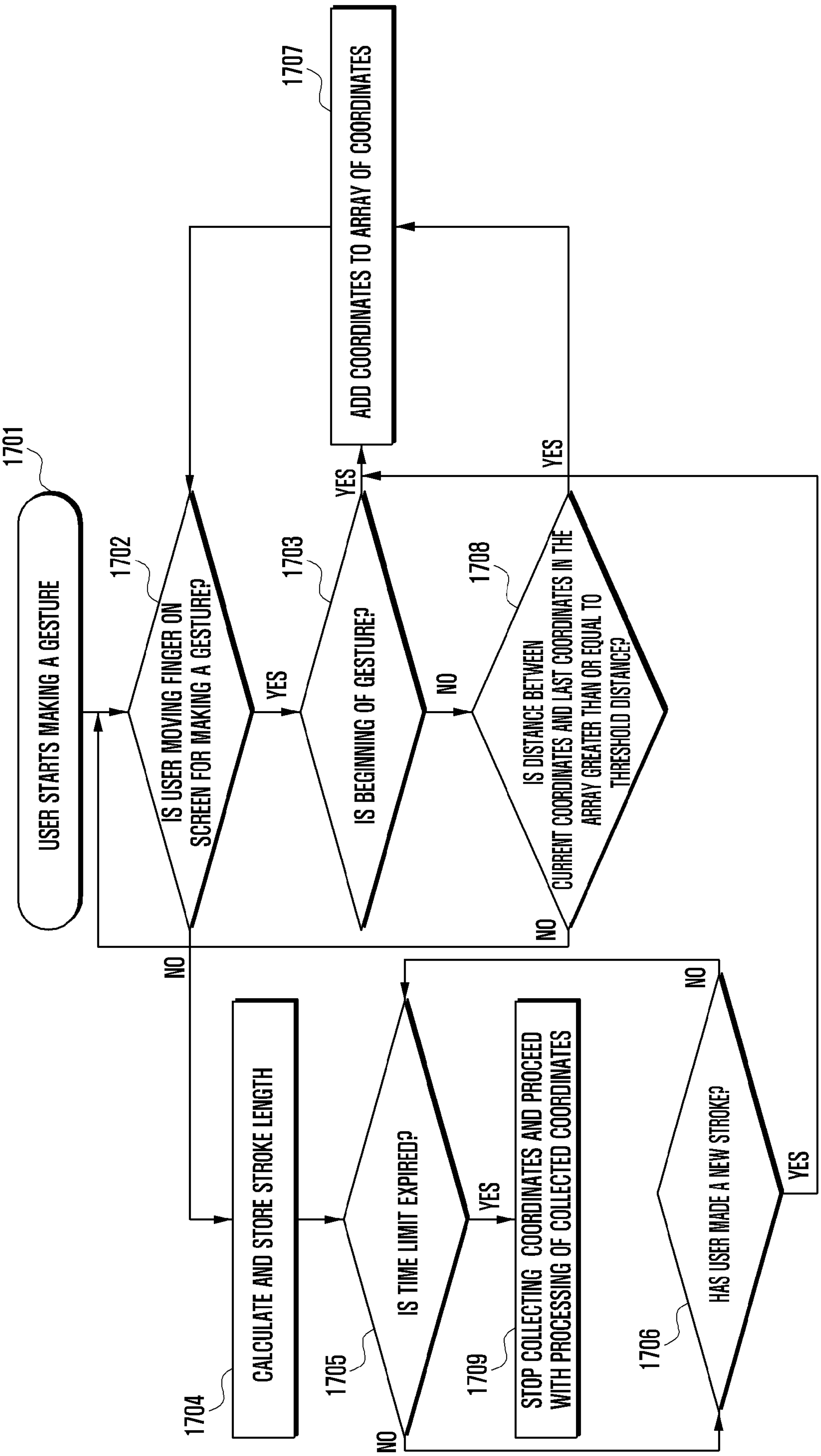
[Fig. 16a]



