Innovation, Sciences et Développement économique Canada
Office de la Propriété Intellectuelle du Canada

Innovation, Science and Economic Development Canada
Canadian Intellectual Property Office

CA 3067400 A1 2021/05/28

(21) 3 067 400

(12) DEMANDE DE BREVET CANADIEN
CANADIAN PATENT APPLICATION

(13) A1

(22) Date de dépôt/Filing Date: 2019/11/29

(41) Mise à la disp. pub./Open to Public Insp.: 2021/05/28

(30) Priorité/Priority: 2019/11/28 (US62/941,779)

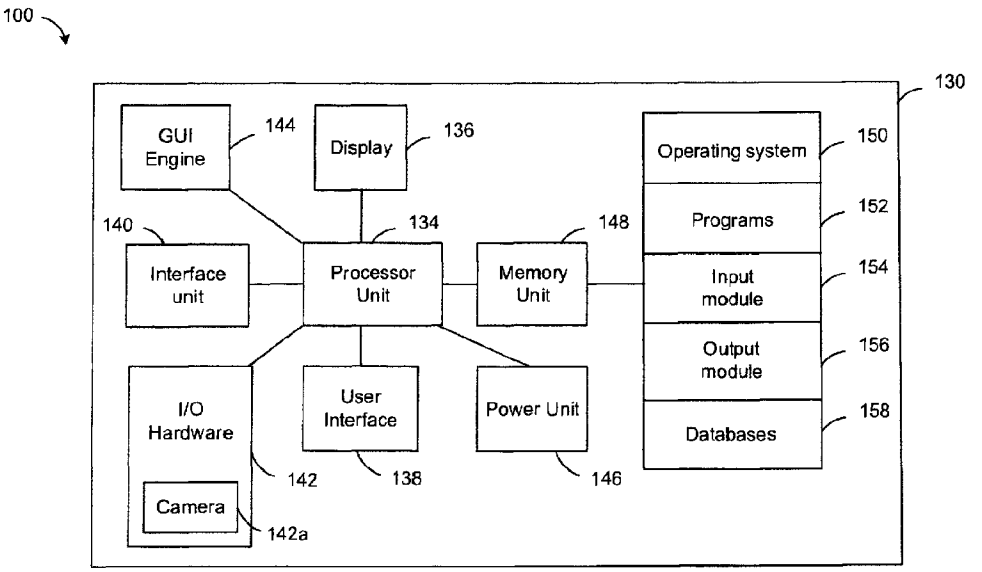
(51) Cl.Int./Int.Cl. *G01B 21/20* (2006.01), *A43D 1/02* (2006.01), *G01B 11/245* (2006.01), *G01J 5/00* (2006.01), *G06T 11/60* (2006.01), *G06T 3/00* (2006.01)

(71) Demandeur/Applicant:
XESTO INC., CA

(72) Inventeurs/Inventors:
AKDEMIR, MEHMET AFINY AFFAN, CA;
SALGUERO, CHRISTIAN GARCIA, CA;
HOWE, VICTORIA SOPHIE, CA

(74) Agent: PRIMA IP

(54) Titre : SYSTEME ET METHODE POUR LA MESURE ET LA NUMERISATION 3D MOBILE
(54) Title: SYSTEM AND METHOD FOR MOBILE 3D SCANNING AND MEASUREMENT



(57) Abrégé/Abstract:
Computer-implemented methods and systems are provided for three-dimensional scanning and measurement by a device having a processor. The processor is configured to receive images of an object from at least two angles; preprocess the images using morphological refinement; create a source point cloud based on the images; remove outliers from the source point cloud; globally register the source point cloud to generate a transformed source point cloud; compare the transformed source point cloud with a target point cloud to generate a stitched point cloud that thereby creates a stitched 3D model; measure the resulting stitched 3D model; and output the resulting stitched 3D model in a format capable of being displayed.

ABSTRACT

Computer-implemented methods and systems are provided for three-dimensional scanning and measurement by a device having a processor. The processor is configured to receive images of an object from at least two angles; preprocess the images using morphological refinement; create a source point cloud based on the images; remove outliers from the source point cloud; globally register the source point cloud to generate a transformed source point cloud; compare the transformed source point cloud with a target point cloud to generate a stitched point cloud that thereby creates a stitched 3D model; measure the resulting stitched 3D model; and output the resulting stitched 3D model in a format capable of being displayed.

SYSTEM AND METHOD FOR MOBILE 3D SCANNING AND MEASUREMENT

FIELD

[0001] Various embodiments are described herein that generally relate to a mobile three-dimensional (3D) scanning and measurement device, as well as the methods and systems for recommending sizing based thereon.

BACKGROUND

[0002] The following paragraphs are provided by way of background to the present disclosure. They are not, however, an admission that anything discussed therein is prior art or part of the knowledge of persons skilled in the art.

[0003] 3D scanning and measurement can be applied to any body part and to any other object that requires sizing. Some such examples include a wrist, chest or breasts, and feet.

[0004] Methods of measuring feet and body parts using mechanical measuring devices have developed over time, from the Brannock device to currently available electronic measuring instruments that use contact sensors. In addition to mechanical measuring devices, advances in electrical and optical devices have made it possible to have non-contact measurement devices. Non-contact measurement can be performed using such techniques as structured light vision, stereoscopic vision, and laser measurement.

[0005] In the structured light 3D vision system, the measurement is based on the principle of optical triangulation. The basic principle is that a structured light projector projects a controllable light spot or light bar onto the surface of the object to be measured. A camera obtains an image, and the three-dimensional coordinates of the object are calculated using trigonometry. The disadvantages of structured light vision include: (a) measurement accuracy limited by physical optics; (b) object occlusion; and (c) the inverse relationship between measurement accuracy and speed.

[0006] In the stereoscopic vision system, two cameras with a relatively fixed position are used to acquire two images of the same scene from different angles at two different positions, by calculating the spatial points in the two images. Parallax is used to get the three-dimensional coordinate values. The disadvantages of stereoscopic vision include: (a) large amounts of data processing; (b) long processing times; (c) the necessity of matching two images; and (d) reduced matching and measurement accuracy when the surface gray level and surface shape change by a small amount.

[0007] In the laser measurement system, a visible laser beam is positioned by a polygonal lens, and the laser beam scans and measures the surface of the object using high-frequency scanning. The laser beam is reflected by the surface of the object and then received by the laser receiver. The system can then calculate the coordinates of the surface of the object. The disadvantages of the laser measurement system include: (a) high costs; and (b) the inverse relationship between accuracy and the scanning rate.

[0008] After acquiring a set of 3D scans of the same object from multiple point clouds for the purpose of 3D measurement, often the required measurement on the object spans across different scans that are partially overlapping. It is possible to combine the multiple 3D scans through finding a rigid transformation between each scan. This can be performed using the iterative closest point algorithm (ICP), which is a method of estimating the optimal alignment between two 3D objects. The disadvantages of ICP include (a) the susceptibility of ICP to get stuck in local minima, which corresponds to an incorrect alignment; and (b) movement of the object in between the scans deforming the intermediate object to the point where no rigid transformation can be found. This is relevant in the context of body part scans, as users of the system do not stand still in between scans.

[0009] There is a need for a system and device for 3D scanning and measurement and use thereof that address the challenges and/or shortcomings described above.

SUMMARY OF VARIOUS EMBODIMENTS

[0010] Various embodiments of a device for mobile 3D scanning and measurement, system and methods of use thereof, and computer products for use therewith, are provided according to the teachings herein.

[0011] According to one aspect of the invention, there is disclosed a computer-implemented method for three-dimensional scanning and measurement comprising: receiving a plurality of images of an object, the plurality of images providing views of an object from at least two angles; preprocessing the plurality of images using morphological refinement; creating a source point cloud based at least in part on the plurality of images; removing outliers from the source point cloud; globally registering the source point cloud, thereby creating a globally registered source point cloud; generating a transformed source point cloud based at least in part on the globally registered source point cloud; comparing the transformed source point cloud with a target point cloud, thereby creating a point cloud comparison; generating a stitched point cloud based at least in part on the point cloud comparison, thereby creating a resulting stitched 3D model; measuring the resulting stitched 3D model; and outputting the resulting stitched 3D model in a format capable of being displayed.

[0012] In at least one embodiment, the method further comprises inputting the source point cloud into a neural network configured for parameter optimization based at least in part on one of a significant amount of overlap between the stitched point cloud or labeled correspondences.

[0013] In at least one embodiment, the morphological refinement causes morphological changes by at least one of an erosion of points in a 3D model, noise removal, or edge refinement.

[0014] In at least one embodiment, the 3D model is a 3D point cloud.

[0015] In at least one embodiment, the morphological refinement is based at least in part on at least one of filtering masks or kernels.

[0016] In at least one embodiment, the method further comprises inputting the source point cloud into a neural network, the source point cloud having a surface with a discontinuity on the surface and a radius for each point around the discontinuity, the neural network configured to output an optimal radius for each point around the discontinuity that corresponds to areas to be removed from the source point cloud.

[0017] In at least one embodiment, creating the source point cloud is based at least in part on creating a geometrical representation of the plurality of images using spatial information.

[0018] In at least one embodiment, the spatial information is based at least in part on camera intrinsics of a camera that scanned the plurality of images.

[0019] In at least one embodiment, removing outliers comprises 2D or 3D processing that accentuates areas of interest of the object.

[0020] In at least one embodiment, globally registering the source point cloud comprises aligning 3D assets using geometrically relevant features, and generating a transformed source asset from the aligned 3D assets.

[0021] In at least one embodiment, comparing the transformed source point cloud with a target point cloud comprises iterative closes point (ICP) stitching to generate the stitched point cloud.

[0022] In at least one embodiment, the ICP stitching comprises an overlap step.

[0023] In at least one embodiment, measuring the resulting stitched 3D model comprises using a distance function.

[0024] In another aspect, there is provided a computer-implemented method for deformable object stitching comprising: receiving a first point cloud and a second point cloud having a partial overlap; finding first prominent features in a subregion G_1 and a subregion G_2 in a first region of the partial overlap over the first point cloud; computing first geodesics based on the first prominent features; finding second prominent features in a

subregion H_1 and a subregion H_2 in a second region of the partial overlap over the second point cloud; computing second geodesics based on the second prominent features; building a first correspondence between the subregion G_1 and the subregion H_1; building a second correspondence between the subregion G_2 and the subregion H_2; deforming the first geodesics and the second geodesics based on the first correspondence and the second correspondence; stitching the first point cloud and the second point cloud, thereby generating a stitched point cloud; calculating a required deformation to transform the first geodesics to the second geodesics; and applying the required deformation to the second point cloud to match the second point cloud to the first point cloud, thereby refining the stitched point cloud; and outputting the refined stitched point cloud in a format capable of being displayed.

[0025] In at least one embodiment, finding first prominent features comprises estimating a curvature in subregion G_1 or subregion G_2, then refining the estimated curvature by iteratively calculating the curvature over one or more iterations, such that a region of high curvature is re-calculated by considering a smaller neighborhood in the region, and calculating the curvature based on the smaller neighborhood, repeating the iterations until sampling in a smaller neighborhood does not increase the curvature in the region by more than a specified threshold.

[0026] In at least one embodiment, finding second prominent features comprises estimating a curvature in subregion H_1 or subregion H_2, then refining the estimated curvature by iteratively calculating the curvature over one or more iterations, such that a region of high curvature is re-calculated by considering a smaller neighborhood in the region, and calculating the curvature based on the smaller neighborhood, repeating the iterations until sampling in a smaller neighborhood does not increase the curvature in the region by more than a specified threshold.

[0027] In at least one embodiment, finding the first prominent features comprises approximating a curvature on subregion G_1 and subregion G_2 by

discretizing a curvature operator, and extracting the first prominent features with respect to the curvature where the curvature is high or zero, a high curvature representing sharp corners and a zero curvature representing a flat region.

[0028] In at least one embodiment, finding the second prominent features comprises approximating a curvature on subregion H_1 and subregion H_2 by discretizing a curvature operator, and extracting the second prominent features with respect to the curvature where the curvature is high or zero, a high curvature representing sharp corners and a zero curvature representing a flat region.

[0029] In at least one embodiment, computing the first geodesics and computing the second geodesics comprises triangulating on respective subregions and determining a shortest path between two points on a triangulated surface.

[0030] In at least one embodiment, building the first correspondence and building the second correspondence comprises ranking the first prominent features and the second prominent features, respectively, by curvature in the first point cloud and the second point cloud, respectively, and matching points therebetween.

[0031] In at least one embodiment, calculating the required deformation comprises producing a surface determined by connecting any two points between subregion G_1 and subregion H_1, producing a geodesic, and extending the geodesic into a surface by considering a small neighborhood about the geodesic.

[0032] In at least one embodiment, calculating the required deformation further comprises estimating a first fundamental form of the surface by finite difference methods, and deforming the surface by an As Rigid As Possible (ARAP) deformation using the first correspondence and the second correspondence.

[0033] In at least one embodiment, calculating the required deformation further comprises determining a similarity between deformed subregions by estimating a second fundamental form for each of the subregions, and comparing a coefficient between the first fundamental form and the second fundamental form.

[0034] In another aspect, there is provided a computer-implemented method for fitting a first 3D model of an item over a second 3D model of an object comprising: receiving a model point cloud of the item and an object point cloud of the object; aligning the object point cloud inside the model point cloud; extracting a surface heat map based at least in part on an interaction of the object point cloud with the model point cloud, the surface heat map having highest heat values; determining a corresponding region on the model point cloud based at least in part on the highest heat values; determining an elasticity of the corresponding region based at least in part on material properties of the item; and displaying an enhanced surface heat map based at least in part on the surface heat map and the elasticity of the corresponding region.

[0035] In at least one embodiment, determining the elasticity of the corresponding region is based at least in part on a predetermined set of weights, where each element in the set corresponds to the elasticity of a particular physical material of the object.

[0036] In another aspect, there is provided a system for providing three-dimensional scanning and measurement comprising a data store and at least one processor coupled to the data store, the data store comprising a non-transient computer-readable storage medium having stored thereon computer-executable instructions for execution by the processor to perform the method for providing three-dimensional scanning and measurement.

[0037] In another aspect, there is provided a system for deformable object stitching comprising a data store and at least one processor coupled to the data store, the data store comprising a non-transient computer-readable storage medium having stored thereon computer-executable instructions for

execution by the processor to perform the method for deformable object stitching.

[0038] In another aspect, there is provided a system for fitting a first 3D model of an item over a second 3D model of an object comprising a data store and at least one processor coupled to the data store, the data store comprising a non-transient computer-readable storage medium having stored thereon computer-executable instructions for execution by the processor to perform the method for fitting a first 3D model of an item over a second 3D model.

[0039] In another aspect, there is provided a computer-readable medium comprising a plurality of instructions that are executable on a processor of a system for adapting the system to implement the method for providing three-dimensional scanning and measurement.

[0040] In another aspect, there is provided a computer-readable medium comprising a plurality of instructions that are executable on a processor of a system for adapting the system to implement the method for deformable object stitching.

[0041] In another aspect, there is provided a computer-readable medium comprising a plurality of instructions that are executable on a processor of a system for adapting the system to implement the method for fitting a first 3D model of an item over a second 3D model of an object.

[0042] Other features and advantages of the present application will become apparent from the following detailed description taken together with the accompanying drawings. It should be understood, however, that the detailed description and the specific examples, while indicating preferred embodiments of the application, are given by way of illustration only, since various changes and modifications within the spirit and scope of the application will become apparent to those skilled in the art from this detailed description.

BRIEF DESCRIPTION OF THE DRAWINGS

[0043] For a better understanding of the various embodiments described herein, and to show more clearly how these various embodiments may be carried into effect, reference will be made, by way of example, to the accompanying drawings which show at least one example embodiment, and which are now described. The drawings are not intended to limit the scope of the teachings described herein.

[0044] FIG. 1 shows a block diagram of an example embodiment of a system for mobile 3D scanning and measurement.

[0045] FIG. 2 shows a flow chart of an example embodiment of a method of 3D mobile scanning and measurement.

[0046] FIG. 3A shows an example of an image of an object before applying a morphological refinement step.

[0047] FIG. 3B shows an example of an image of an object after applying the morphological refinement step.

[0048] FIG. 4A shows an example of a point cloud before 2-dimensional (2D) and 3-dimensional (3D) outlier removal.

[0049] FIG. 4B shows an example of a point cloud after 2D and 3D outlier removal.

[0050] FIG. 5 shows an example of a point cloud after extracting areas of importance.

[0051] FIG. 6A shows an example of a point cloud before global alignment

[0052] FIG. 6B shows an example of a point cloud after global alignment.

[0053] FIG. 7A shows an example of a point cloud after gathering points from dense source point clouds.

[0054] FIG. 7B shows an example of a point cloud after gathering points from a dense target point cloud.

[0055] FIG. 7C shows an example of a point cloud after an overlap step.

[0056] FIG. 8A shows an example of a point cloud after stitching.

[0057] FIG. 8B shows an example of a point cloud after aggregation of stitching of all pairs of images.

[0058] FIG. 8C shows an example of a segmented and extracted object and its corresponding measurement using an appropriate distance function.

[0059] FIG. 9 shows a flow chart of an example embodiment of a method of deformable object stitching.

[0060] FIG. 10 shows a flow chart of an example embodiment of a method of fitting a 3D model item over another 3D model.

[0061] FIG. 11A shows an example of a point cloud of a 3D model of a sample of an artifact such as a shoe.

[0062] FIG. 11B shows an example of a point cloud of the interior of a shoe aligned with a foot.

[0063] FIG. 12 shows an example of a heat map of an interior geometry such as a foot to be fitted onto an artifact such as a shoe which represents a foot interacting with the inside of a shoe.

[0064] FIG. 13 shows an example of an enhanced heat map.

[0065] Further aspects and features of the example embodiments described herein will appear from the following description taken together with the accompanying drawings.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0066] Various embodiments in accordance with the teachings herein will be described below to provide an example of at least one embodiment of the claimed subject matter. No embodiment described herein limits any claimed subject matter. The claimed subject matter is not limited to devices, systems, or methods having all of the features of any one of the devices, systems, or methods described below or to features common to multiple or all of the devices, systems, or methods described herein. It is possible that there may be a device, system, or method described herein that is not an embodiment

of any claimed subject matter. Any subject matter that is described herein that is not claimed in this document may be the subject matter of another protective instrument, for example, a continuing patent application, and the applicants, inventors, or owners do not intend to abandon, disclaim, or dedicate to the public any such subject matter by its disclosure in this document.

[0067] All novel and nonobvious combinations and sub-combinations are included in the subject matter described herein. This includes combinations and sub-combinations of systems, pipelines, features, or implications of the features, as well as functions, acts, and/or properties disclosed herein and all of the implications thereof.

[0068] It will be appreciated that for simplicity and clarity of illustration, where considered appropriate, reference numerals may be repeated among the figures to indicate corresponding or analogous elements. In addition, numerous specific details are set forth in order to provide a thorough understanding of the embodiments described herein. However, it will be understood by those of ordinary skill in the art that the embodiments described herein may be practiced without these specific details. In other instances, well-known methods, procedures, and components have not been described in detail so as not to obscure the embodiments described herein. Also, the description is not to be considered as limiting the scope of the embodiments described herein.

[0069] It should also be noted that the terms "coupled" or "coupling" as used herein can have several different meanings depending on the context in which these terms are used. For example, the terms coupled or coupling can have a mechanical or electrical connotation. For example, as used herein, the terms coupled or coupling can indicate that two elements or devices can be directly connected to one another or connected to one another through one or more intermediate elements or devices via an electrical signal, electrical connection, or a mechanical element depending on the particular context.

[0070] It should also be noted that, as used herein, the wording "and/or" is intended to represent an inclusive-or. That is, "X and/or Y" is intended to mean X or Y or both, for example. As a further example, "X, Y, and/or Z" is intended to mean X or Y or Z or any combination thereof.

[0071] It should be noted that terms of degree such as "substantially", "about" and "approximately" as used herein mean a reasonable amount of deviation of the modified term such that the end result is not significantly changed. These terms of degree may also be construed as including a deviation of the modified term, such as by 1%, 2%, 5%, or 10%, for example, if this deviation does not negate the meaning of the term it modifies.

[0072] Furthermore, the recitation of numerical ranges by endpoints herein includes all numbers and fractions subsumed within that range (e.g., 1 to 5 includes 1, 1.5, 2, 2.75, 3, 3.90, 4, and 5). It is also to be understood that all numbers and fractions thereof are presumed to be modified by the term "about" which means a variation of up to a certain amount of the number to which reference is being made if the end result is not significantly changed, such as 1%, 2%, 5%, or 10%, for example.

[0073] It should also be noted that the use of the term "window" in conjunction with describing the operation of any system or method described herein is meant to be understood as describing a user interface for performing initialization, configuration, or other user operations.