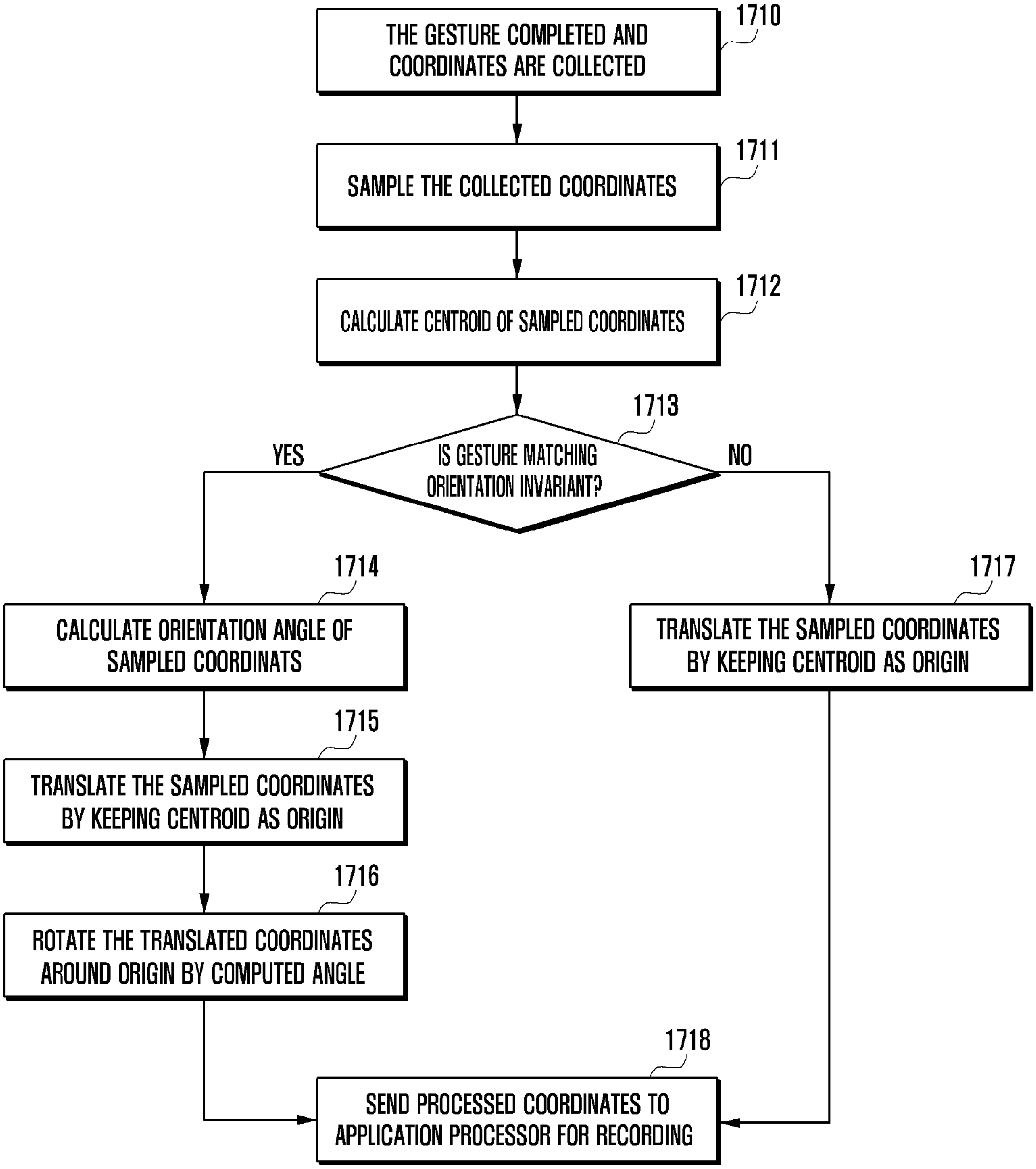
[Fig. 16b]



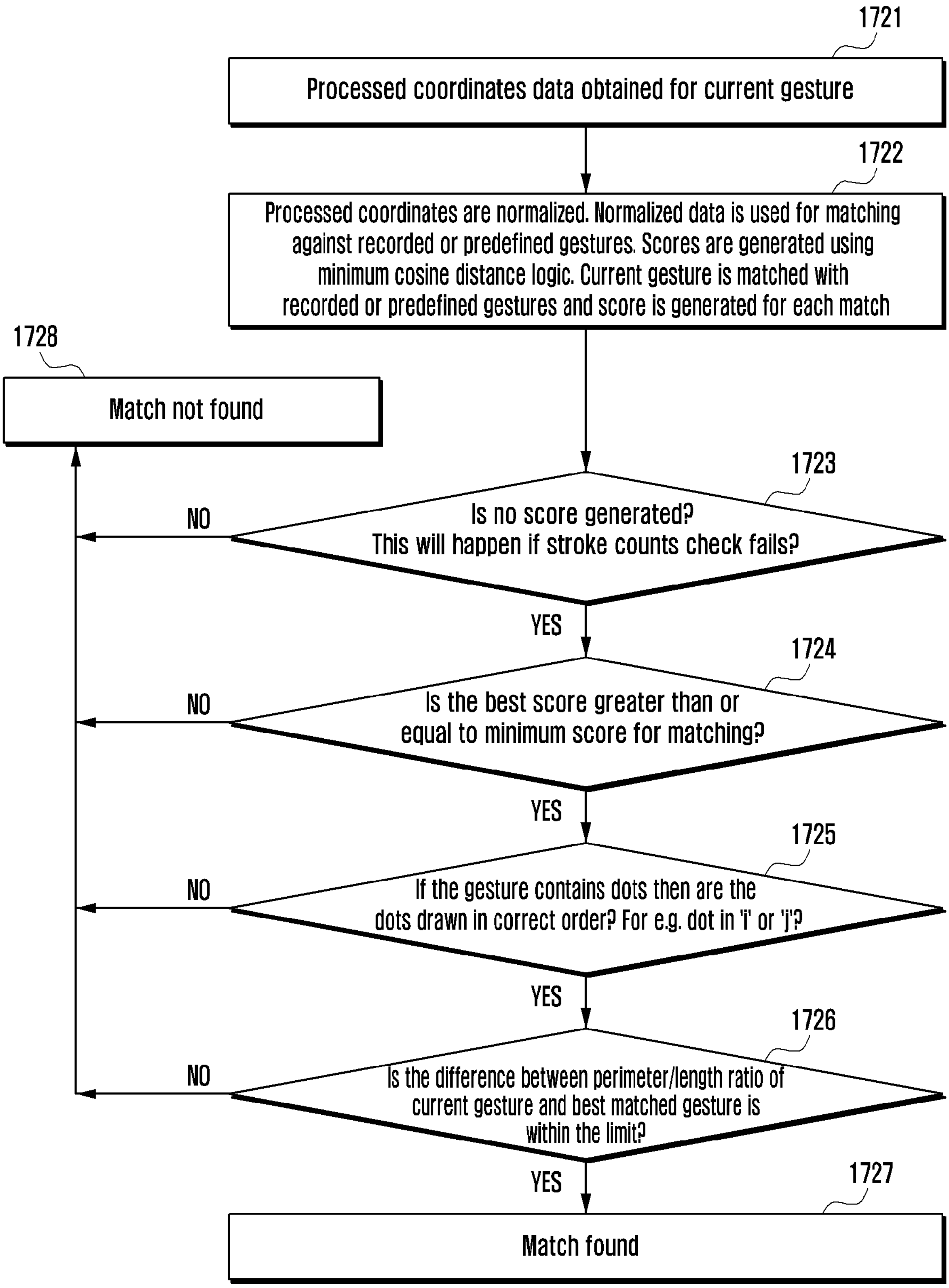
[Fig. 17a]


1701
USER STARTS MAKING A GESTURE
1702
IS USER MOVING FINGER ON SCREEN FOR MAKING A GESTURE?
YES
NO
1703
IS BEGINNING OF GESTURE?
YES
NO
1707
ADD COORDINATES TO ARRAY OF COORDINATES
1708
IS DISTANCE BETWEEN CURRENT COORDINATES AND LAST COORDINATES IN THE ARRAY GREATER THAN OR EQUAL TO THRESHOLD DISTANCE?
YES
NO
1704
CALCULATE AND STORE STROKE LENGTH
1705
IS TIME LIMIT EXPIRED?
YES
NO
1709
STOP COLLECTING COORDINATES AND PROCEED WITH PROCESSING OF COLLECTED COORDINATES
1706
HAS USER MADE A NEW STROKE?
YES
NO

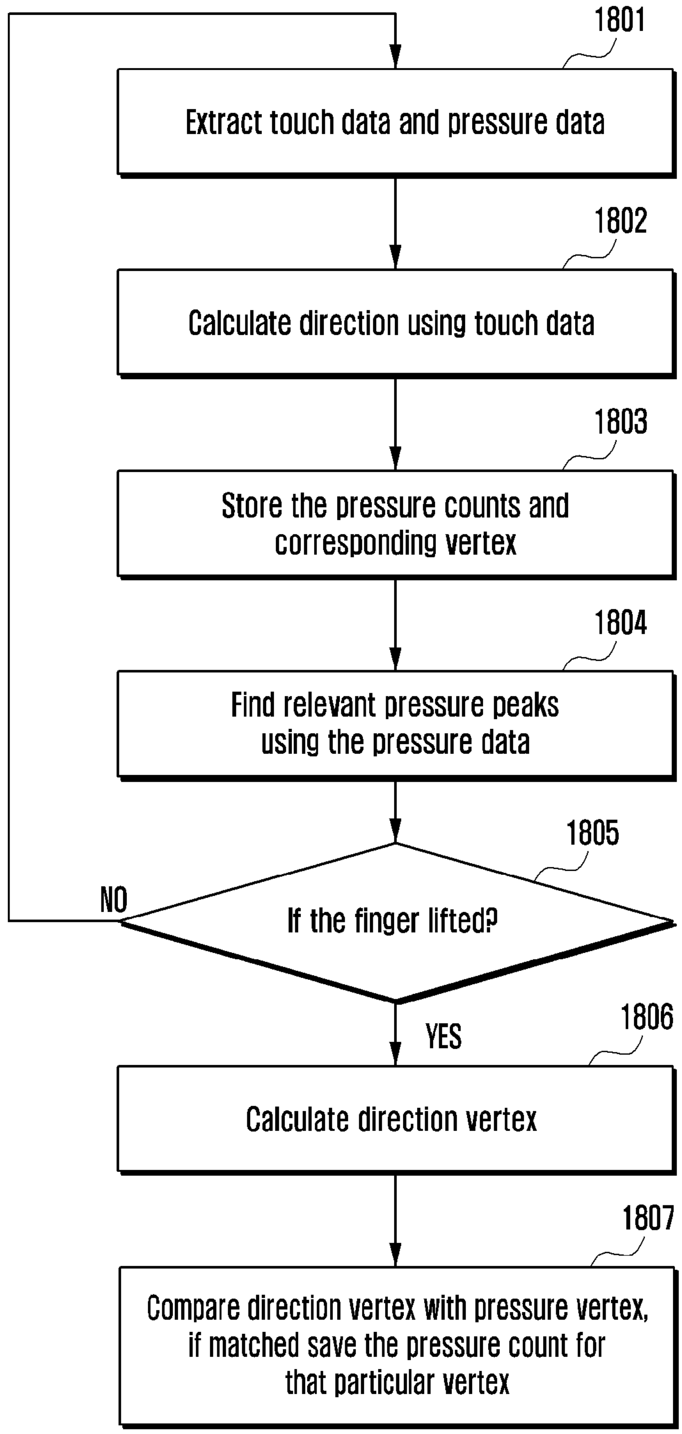
[Fig. 17b]