[0074] The example embodiments of the devices, systems, or methods described in accordance with the teachings herein may be implemented as a combination of hardware and software. For example, the embodiments described herein may be implemented, at least in part, by using one or more computer programs, executing on one or more programmable devices comprising at least one processing element and at least one storage element (i.e., at least one volatile memory element and at least one non-volatile memory element). The hardware may comprise input devices including at least one of a touch screen, a keyboard, a mouse, buttons, keys, sliders, and the like, as well as more specialized input devices such as a controller or a

sensor input for a depth image. The hardware may comprise output devices including one or more of a display, a printer, and the like depending on the implementation of the hardware. The combination of software and hardware may include single core or multicore processors with programs (or engines, a library, an API, scripts, and the like) executed on any combination of single, parallel, and distributed processing elements, which may be local or remotely located.

[0075] It should also be noted that there may be some elements that are used to implement at least part of the embodiments described herein that may be implemented via software that is written in a high-level procedural language such as object-oriented programming. The program code may be written in C++, C#, JavaScript, Python, or any other suitable programming language and may comprise modules or classes, as is known to those skilled in object-oriented programming. Alternatively, or in addition thereto, some of these elements implemented via software may be written in assembly language, machine language, or firmware as needed. In either case, the language may be a compiled or interpreted language.

[0076] At least some of these software programs may be stored on a computer-readable medium such as, but not limited to, a ROM, a magnetic disk, an optical disc, a USB key, and the like that is readable by a device having a processor, an operating system, and the associated hardware and software that is necessary to implement the functionality of at least one of the embodiments described herein. The software program code, when read by the device, configures the device to operate in a new, specific, and predefined manner (e.g., as a specific-purpose computer) in order to perform at least one of the methods described herein.

[0077] At least some of the programs associated with the devices, systems, and methods of the embodiments described herein may be capable of being distributed in a computer program product comprising a computer-readable medium that bears computer-usable instructions, such as program code, for one or more processing units. The medium may be provided in

various forms, including non-transitory forms such as, but not limited to, one or more diskettes, compact disks, tapes, chips, and magnetic and electronic storage. In alternative embodiments, the medium may be transitory in nature such as, but not limited to, wire-line transmissions, satellite transmissions, internet transmissions (e.g., downloads), media, digital and analog signals, and the like. The computer-usable instructions may also be in various formats, including compiled and non-compiled code.

[0078] Data storage systems may include any form of removable media or standalone devices of physical, non-transitory devices to hold data and or any logical instructions or signals required to support the method and processes described herein.

[0079] In accordance with the teachings herein, there are provided various embodiments for a device for mobile 3D scanning and measurement, system and methods of use thereof, and computer products for use therewith.

[0080] A system that employs mobile devices in a mobile environment is a non-limiting example. The method used in a mobile device for 3D scanning and measurement may apply to a stationary or a motionless environment or any combination of mobile and any other such non-mobile device made to perform same method or process. In accordance with the teachings herein, there are various embodiments for a device for stationary usage or a combination of stationary with a mobile device, any of which use 3D scanning and measurement, system, and methods of use thereof, and computer products for use therewith that are made to perform same method or process implemented in any online or offline computing system, under any suitable system configuration.

[0081] Broadly, at least one embodiment described in accordance with the teachings herein relates to using a mobile device for 3D scanning of an object to provide a measurement for at least one part of the object. One or more processors (e.g., on the mobile device and/or a remote server) executes program code to: (1) acquire front camera images of the object from a few angles; (2) preprocess the images in 2D and 3D using geometrical structures

such as point clouds or meshes; (3) utilize a dynamic global alignment procedure to initialize for a better iterative closest point (ICP) stitch; (4) compare points in the aligned source point cloud with points in the target point cloud; and (5) segment and extract the required object and measure the extracted object using an appropriate distance function.

[0082] Reference is first made to FIG. 1, showing a block diagram of an example embodiment of a system 100 for mobile 3D scanning and measurement described herein. The system 100 comprises a device 130 that can be used to perform the various scanning and measurement functions. The device 130 includes a processor unit 134, a display 136, a user interface 138, an interface unit 140, input/output (I/O) hardware 142 having a camera 142a, a Graphical User Interface (GUI) unit 144, a power unit 146, and a memory unit (also referred to as "data store") 148. In other embodiments, the device 130 may have more or less components but generally function in a similar manner.

[0083] The processor unit 134 controls the operation of the device 130 and can be any suitable processor, controller, or digital signal processor that can provide sufficient processing power depending on the configuration, purposes, and requirements of the device 130 as is known by those skilled in the art. The processor unit 134 may include one processor. Alternatively, there may be a plurality of processors that are used by the processor unit 134, and these processors may function in parallel and perform certain functions. In alternative embodiments, specialized hardware can be used to provide some of the functions provided by the processor unit 134.

[0084] The processor unit 134 can execute a graphical user interface (GUI) engine 144 that is used to generate various GUIs, some examples of which are shown and described herein. The GUI engine 144 provides data according to a certain layout and also receives inputs from a user. The processor unit 134 then uses the inputs received by the GUI from the user to change the operation of the various methods that may be performed in

accordance with the teachings herein, to change data that is shown on the display 136, or to show a different GUI.

[0085] The display 136 can be any suitable display that provides visual information depending on the configuration of the device 130. For instance, the display 136 may output the various GUIs that are generated by the GUI engine 139. The display 136 may be, but not limited to, a computer monitor or an LCD display depending on the implementation of the electronic device (e.g., a smartphone, a tablet, a laptop, or a desktop computer).

[0086] The user interface 138 can include at least one of a mouse, a keyboard, a touch screen, a thumbwheel, a track-pad, a track-ball, a card-reader, gesture or voice recognition software, a virtual reality (VR) headset, and the like again depending on the particular implementation of the device 130. In some cases, some of these components can be integrated with one another.

[0087] The interface unit 140 can be any interface that allows the device 130 to communicate with other devices or computers, such as a remote computer or server (e.g., back-end server, web server, application server). In some cases, the interface unit 140 can include at least one of a serial port, a parallel port, or a USB port that provides USB connectivity. The interface unit 140 can also include at least one of a Wi-Fi, Local Area Network (LAN), Wide Area Network (WAN), Neighborhood Area Network (NAN), Ethernet, Firewire, modem, digital subscriber line connection, or Internet connection. For example, the interface unit 140 can include a standard network adapter such as an Ethernet or 802.11x adapter. The interface unit 140 may include a radio that communicates utilizing CDMA, GSM, GPRS, or Bluetooth protocol according to standards such as the IEEE 802.11 family of standards such as IEEE 802.11a, 802.11b, 802.11g, or 802.11n. If such communication is needed, the method or process described in this disclosure may be implemented on an available or custom-designed network technology such as but not limited to power line communication carriers or 5G network infrastructure. Various combinations of these elements, and any other

element with similar functionality, can be incorporated within the interface unit 140.

[0088] The I/O hardware 142 includes, but is not limited to, at least one of a microphone, a speaker, and a printer, for example, depending on the implementation of the device 130. The I/O hardware 142 includes a camera 142a that can be used to obtain the images or video frames of an object to be modeled or measured. The camera 142a may be an RGB-D (i.e., red, green, blue, and depth) camera, a Kinect camera, a Time of Flight (TOF) camera, an infrared camera, or a thermal imaging camera, or any other camera (or plurality of cameras) that has (or provides) 3D imaging capability. The camera 142a may, for example, be one or more cameras located on any combination of the front or back of the device 130, such as a pair of front and back cameras on a smartphone, or a pair of front cameras on a smartphone, or two front cameras and one back camera on a smartphone.

[0089] The power unit 146 can be any suitable power source that provides power to the device 130 such as a power adaptor or a rechargeable battery pack depending on the implementation of the device 130 as is known by those skilled in the art.

[0090] The memory unit 148 can include RAM, ROM, one or more hard drives, one or more flash drives, or some other suitable data storage elements such as disk drives, solid state drives, etc. The memory unit 148 may store the program instructions for an operating system 150, program code 152 for various applications, an input module 154, an output module 156, and a database 158. The programs 152 comprise program code that, when executed, configures the processor unit 134 to operate in a particular manner to implement various functions, tools, processes, and/or methods for the device 130. For example, the program code 152 may include software instructions for various methods described in accordance with the teachings herein. The memory unit 148 may also store various operational parameters, video recordings, images, and/or past results in the database 158.

[0091] In at least one embodiment, the programs 152 comprise program code that, when executed, configures the processor unit 134 to cause data to be sent or received via the interface unit 140 to or from a remote computer or server such that at least part of the software instructions for the various methods described in accordance with the teachings herein are performed by the remote computer or server.

[0092] Referring now to FIG. 2, shown therein is a flow chart of an example embodiment of a method 200 of 3D mobile scanning and measurement. Method 200 may be carried out, for example, by mobile device 130, processor unit 134, and/or a remote computer or server. At any of the acts/steps in method 200, dynamic approaches may involve machine learning, which in one embodiment, for instance, may come in the form of parameter optimization, where the objective is to create good stitches, where good can be defined, for example, as a significant amount of overlap between stitched point clouds or based on labeled correspondences.

[0093] Depending on the use case, the type of the object to be stitched, and the sensor being used, as well as other characteristics specific to the problem, the system 100 may need to be tuned in order to achieve the optimal stitching. The optimal stitch may be determined by deducing the optimal parameters relevant to the problem. In at least one embodiment, the optimal parameters are determined by observational data. In at least one embodiment, the training and test sets consist of point clouds to be stitched, together with a set of features on these point clouds. The machine learning algorithm may, for example, learn the optimal parameters for the system 100 and output the optimal transformation. For example, suppose point cloud A and point cloud B each have 10 labeled feature points that correspond to each other. In such a case, the loss function may incorporate the absolute distance between the 10 corresponding labeled feature points after the stitch is performed.

[0094] A variety of machine learning algorithms may be used to achieve the above. As a non-limiting example, the parameter training module may be

trained with a supervised learning algorithm such as a convolutional neural network (CNN). Other available machine learning techniques such as recurrent neural networks, Bayesian neural networks, and boosting and bagging algorithms may be used.

[0095] At 210, the mobile device 130 acquires camera images of an object of interest from different angles or a video stream. The mobile device 130 may collect depth images from a camera 142a, in which case a front camera and/or a back camera, for example, may be used.

[0096] At 220, the mobile device 130 preprocesses the images using morphological refinement. This can be done either in 2D or 3D inputs. The preprocessing may include filtering masks or kernels to accentuate edges and remove noise around them. FIGS. 3A and 3B show an example of an image 300 of the object (here, a foot) before and after, respectively, one such possible filtering kernel is applied to accentuate edges 320. The preprocessing may include a filtering step to remove noise around edges.

[0097] In at least one embodiment, everything is done in 3D. The edges in 3D are defined as regions that have any discontinuity in the point cloud/mesh. After identifying such regions, a 3D ball of a dynamic radius (defined by the region of the edge) slides around these edges. The mobile device 130 may then remove all the points on the point cloud/mesh that overlap with this ball.