THE GESTURE COMPLETED AND COORDINATES ARE COLLECTED
1710
SAMPLE THE COLLECTED COORDINATES
1711
CALCULATE CENTROID OF SAMPLED COORDINATES
1712
IS GESTURE MATCHING ORIENTATION INVARIANT?
1713
YES
NO
CALCULATE ORIENTATION ANGLE OF SAMPLED COORDINATS
1714
TRANSLATE THE SAMPLED COORDINATES BY KEEPING CENTROID AS ORIGIN
1715
ROTATE THE TRANSLATED COORDINATES AROUND ORIGIN BY COMPUTED ANGLE
1716
TRANSLATE THE SAMPLED COORDINATES BY KEEPING CENTROID AS ORIGIN
1717
SEND PROCESSED COORDINATES TO APPLICATION PROCESSOR FOR RECORDING
1718

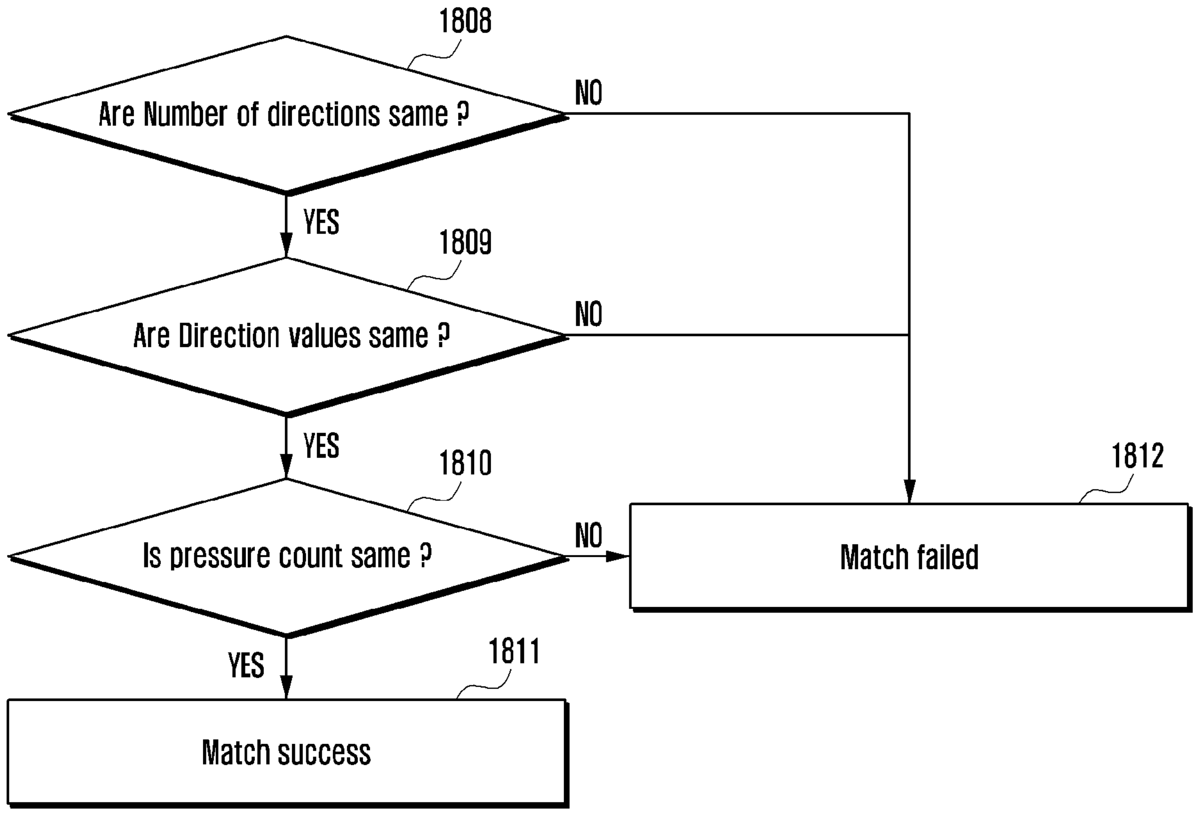
[Fig. 17c]


1721
Processed coordinates data obtained for current gesture
1722
Processed coordinates are normalized. Normalized data is used for matching against recorded or predefined gestures. Scores are generated using minimum cosine distance logic. Current gesture is matched with recorded or predefined gestures and score is generated for each match
1728
Match not found
1723
Is no score generated? This will happen if stroke counts check fails?
NO
YES
1724
Is the best score greater than or equal to minimum score for matching?
NO
YES
1725
If the gesture contains dots then are the dots drawn in correct order? For e.g. dot in 'i' or 'j'?
NO
YES
1726
Is the difference between perimeter/length ratio of current gesture and best matched gesture is within the limit?
NO
YES
1727
Match found

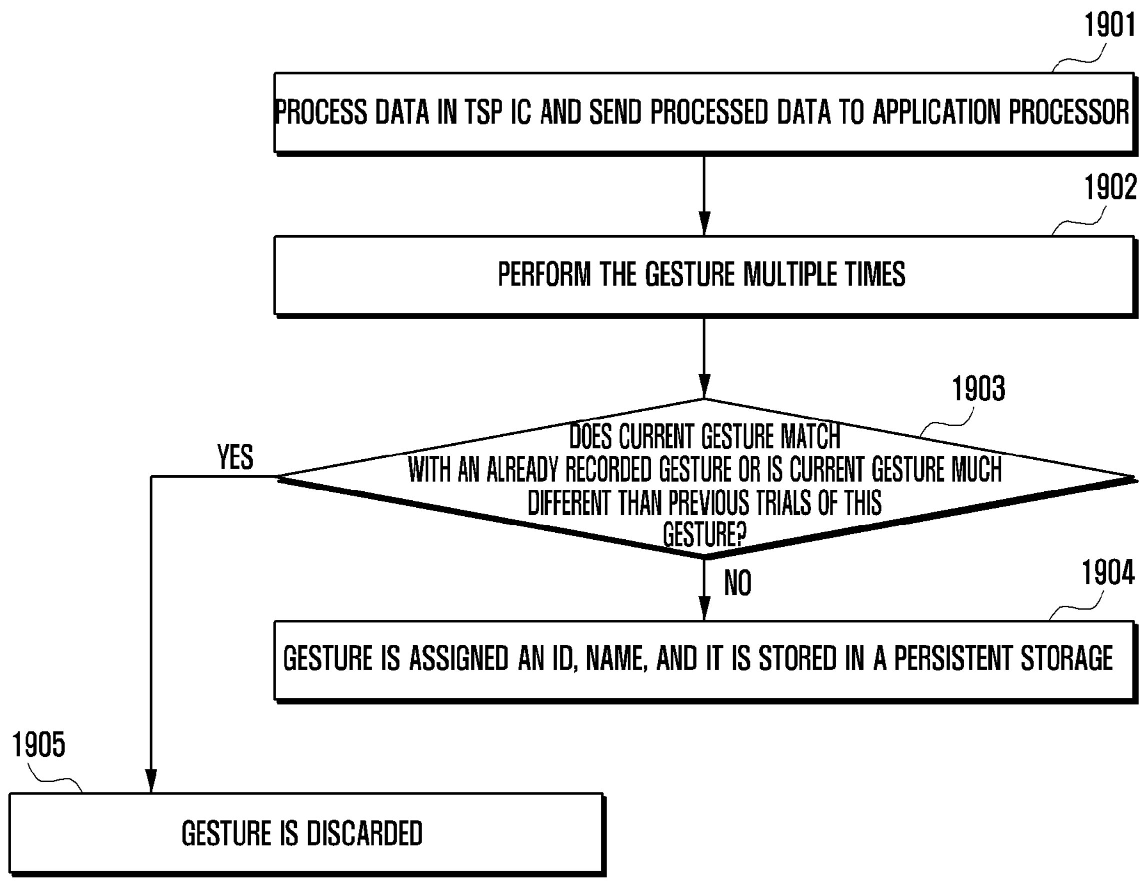
[Fig. 18a]


1801
Extract touch data and pressure data
1802
Calculate direction using touch data
1803
Store the pressure counts and corresponding vertex
1804
Find relevant pressure peaks using the pressure data
1805
If the finger lifted?
NO
YES
1806
Calculate direction vertex
1807
Compare direction vertex with pressure vertex, if matched save the pressure count for that particular vertex

[Fig. 18b]