[0098] In at least one embodiment, everything is done in 2D. The edges in 2D are defined as contours around areas of discontinuity in 2D inputs. Once such areas are identified, the mobile device 130 may remove all the points using a 2D mask.

[0099] In at least one embodiment, one or more operations are done in 3D and the remaining one or more operations are done in 2D.

[00100] Depending on the use case, the type of the object to be stitched, and the sensor being used, as well as other characteristics specific to the problem, the system 100 may need to be tuned in order to deduce the optimal radius necessary for discontinuity removal in a point cloud. In at least one

embodiment, the edge removal is done on a 2D projected depth image. The optimal radius for edge removal at a point on the depth image may be determined by observational data. In at least one embodiment, the training and test sets consist of depth images with discontinuity, together with a radius for each point along the discontinuity. The machine learning algorithm may learn the optimal radius for each point around the discontinuity, which may correspond to the areas that are to be removed. Since each point has a different radius associated to it, the predicted radius varies along a discontinuity. The localized discontinuity may be removed according to the predicted dynamic radius. The loss function may incorporate the radius associated with each point. A variety of machine learning algorithms may be used to achieve the above. As a non-limiting example, the radius detection on the image may be trained with a supervised learning algorithm such as a convolutional neural network. Other available machine learning techniques may be used, such as recurrent neural networks, Bayesian neural networks, and boosting and bagging algorithms.

[00101] In at least one embodiment, the discontinuity removal is done in 3D, where the dynamic radii can be determined for example, by training a convolutional neural network on 3D point clouds. The training and test sets may consist of point clouds with a discontinuity on the surface, together with a radius for each point around the discontinuity. The machine learning algorithm may learn the optimal radius for each point around the discontinuity which corresponds to the 3D regions that are to be removed. Since each point has a different radius associated to it, the predicted radius varies along a discontinuity. The localized discontinuity may be removed according to the predicted dynamic radius. The loss function may incorporate the radius associated with each point. A variety of machine learning algorithms may be used to achieve the above. Other available machine learning techniques may be used, such as recurrent neural networks, Bayesian neural networks, and boosting and bagging algorithms.

[00102] At 230, the mobile device 130 applies 2D and 3D outlier removal methods. FIGS. 4A and 4B show an example of a point cloud before and after, respectively, outlier 420 was removed. Areas of importance may be extracted. This may involve downsampling, such that the resultant image is the object being measured (here, the foot) and not other small objects around the object. FIG. 5 shows an example of a point cloud after extracting areas of importance.

[00103] Areas of importance may be computed through its first and second order information. This can be used to actively find parameters for downsampling and alignment. Parameter search may be based on, but it is not limited to, the distance from the camera, denseness of the local region, corners, boundaries, and/or other geometric features. The idea is that around the chosen geometric features, the number of sampled points may be larger.

[00104] The following is a non-limiting pseudocode representation of possible features or steps in a process applied to a sequence of images captured from a mobile device, the pre-processing steps, creation of a geometrical representation such as point clouds from images, and the removal of points or pixels outliers:

[00105] Input: RGB and/or Depth image;

Apply morphological refinement, if needed;

point_cloud <- depth_image + color image, Such as:

Perform Transformation on the Set of {depth_image} in order to obtain the {point_cloud} Set

This may involve using camera intrinsics to generate 3D assets where one such transformation from 2D to 3D may involve the following: 3d_pixel_coordinate = (initial_x * focal_length / initial_z) + principal_point;

[00106] Outlier removal may involve, among other possible options, statistical approaches such as:

[00107] Let n=len(point_cloud);

For p_k in point_cloud, where k $\in$ {1, ... , n}:

avg_{pk}<- average distance of neighbors of p_k

stdev.p = square_root(sum(avg_{pk} - global_mean)**2 for all p values/n-1)

For p_k in point_cloud:

inliers <- keep if avg_{pk} < global_mean(avg_{p1} ... avg_{pn}) + deviation_constant * stdev.p,

Otherwise, eliminate each outlier value whose avg_{pk} falls outside the margin established by the deviation_constant

End;

[00108] At 250, the mobile device 130 globally aligns point clouds to, for example, initialize for a better iterative closest point (ICP) stitch. The mobile device 130 may use here geometrically relevant features such as "fast histogram point features" or machine learning generated features based on labeled correspondences that aim at reducing, for instance, the distance between points in a point cloud that training data has established should be close (in a feature space) to one another. FIGS. 6A and 6B show examples of a point cloud subject to global registration.

[00109] FIG. 6A shows an example of the point cloud 600 before global alignment of the foot 610 on a surface 620, at two different angles, for which features are computed. FIG. 6B shows an example of the point cloud 650 after feature computation and after global alignment.

[00110] At 260, the mobile device 130 compares points in a transformed (based on the global registration) source point cloud with points in the target point cloud. The mobile device 130 may use a modified version of an ICP stitch, such as an "overlap" ICP. FIGS. 7A, 7B, 7C, 8A, 8B, and 8C show examples of a point cloud subject to an overlap ICP stitch.

[00111] The system 100 may register the point clouds through a comparison between source and target point clouds to find a transformation matrix based on features such as, but not limited to, the Fast Point Feature Histograms. The measurement step may be achieved with a method based on a distance function.

[00112] In at least one embodiment, the transformation matrix is defined as:

R	t
0^T	1

where R_{3x3} is a rotation matrix, 0^T_{1x3} is a zero row vector, and t_{3x1} is a transformation vector applied to each point (x,y,z) of the point cloud.

[00113] FIG. 7A shows an example of a point cloud 710 of a foot represented as a dense source point cloud 712, where points in the dense source point cloud 712 close to a downsampled representation 714 of the foot are gathered. For each point in the downsampled representation 714 of the foot, neighbors in the corresponding dense source point cloud 712 are identified. The purpose is to prepare the image for ICP stitching by getting points from a dense transformed source and dense target close to a downsampled target.

[00114] FIG. 7B shows an example of a point cloud 720 of a foot represented as a dense target point cloud 722, where points in the dense target point cloud 722 close to scattered points in a downsampled representation 714 of the foot are gathered. Again, for each point in the downsampled representation 714 of the foot, neighbors in the corresponding dense target point cloud 722 are identified. Once the points are gathered, the result is two input images shown as they appear in space. In this example, both point clouds are quite close to one another already, making final refinement with ICP stitching beneficial but not necessary.

[00115] FIG. 7C shows an example of the point cloud 730 after the overlap step, where the image can then be used as an input to the ICP stitching. This ICP procedure may be dynamic in the sense that it is based on the parameters computed in method 200, such as at 230.

[00116] FIG. 8A shows an example of the point cloud 810 after final ICP stitching with two different colors, such as dark grey 812 and light grey 814. The different colors show the resultant elements from the stitching. Alternatively, or in addition, more than two shades or different colors may be used. Alternatively, or in addition, the point cloud after final ICP stitching may be displayed using graphic symbols (or variations in shading) to represent the resultant elements from the stitching (e.g., for black and white, or monochromatic, views).

[00117] FIG. 8B shows an example of the point cloud 820 that results from applying ICP stitching for all pairs of images and then aggregating the stitches to get an image for which a measurement can be made.

[00118] At 270, the mobile device 130 measures the final stitched point cloud using, for example, a bounding box from the convex hull of the point cloud, or any appropriate distance function. Another example is the extraction of the fingers on a hand and using the geodesic distance to compute the region where there will be a ring. FIG. 8C shows an example of the segmented and extracted object 830 to be measured using an example of an appropriate distance function, utilizing a bounding box 832 and perimeter points 834.

[00119] Referring now to FIG. 9, shown therein is a flow chart of an example embodiment of a method 900 of deformable object stitching. Method 900 may be carried out, for example, by mobile device 130, processor unit 134, and/or a remote computer or server.

[00120] At 910, the mobile device 130 receives a point cloud A and a point cloud B that have partial overlap PO. Point cloud A is a deformed version of point cloud B.

[00121] At 920, the mobile device 130 finds all prominent features in G_1 and G_2 in the PO region of point cloud A. To find these points, the primary tool may be the discovery of prominent features such as, but not limited to, curvature and lack of smoothness of the point cloud in the PO region.

[00122] In at least one embodiment, the initial prominent points are discovered by first estimating the curvature of the whole point cloud. The neighborhoods with high curvature are then further refined by iteratively re-calculating curvature, wherein in the neighborhoods of prominent points, curvature is re-calculated by considering a smaller neighborhood around the prominent point. This procedure may be repeated until sampling in a smaller neighborhood does not increase the curvature in that region by more than the specified threshold.

[00123] At 922, the mobile device 130 computes geodesics from G_1 to G_2, where G_2 is re-arranged so that the geodesics connecting G_1 to G_2 do not overlap at intermediate times. These geodesics may be referred to as the "first geodesics" for ease of reference. In some cases, the geodesics do not overlap due to the lack of complete elasticity of some objects such as a foot, but they may overlap for a hand.

[00124] At 930, the mobile device 130 finds the corresponding prominent features in H_1 and H_2 in the PO region of point cloud B.

[00125] At 932, the mobile device 130 computes geodesics from H_1 to H_2, where H_2 is re-arranged so that the geodesics connecting H_1 to H_2 do not overlap at intermediate times. These geodesics may be referred to as the "second geodesics" for ease of reference. The typical result is that the number of points in G_1 is the same as those in H_1.

[00126] At 940, the mobile device 130 builds a correspondence between G_1 and H_1 using a matching technique that matches G_1 and H_1 according to prominent features (e.g., high curvature), and distance to an element with such features in the PO region of each corresponding point cloud. The

number of points in G_2 and H_2 are typically the same; if not, throwing away points in G_1 and H_1 is needed to get the equality of both sets.

[00127] At 942, the mobile device 130 builds a correspondence between G_2 and H_2 using, for example, a matching technique that matches G_2 and H_2 according to the prominent features, and distance to an element with such features in the PO region of each corresponding point cloud.

[00128] At 950, the mobile device 130 deforms the geodesics obtained from G_1 to H_1, to the geodesics obtained from G_2 to H_2 using, for example, a surface-based deformation method, such as As Rigid As Possible (ARAP) deformation, mesh deformation, or Laplacian deformation. Deformations such as bending do not change the length of the geodesic on the surface, but change their first and second order information, whereas affine deformations such as stretching change the length of the geodesics.