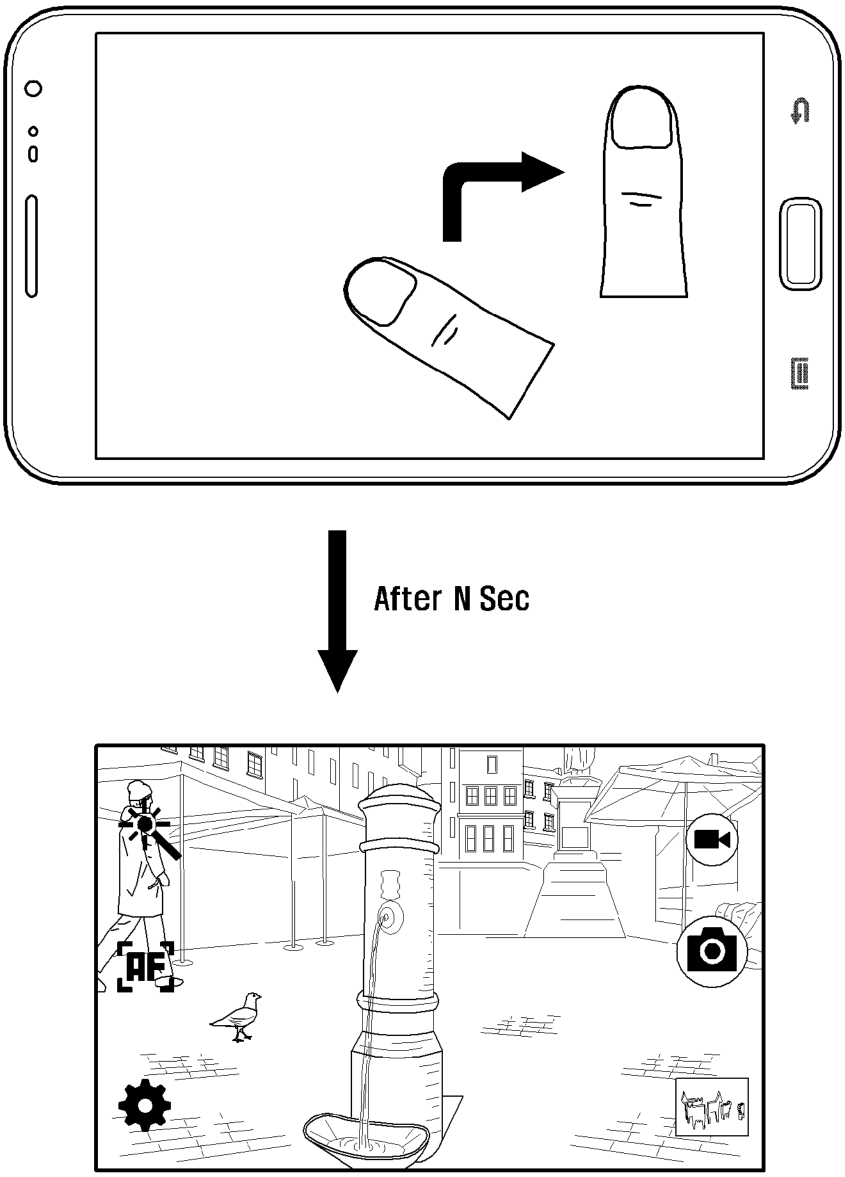


[Fig. 19]

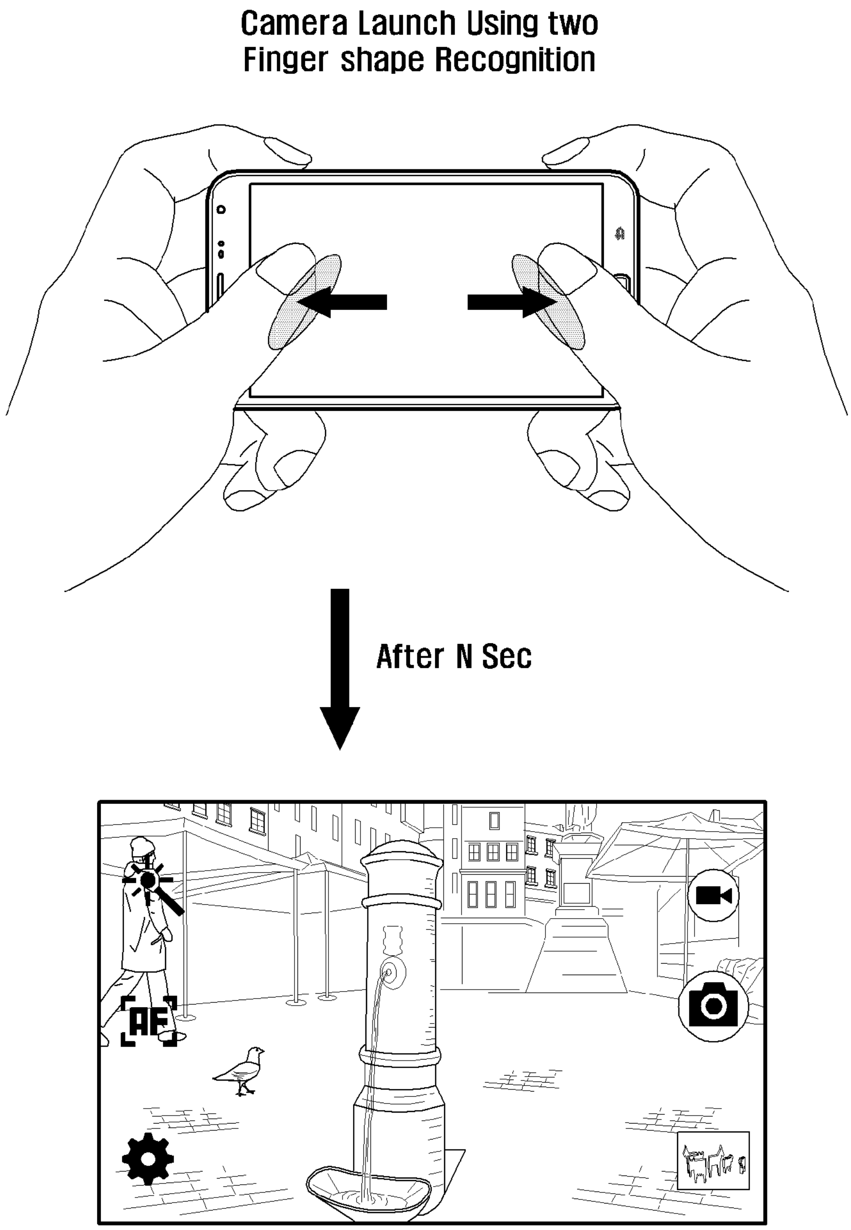


[Fig. 20]