[00129] The initial points may be specified by G_1 and G_2, and the final points specified by H_1 and H_2. The optimal rigid transformations may be performed using ARAP (see, e.g., Sorkine, O., and Alexa, M (2007), As-rigid-as-possible surface modeling, In Proc, SGP, 109–116), from the initial to the final points. Note that by the correspondences found between G_1 and H_1 (or G_2 and H_1), it can be determined that G_1 and H_1 (or possibly a subset thereof) are a collection of fixed point. For example, one can run "overlap ICP" to align G_1 and H_1, and use G_2 and H_2 as initial points and final points, respectively, for the ARAP deformation method.

[00130] In at least one implementation, the fixed points are deduced from the geometry of the surface (i.e., they are intrinsic to the surface, as opposed to pre-specified by the user). ARAP is used to stitch the point clouds, as opposed to deform the final stitched point clouds. This can be interpreted as an inverse problem to that in the ARAP deformation method.

[00131] At 960, the mobile device 130 stitches the point clouds using, for example, an image alignment technique such as ICP.

[00132] At 962, the mobile device 130 calculates the deformation required to transform the geodesics from G_1 to G_2 to the geodesics from H_1 to H_2 using, for example, the surface-based deformation method.

[00133] In at least one implementation, the mobile device 130 picks a particular geodesic connecting one point from G_1, call it x_1, to another point in G_2, call it x_2. It denotes the geodesic connecting these points by $g(t)$ for $t \in [0,1]$. It denotes $v_1 = \frac{d}{dt} g(t)|_{t=0}$, and $v_2 = \frac{d}{dt} g(t)|_{t=1}$. It uses the inner product to determine the tangent vectors that are perpendicular to g at x_1 and at x_2 (i.e., picks w_1 and w_2 that are perpendicular to v_1 and v_2, respectively). It then produces two more geodesics by traveling along the surface with the directions w_1 and w_2. It produces two more points by following these geodesics for ϵ time, call them x_3, x_4. Then it can produce a geodesic that connects x_3 to x_4. This produces a parameterized surface Σ_1 on the point cloud A. It repeats the procedure above with points from H_1 and H_2 that match up with x_1 and x_2 to produce a similar surface Σ_2 on the point cloud B.

[00134] In at least one implementation, the mobile device 130 computes the first and the second fundamental forms of Σ_1, Σ_2. By Bonnet's theorem (see, e.g., Toponogov, Victor Andreevich (2006), Differential geometry of curves and surfaces, Boston, MA: Birkhäuser Boston, Inc., p. 132, ISBN 978-0-8176-4384-3, MR 2208981), the first and second fundamental forms of a surface classify the surface uniquely up to rigid transformations. It uses this to match up the surfaces $\Sigma_1$and Σ_2. If the fundamental forms of these two surfaces are identical, then it is done. Otherwise, it uses "overlap ICP" as well as ARAP to modify the surfaces until their fundamental forms of surfaces Σ_1, and Σ_2 match up completely.

[00135] In at least one implementation, the mobile device 130 repeats this process over all combinations in the points in G_1, G_2, and H_1, H_2. Since the fundamental forms are modified in every step until they match up, and there are finitely many points in G_1, G_2, H_1, H_2, it results in the rigid transformations (derived from overlap ICP) that take the respective parameterized surfaces

point cloud A to point cloud B. Once the fundamental forms on all patches that are in correspondence match up, the process terminates, and it ends up with point clouds that are related to one another by a rigid transformation. By reverting this process, it has a transformation that takes two point clouds that are deformed versions of one another, and produces a transformation that aligns one of the point clouds to the other point cloud.

[00136] At 964, the mobile device 130 applies the deformation to the whole point cloud B to deform it to match up to point cloud A.

[00137] In at least one embodiment, method 900 iterates from 920 to 964 for all the images in the point clouds. Temporal information and orientation of the object may be used to identify exactly how each pair of photos is related. The ranking of points based on features such as similar curvature in each point cloud may aid with loop closure.

[00138] Referring now to FIG. 10, shown therein is a flow chart of an example embodiment of a method 1000 of fitting a 3D model item over an object. Method 1000 may be carried out, for example, by mobile device 130, processor unit 134, and/or a remote computer or server.

[00139] At 1010, the mobile device 130 receives a model point cloud and an object point cloud. The model point cloud may be a 3D model that may include the interior of an item (e.g., shoe, glove, pillow case) to be fitted to an object of interest (e.g., foot, hand, pillow). For some use cases, the object may have different weight loadings.

[00140] At 1020, the mobile device 130 aligns a 3D item and an object of interest (may have been stitched), both represented with geometries (e.g., point clouds or meshes). The alignment may be done by finding the interior of the item to be fitted, aligning the object and the interior of the item with an initial transformation, and generating a final fitting of the interior of the item and the object of interest. FIG. 11A shows an example of a point cloud of the 3D model of a shoe (the item to be fitted). FIG. 11B shows an example of a

point cloud of the interior of the shoe aligned with a foot (the object of interest).

[00141] In at least one embodiment, the mobile device 130 aligns the stitched object inside a 3D item, such as a shoe (see e.g., FIG. 11A and FIG. 11B) in accordance with the following pseudocode representation:

[00142] Input 1: Model point cloud of the shoe

Input 2: Stitched object point cloud

Make sure the object and the item have the same scale so there is a sense of comparison.

Otherwise, rescale one object relative to the other.

As an example, find the center of mass for both objects to align the objects. The center of mass can be defined as the average of the x,y,z of all the points in the point cloud for each object.

Subtract the center of mass from each point in the point cloud of the first object.

Repeat for the second object.

Apply an alignment procedure, comprising:

Transformation and perturbation of coordinates of the stitched object that needs to be fitted into the model item.

Optimization according to an objective function resulting from the interaction between the two objects.

Extraction of the surface heat map of the interaction between the fitted object and the item it fits into (see e.g., FIG. 12).

End.

[00143] FIG. 11A shows an example of a point cloud of a 3D model of a shoe 1110. FIG. 11B shows an example of a point cloud of the interior of the shoe 1120 aligned with a foot. FIGS. 11A-11B show examples of the types of point clouds that method 1000 may use for extracting a heat map.

[00144] At 1030, the mobile device 130 extracts a heat map from the interaction of the object with the interior of the item to be fitted. This may, for example, try to minimize the amount of collision between the item and the object. FIG. 12 shows an example of a heat map 1200 of a foot interacting with the surface of the inside of a shoe.

[00145] For example, FIG. 12, when viewed in color, may provide a heat map 1200 where the degree of interaction between the object with the interior of the item to be fitted may be represented by a range of colors such as blue 1210, green 1220, yellow 1230, orange 1240, and red 1250, where the "cold" colors (e.g., blue and green) represent the least amount of interaction, the "warm" colors (e.g., yellow) represent a middle amount of interaction, and the "hottest" colors (e.g., orange and red) represent the most interaction, while grey 1260 represents no interaction. When viewed in greyscale, the heat map 1200 may be represented, for example, by a darkest shade of grey (analogous to blue) to a lightest shade of grey (analogous to red).

[00146] At 1040, the mobile device 130 uses the points with the highest heat value (e.g., represented with reddish colors) to determine areas of interest that may need some stretching. Such areas of interest may include areas in the item that are too close to the object that it fits to.

[00147] Method 1000 can be adapted to provide a heat map for various possible interactions between an item and another item to determine those areas of interest needed for further processing such as, but not limited to, stretching. In the different possible interactions, method 1000 may apply to self-measurement or measurements by another. In the case of self-measurement, method 1000 may be carried out, for example, within the privacy of a user's home. While primarily described with reference to the mobile device 130, it should be understood that method 1000 may be performed by any device(s) and/or processor(s) in a mobile, static, or hybrid of mobile-static configuration.

[00148] As an example, for the adjustment of a ski boot, a heat map can be applied to the interaction between a body part and an item to determine those

areas of interest that need stretching. The ski boot use case is an example of measuring the form of a foot to fit into a ski boot and determine the areas that need adjustment. Similar use cases include any sporting gear or equipment that requires sizing in relation to the size of a body part.

[00149] As an example, for the digital sizing of a bra, a heat map can be applied to the interaction between a body part such as breasts and an item such as a bra to determine those areas of interest that need stretching to determine optimal sizing.

[00150] As an example, for the digital sizing of a prosthetic, cast, walking cast, or any splint care, a heat map can be applied to the interaction between a body part that requires an artificial body part or supporting object in nearly any shape, size, or configuration to provide support to a body part in order to determine those areas of interest that need stretching to determine optimal sizing.

[00151] As an example, for the digital sizing of an oxygen mask, a heat map can be applied to the interaction between a face and the item (i.e., the oxygen mask in this instance) to ensure a perfect seal between the face of a person and the oxygen mask.

[00152] As an example, for the digital sizing of jewelry or accessories (e.g., rings, headsets, bracelets, necklaces, watches and their straps, and the like), a heat map can be applied to the interaction between a body part and an item to determine those areas of interest that need stretching to perform same method or process disclosed herein in order to determine the optimal sizing.

[00153] As an example, for the digital sizing of an article of clothing, a heat map can be applied to the interaction between a body part and an item to determine those areas of interest that need stretching to determine optimal sizing.

[00154] More generally, any deformable object stitching illustrates the modifications or deformation of the original form of an object. Method 1000 may, for example, be applied to self-assess or analyze a body part such as a

human's nose to allow illustrations of a range of modifications or deformations or departure from the original form. Alternatively, or in addition, method 1000 may apply a heat map to the interaction of a virtual representation of a body part to an item, to the interaction of the body part to a virtual representation of the item, or to the interaction of a virtual representation of a body part to a virtual representation of the item, such as in virtual reality (VR) or augmented reality (AR).

[00155] At 1050, the mobile device 130 uses the item's material properties (e.g., from material information provided by the manufacturer of the item) to determine the elasticity of areas of interest. The material information may include, but is not limited to, yield strength, tensile strength, yield point, fracture point, and material fatigue. A global consistency check may be run to make sure the item to fit has a reasonable shape with respect to the object that it fits to or into, or to ensure the well-being of the object by avoiding among other issues, breakage or permanent deformation of materials due to excessive stretching.

[00156] In at least one implementation, 1040 and 1050 are repeated (e.g., in a program loop) to better fit the item to the object of interest. If the item's material has stretched within its elasticity bounds and it accurately fits to the object of interest, the iteration of 1040 and 1050 may then stop. Otherwise, other item sizes can be tried depending on the use case.

[00157] The information may then be used directly for, among others, visualization purposes, or as input for further processing to, among other use cases, find the correct size for the object to fit to (e.g., shoe size).

[00158] At 1060, the mobile device 130 displays the heat map (e.g., to the person being fitted), and uses the elastic material information to suggest by how much the item will stretch at the points of interest. FIG. 13 shows an example of a heat map of a foot with the suggested modifications based on the elasticity of the shoe while worn.

[00159] For example, FIG. 13, when viewed in color, may provide a heat map 1300 where the degree of interaction between the object with the interior of the item to be worn may be represented by a range of colors such as blue 1310, green 1320, yellow 1330, orange 1340, and red 1350, where the "cool" colors (e.g., blue and green) represent the least amount of interaction, the "warm" colors (e.g., yellow) represent a middle amount of interaction, and the "hottest" colors (e.g., orange and red) represent the most interaction, while grey 1360 represents no interaction

[00160] In at least one embodiment, the mobile device 130 uses machine learning for at least one of the operations performed during method 200.

[00161] In at least one embodiment, the mobile device 130 uses machine learning for at least one of the operations performed during method 900.

[00162] In at least one embodiment, the mobile device 130 uses machine learning for at least one of the operations performed during method 1000.

[00163] In at least one embodiment, the system 100 uses machine learning, whether it be for optimization purposes or for extensions that rely on, or are related to, one or more of method 200, method 900, method 1000, any steps thereof, alone or in combination, as well as any other operations or processes (or parts thereof) described herein. Alternatively, or in addition, any extension of the methods, operations, or processes (or parts thereof) may benefit from data-driven strategies.

[00164] While the applicant's teachings described herein are in conjunction with various embodiments for illustrative purposes, it is not intended that the applicant's teachings be limited to such embodiments as the embodiments described herein are intended to be examples. On the contrary, the applicant's teachings described and illustrated herein encompass various alternatives, modifications, and equivalents, without departing from the embodiments described herein, the general scope of which is defined in the appended claims.

CLAIMS:

1. A computer-implemented method for three-dimensional scanning and measurement comprising:

receiving a plurality of images of an object, the plurality of images providing views of an object from at least two angles;

preprocessing the plurality of images using morphological refinement;

creating a source point cloud based at least in part on the plurality of images;

removing outliers from the source point cloud;

globally registering the source point cloud, thereby creating a globally registered source point cloud;

generating a transformed source point cloud based at least in part on the globally registered source point cloud;

comparing the transformed source point cloud with a target point cloud, thereby creating a point cloud comparison;

generating a stitched point cloud based at least in part on the point cloud comparison, thereby creating a resulting stitched 3D model;

measuring the resulting stitched 3D model; and

outputting the resulting stitched 3D model in a format capable of being displayed.

2. The computer-implemented method of claim 1, further comprising inputting the source point cloud into a neural network configured for parameter optimization based at least in part on one of a significant amount of overlap between the stitched point cloud or labeled correspondences.

3. The computer-implemented method of claim 1, wherein the morphological refinement causes morphological changes by at least one of an erosion of points in a 3D model, noise removal, or edge refinement.

4. The computer-implemented method of claim 3, wherein the 3D model is a 3D point cloud.

5. The computer-implemented method of claim 3, wherein the morphological refinement is based at least in part on at least one of filtering masks or kernels.

6. The computer-implemented method of claim 1, further comprising inputting the source point cloud into a neural network, the source point cloud having a surface with a discontinuity on the surface and a radius for each point around the discontinuity, the neural network configured to output an optimal radius for each point around the discontinuity that corresponds to areas to be removed from the source point cloud.

7. The computer-implemented method of claim 1, wherein creating the source point cloud is based at least in part on creating a geometrical representation of the plurality of images using spatial information.

8. The computer-implemented method of claim 7, wherein the spatial information is based at least in part on camera intrinsics of a camera that scanned the plurality of images.

9. The computer-implemented method of claim 1, wherein removing outliers comprises 2D or 3D processing that accentuates areas of interest of the object.

10. The computer-implemented method of claim 1, wherein globally registering the source point cloud comprises aligning 3D assets using geometrically relevant features, and generating a transformed source asset from the aligned 3D assets.

11. The computer-implemented method of claim 1, wherein comparing the transformed source point cloud with a target point cloud comprises iterative closes point (ICP) stitching to generate the stitched point cloud.

12. The computer-implemented method of claim 9, wherein the ICP stitching comprises an overlap step.

13. The computer-implemented method of claim 1, wherein measuring the resulting stitched 3D model comprises using a distance function.

14. A computer-implemented method for deformable object stitching comprising:

receiving a first point cloud and a second point cloud having a partial overlap;

finding first prominent features in a subregion G_1 and a subregion G_2 in a first region of the partial overlap over the first point cloud;

computing first geodesics based on the first prominent features;

finding second prominent features in a subregion H_1 and a subregion H_2 in a second region of the partial overlap over the second point cloud;

computing second geodesics based on the second prominent features;

building a first correspondence between the subregion G_1 and the subregion H_1;

building a second correspondence between the subregion G_2 and the subregion H_2;

deforming the first geodesics and the second geodesics based on the first correspondence and the second correspondence;

stitching the first point cloud and the second point cloud, thereby generating a stitched point cloud;

calculating a required deformation to transform the first geodesics to the second geodesics;

applying the required deformation to the second point cloud to match the second point cloud to the first point cloud, thereby refining the stitched point cloud; and

outputting the refined stitched point cloud in a format capable of being displayed.

15. The computer-implemented method of claim 12, wherein finding first prominent features comprises estimating a curvature in subregion G_1 or subregion G_2, then refining the estimated curvature by iteratively calculating the curvature over one or more iterations, such that a region of high curvature

is re-calculated by considering a smaller neighborhood in the region, and calculating the curvature based on the smaller neighborhood, repeating the iterations until sampling in a smaller neighborhood does not increase the curvature in the region by more than a specified threshold.

16. The computer-implemented method of claim 12, wherein finding second prominent features comprises estimating a curvature in subregion H_1 or subregion H_2, then refining the estimated curvature by iteratively calculating the curvature over one or more iterations, such that a region of high curvature is re-calculated by considering a smaller neighborhood in the region, and calculating the curvature based on the smaller neighborhood, repeating the iterations until sampling in a smaller neighborhood does not increase the curvature in the region by more than a specified threshold.

17. The computer-implemented method of claim 12, wherein finding the first prominent features comprises approximating a curvature on subregion G_1 and subregion G_2 by discretizing a curvature operator, and extracting the first prominent features with respect to the curvature where the curvature is high or zero, a high curvature representing sharp corners and a zero curvature representing a flat region.

18. The computer-implemented method of claim 12, wherein finding the second prominent features comprises approximating a curvature on subregion H_1 and subregion H_2 by discretizing a curvature operator, and extracting the second prominent features with respect to the curvature where the curvature is high or zero, a high curvature representing sharp corners and a zero curvature representing a flat region.

19. The computer-implemented method of claim 12, wherein computing the first geodesics and computing the second geodesics comprises triangulating on respective subregions and determining a shortest path between two points on a triangulated surface.

20. The computer-implemented method of claim 12, wherein building the first correspondence and building the second correspondence comprises ranking the first prominent features and the second prominent features, respectively, by curvature in the first point cloud and the second point cloud, respectively, and matching points therebetween.

21. The computer-implemented method of claim 12, wherein calculating the required deformation comprises producing a surface determined by connecting any two points between subregion G_1 and subregion H_1, producing a geodesic, and extending the geodesic into a surface by considering a small neighborhood about the geodesic.

22. The computer-implemented method of claim 12, wherein calculating the required deformation further comprises estimating a first fundamental form of the surface by finite difference methods, and deforming the surface by an As Rigid As Possible (ARAP) deformation using the first correspondence and the second correspondence.

23. The computer-implemented method of claim 20, wherein calculating the required deformation further comprises determining a similarity between deformed subregions by estimating a second fundamental form for each of the subregions, and comparing a coefficient between the first fundamental form and the second fundamental form.

24. A computer-implemented method for fitting a first 3D model of an item over a second 3D model of an object comprising:

receiving a model point cloud of the item and an object point cloud of the object;

aligning the object point cloud inside the model point cloud;

extracting a surface heat map based at least in part on an interaction of the object point cloud with the model point cloud, the surface heat map having highest heat values;

determining a corresponding region on the model point cloud based at least in part on the highest heat values;

determining an elasticity of the corresponding region based at least in part on material properties of the item; and

displaying an enhanced surface heat map based at least in part on the surface heat map and the elasticity of the corresponding region.

25. The computer-implemented method of claim 22, wherein determining the elasticity of the corresponding region is based at least in part on a predetermined set of weights, where each element in the set corresponds to the elasticity of a particular physical material of the object.

26. A system for providing three-dimensional scanning and measurement comprising a data store and at least one processor coupled to the data store, the data store comprising a non-transient computer-readable storage medium having stored thereon computer-executable instructions for execution by the processor to perform the method according to any one of claims 1 to 11.

27. A system for deformable object stitching comprising a data store and at least one processor coupled to the data store, the data store comprising a non-transient computer-readable storage medium having stored thereon computer-executable instructions for execution by the processor to perform the method according to any one of claims 12 to 21.

28. A system for fitting a first 3D model of an item over a second 3D model of an object comprising a data store and at least one processor coupled to the data store, the data store comprising a non-transient computer-readable storage medium having stored thereon computer-executable instructions for execution by the processor to perform the method according to any one of claims 22 to 23.

29. A computer-readable medium comprising a plurality of instructions that are executable on a processor of a system for adapting the system to implement a method for providing three-dimensional scanning and

measurement, wherein the method is defined according to any one of claims 1 to 11.

30. A computer-readable medium comprising a plurality of instructions that are executable on a processor of a system for adapting the system to implement a method for deformable object stitching, wherein the method is defined according to any one of claims 12 to 21.

31. A computer-readable medium comprising a plurality of instructions that are executable on a processor of a system for adapting the system to implement a method for fitting a first 3D model of an item over a second 3D model of an object, wherein the method is defined according to any one of claims 22 to 23.

32. Any and all features of novelty and inventiveness described, referred to, shown as examples, or otherwise described herein.