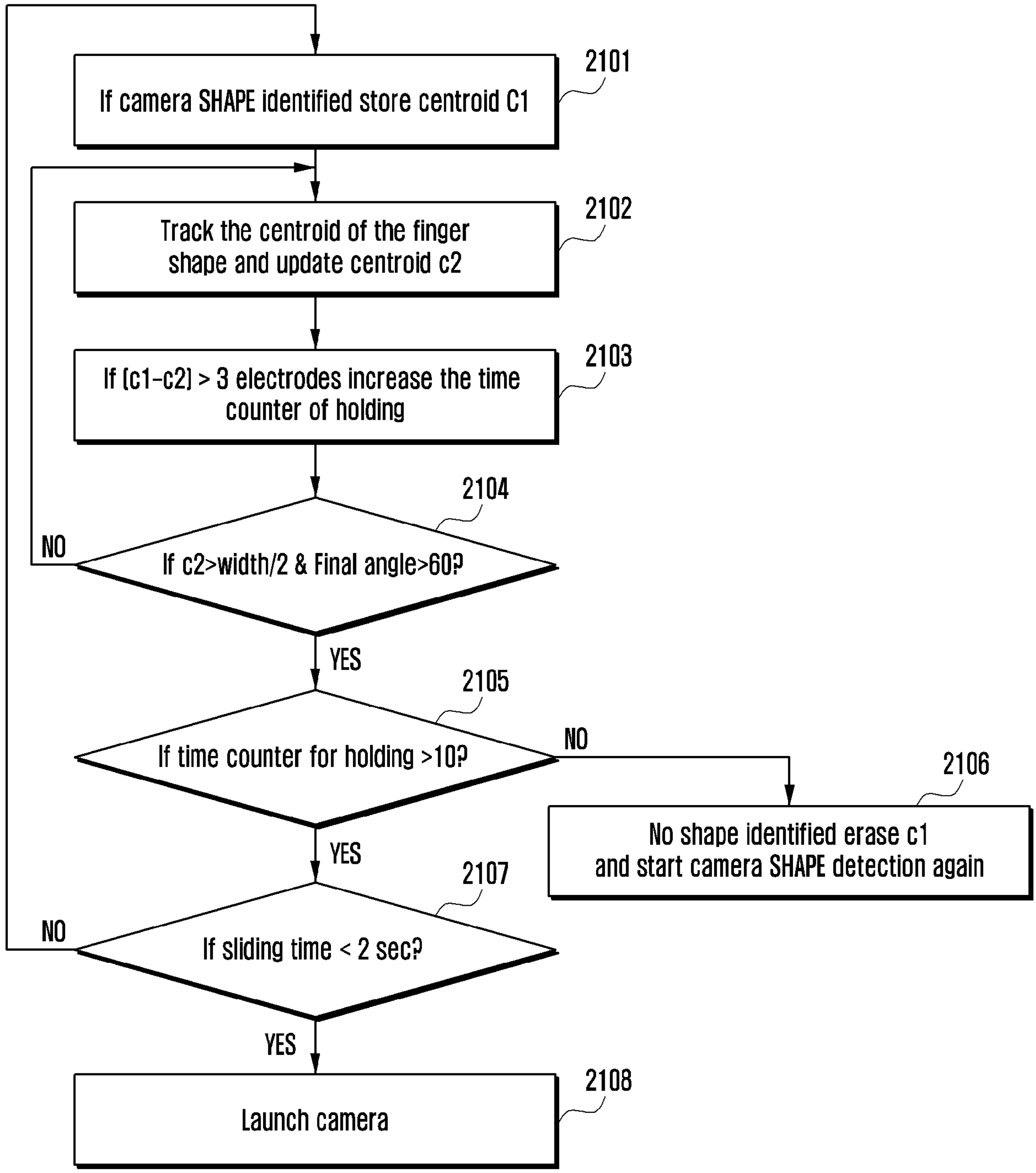
Camera launch Using single
Finger shape Recognition



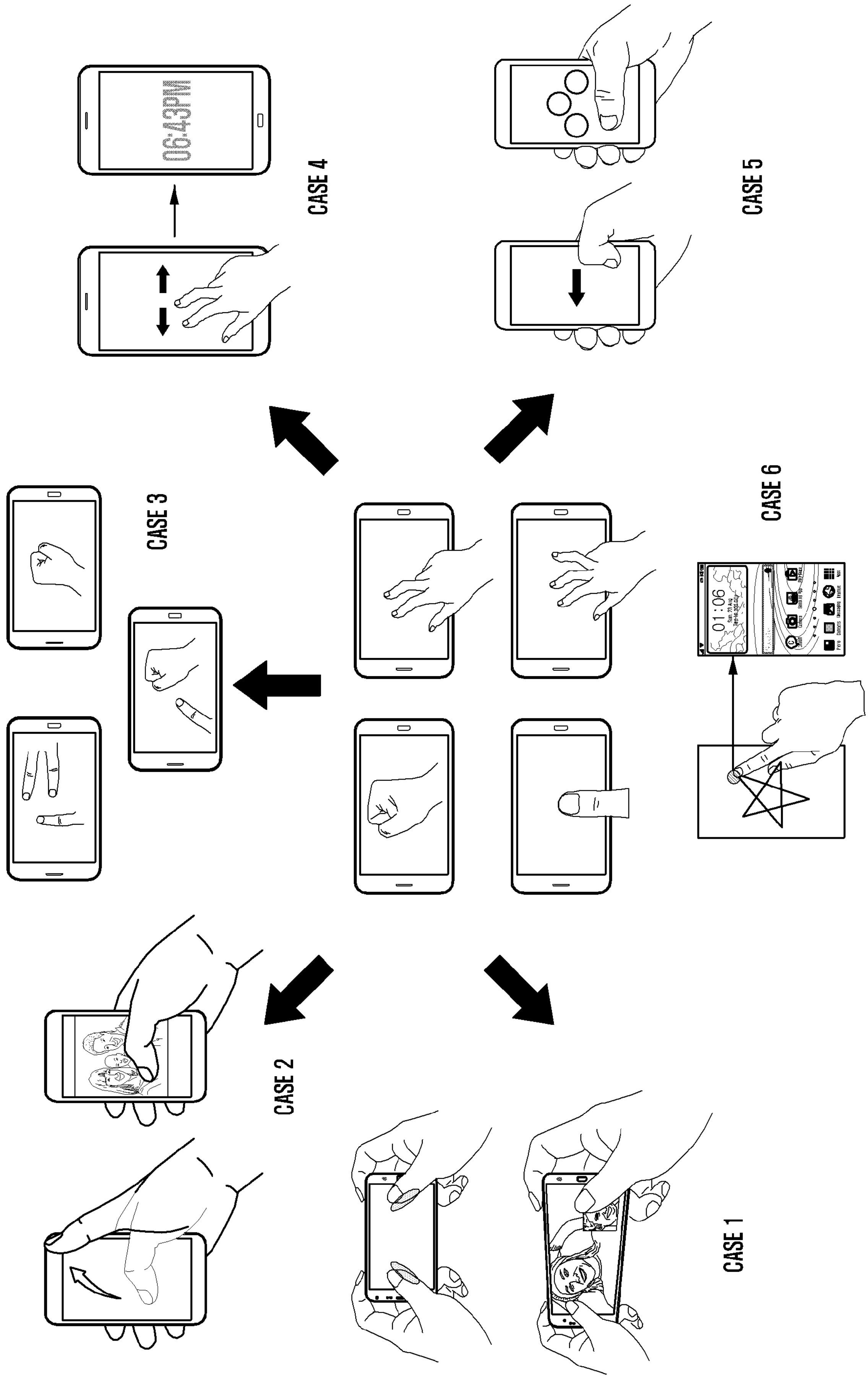
[Fig. 21a]


Camera Launch Using two
Finger shape Recognition
After N Sec
AF

[Fig. 21b]


If camera SHAPE identified store centroid C1
2101
Track the centroid of the finger shape and update centroid c2
2102
If (c1-c2) > 3 electrodes increase the time counter of holding
2103
2104
NO
If c2>width/2 & Final angle>60?
YES
2105
If time counter for holding >10?
NO
2106
No shape identified erase c1 and start camera SHAPE detection again
YES
2107
NO
If sliding time < 2 sec?
YES
2108
Launch camera

[Fig. 22]



INTERNATIONAL SEARCH REPORT

International application No.
PCT/KR2015/007020

A. CLASSIFICATION OF SUBJECT MATTER

G06F 3/0488(2013.01)i, G06F 3/01(2006.01)i, G06F 3/041(2006.01)i, G06F 3/044(2006.01)i

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

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)
G06F 3/0488; H04W 88/02; G06F 3/044; G06F 3/033; G06F 3/03; H04M 3/00; G06F 3/041; G06F 3/01

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched
Korean utility models and applications for utility models
Japanese utility models and applications for utility models

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
eKOMPASS(KIPO internal) & Keywords: touch, input, shape, predefined, action, gesture

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
X	US 2011-0130170 A1 (JAEWON HAN et al.) 02 June 2011 See paragraphs [0011]-[0017], [0038], [0050]-[0059], [0090], [0101]-[0103]; and figures 5-6.	1,8-12,16-17,21-22,24-26,30
Y		2-6,13-14,18,23,27-28
A		7,15,19-20,29
Y	US 2010-0211920 A1 (WAYNE CARL WESTERMAN et al.) 19 August 2010 See paragraphs [0063]-[0066], [0153]; and figure 3.	3-6,13-14,18,23,27-28
Y	US 2010-0099394 A1 (RICHARD ANDREAS AXEL HAINZL) 22 April 2010 See paragraph [0045]; and figure 5.	2
A	WO 2013-119476 A1 (MICROSOFT CORPORATION) 15 August 2013 See paragraphs [0003]-[0004]; and figure 19.	1-30
A	US 2014-0146000 A1 (SAMSUNG ELECTRO-MECHANICS CO., LTD.) 29 May 2014 See paragraphs [0010]-[0012]; and figures 2-3.	1-30

☐ Further documents are listed in the continuation of Box C.

☒ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier application or patent but published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents,such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
17 December 2015 (17.12.2015)	**17 December 2015 (17.12.2015)**

Name and mailing address of the ISA/KR
International Application Division
Korean Intellectual Property Office
189 Cheongsa-ro, Seo-gu, Daejeon Metropolitan City, 35208, Republic of Korea
Facsimile No. +82-42-472-7140

Authorized officer
LEE, Dong Yun
Telephone No. +82-42-481-8734

INTERNATIONAL SEARCH REPORT
Information on patent family members

International application No.
PCT/KR2015/007020

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2011-0130170 A1	02/06/2011	KR 10-2011-0008940 A	27/01/2011
		KR 10-2011-0042634 A	27/04/2011
		US 2015-0077377 A1	19/03/2015
US 2010-0211920 A1	19/08/2010	US 2008-0168403 A1	10/07/2008
		US 2010-0192109 A1	29/07/2010
		US 7877707 B2	25/01/2011
		WO 2008-085788 A2	17/07/2008
		WO 2008-085788 A3	05/03/2009
US 2010-0099394 A1	22/04/2010	EP 2368165 A1	28/09/2011
		TW 201017476 A	01/05/2010
		US 8385885 B2	26/02/2013
		WO 2010-043277 A1	22/04/2010
WO 2013-119476 A1	15/08/2013	US 2013-0201113 A1	08/08/2013
		US 8902181 B2	02/12/2014
US 2014-0146000 A1	29/05/2014	KR 10-2014-0067379 A	05/06/2014