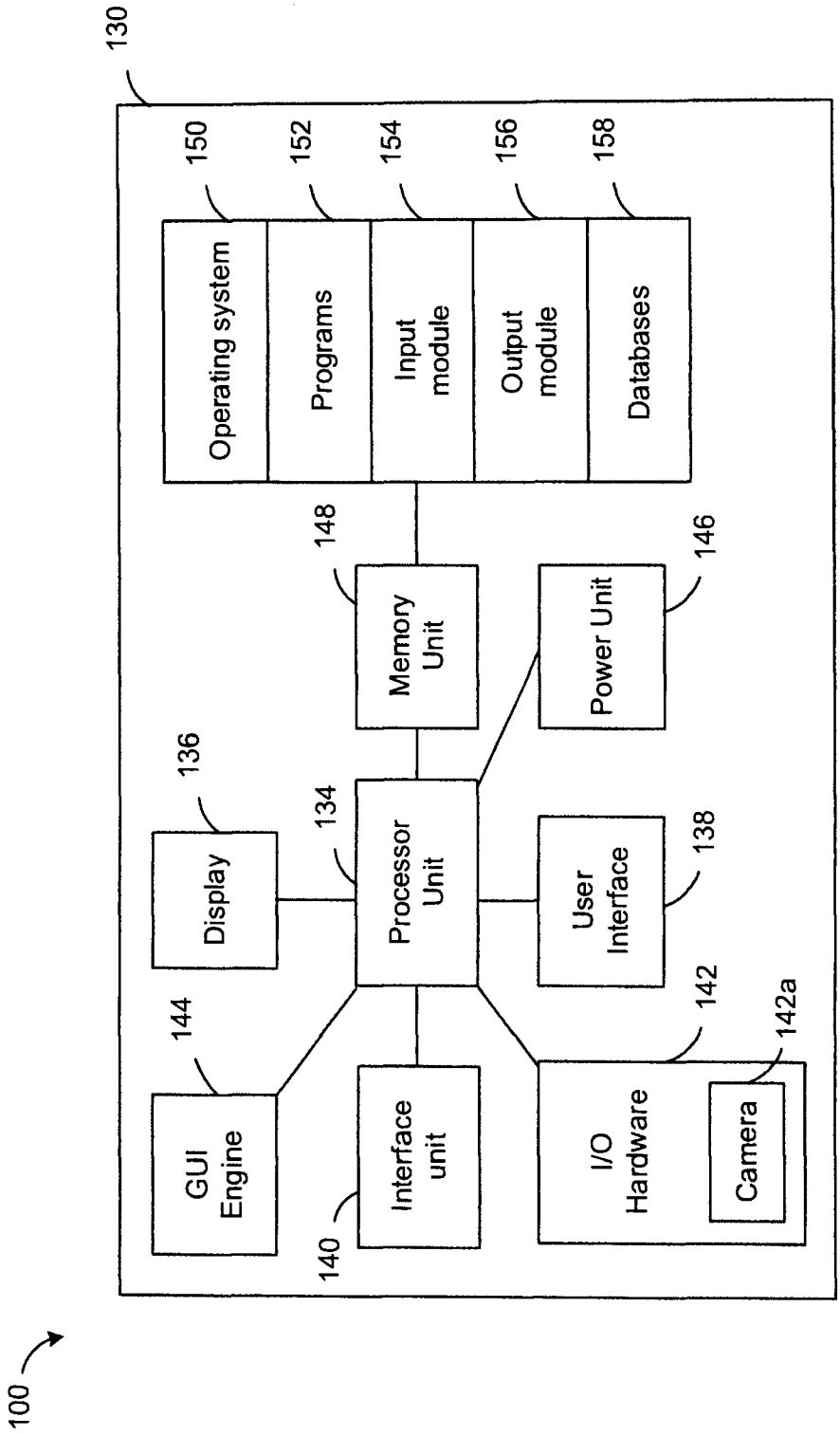

100
130
Operating system
150
Programs
152
Input module
154
Output module
156
Databases
158
Memory Unit
148
Power Unit
146
Display
136
Processor Unit
134
User Interface
138
GUI Engine
144
Interface unit
140
I/O Hardware
142
Camera
142a

FIG. 1

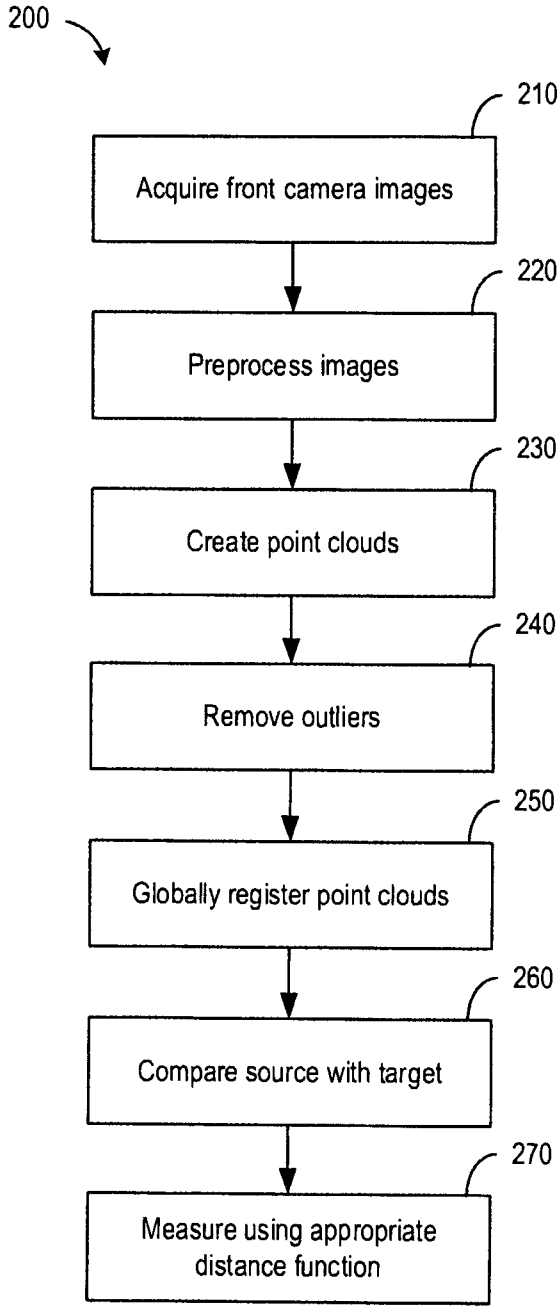

200
210
Acquire front camera images
220
Preprocess images
230
Create point clouds
240
Remove outliers
250
Globally register point clouds
260
Compare source with target
270
Measure using appropriate distance function

FIG. 2

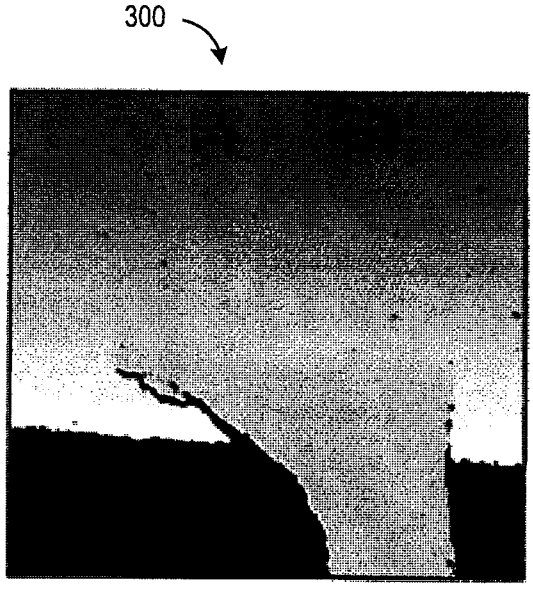


FIG. 3A

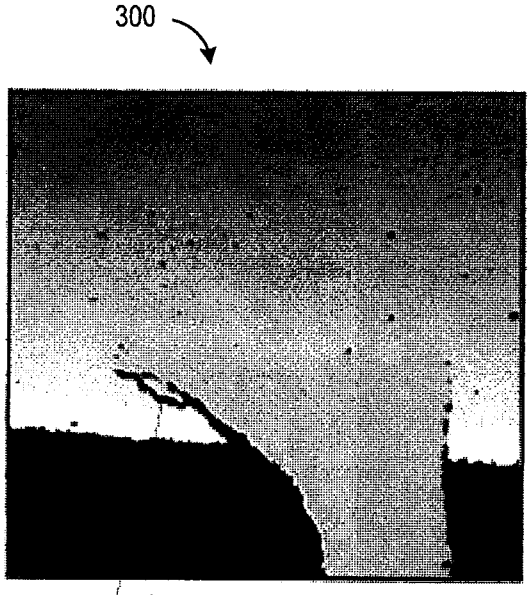


FIG. 3B

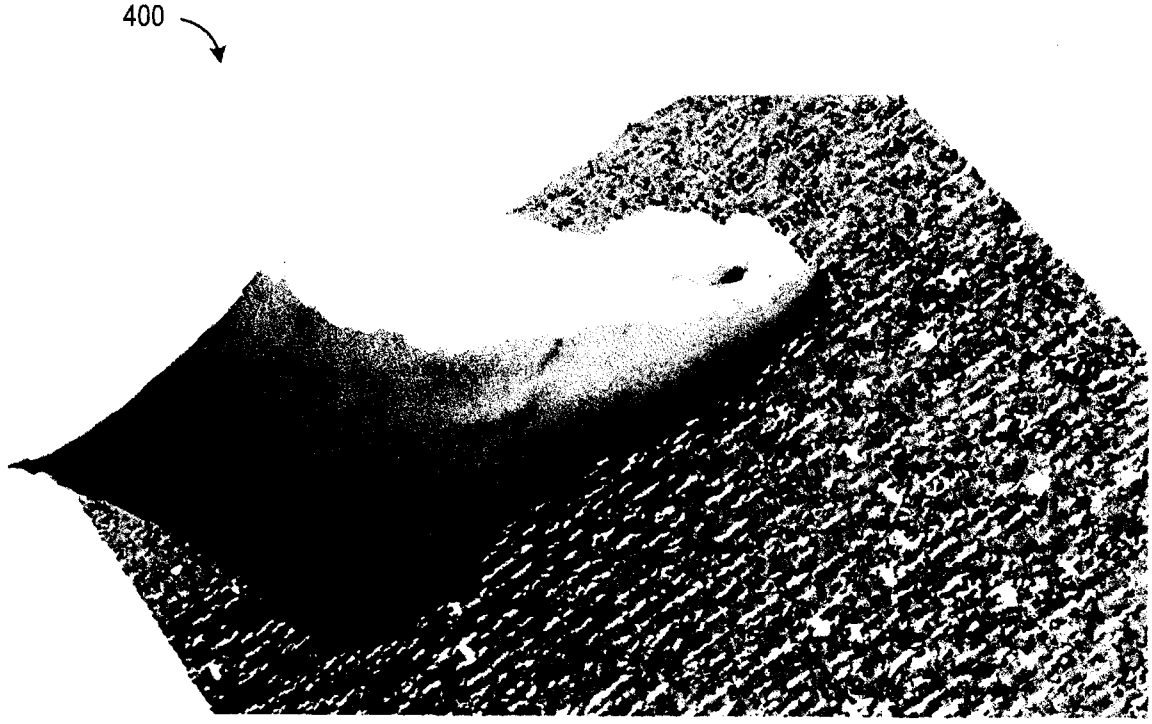


FIG. 4A

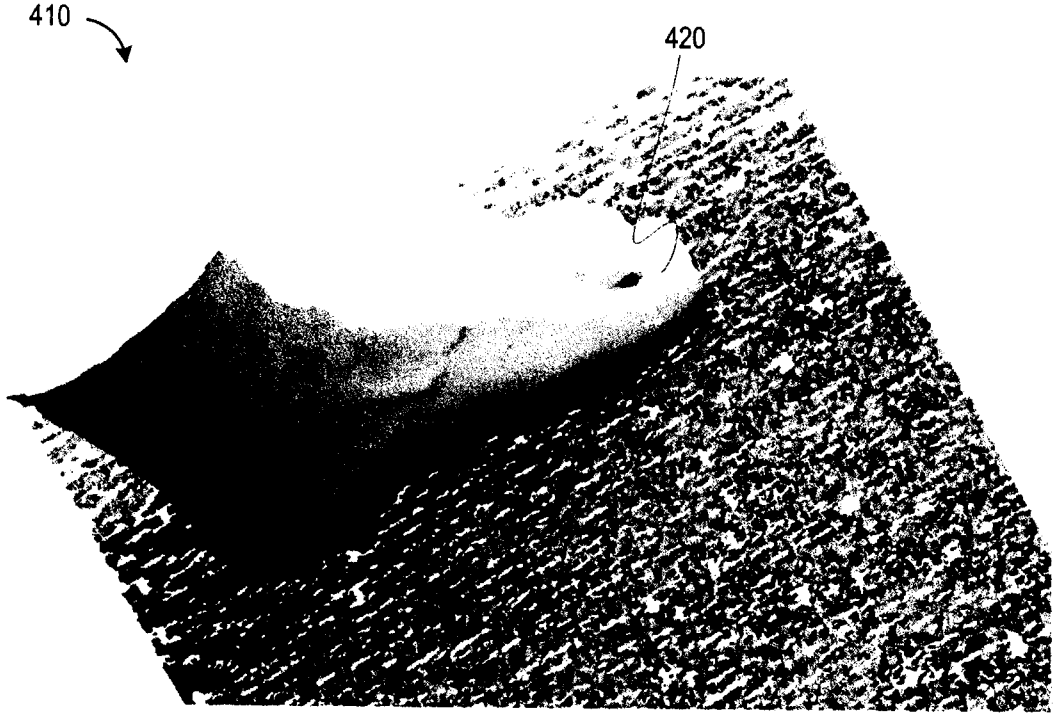


FIG. 4B

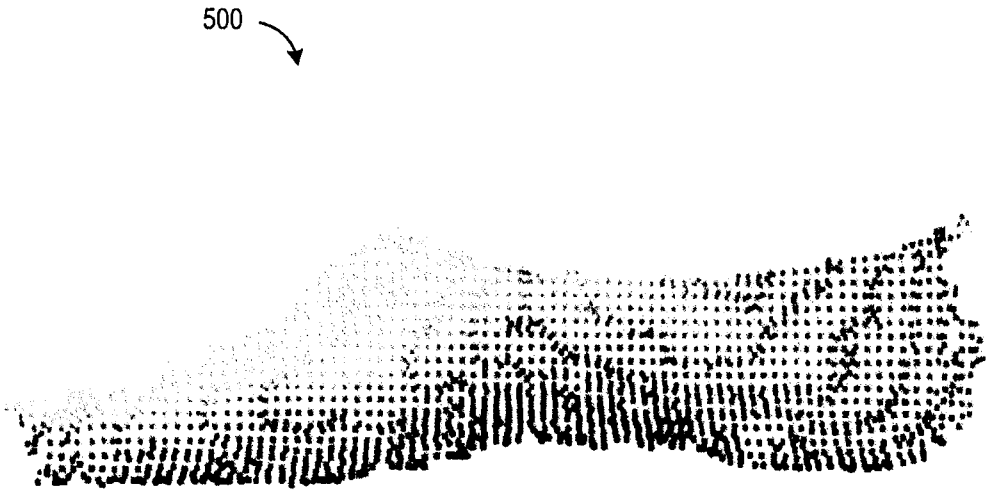


FIG. 5

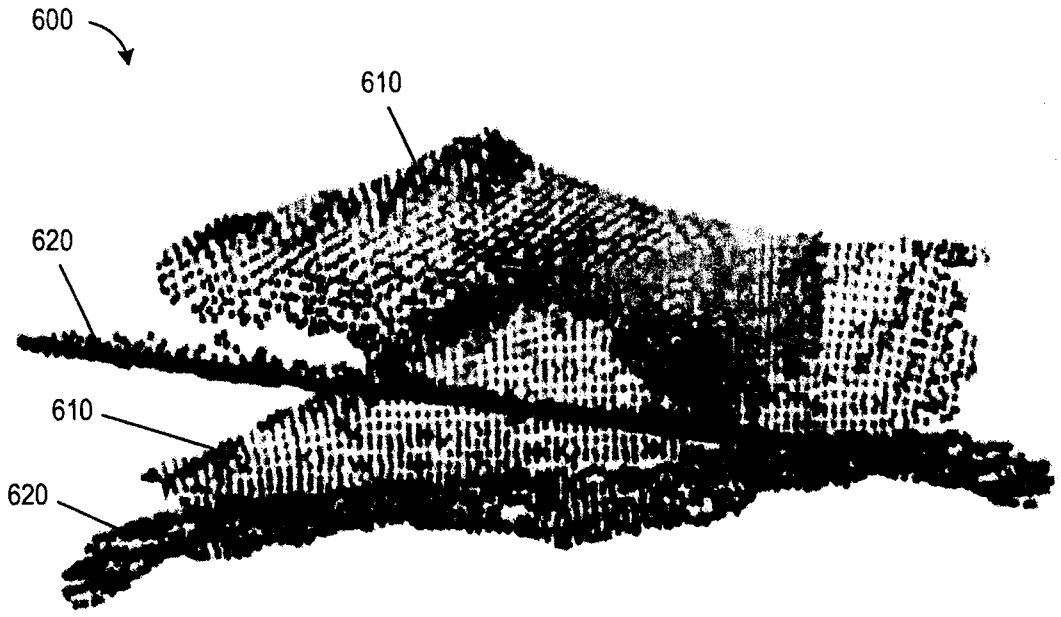


FIG. 6A

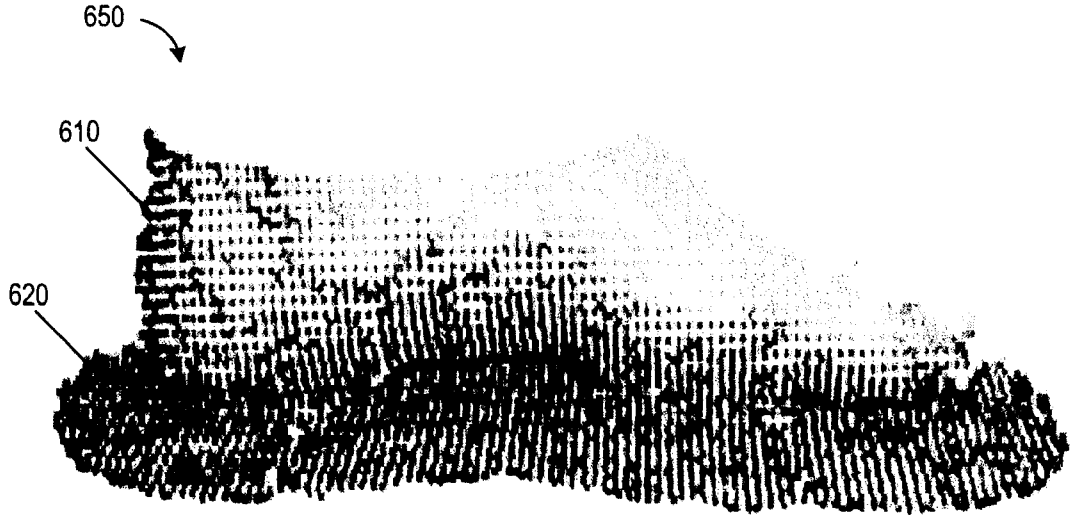


FIG. 6B

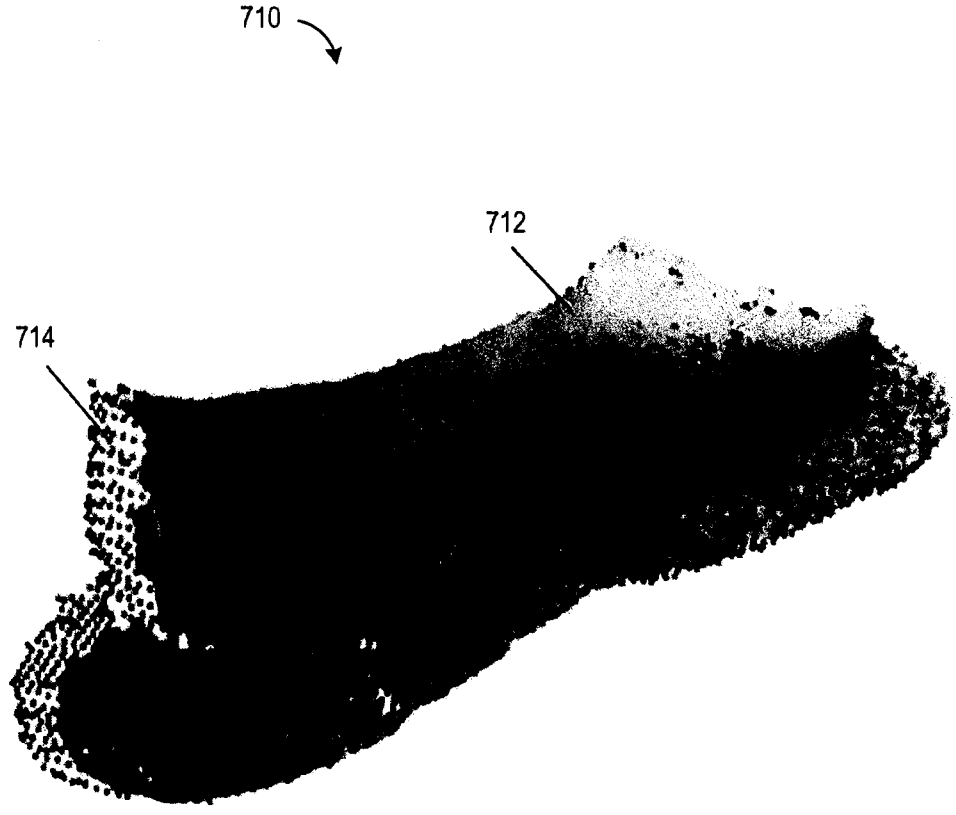


FIG. 7A

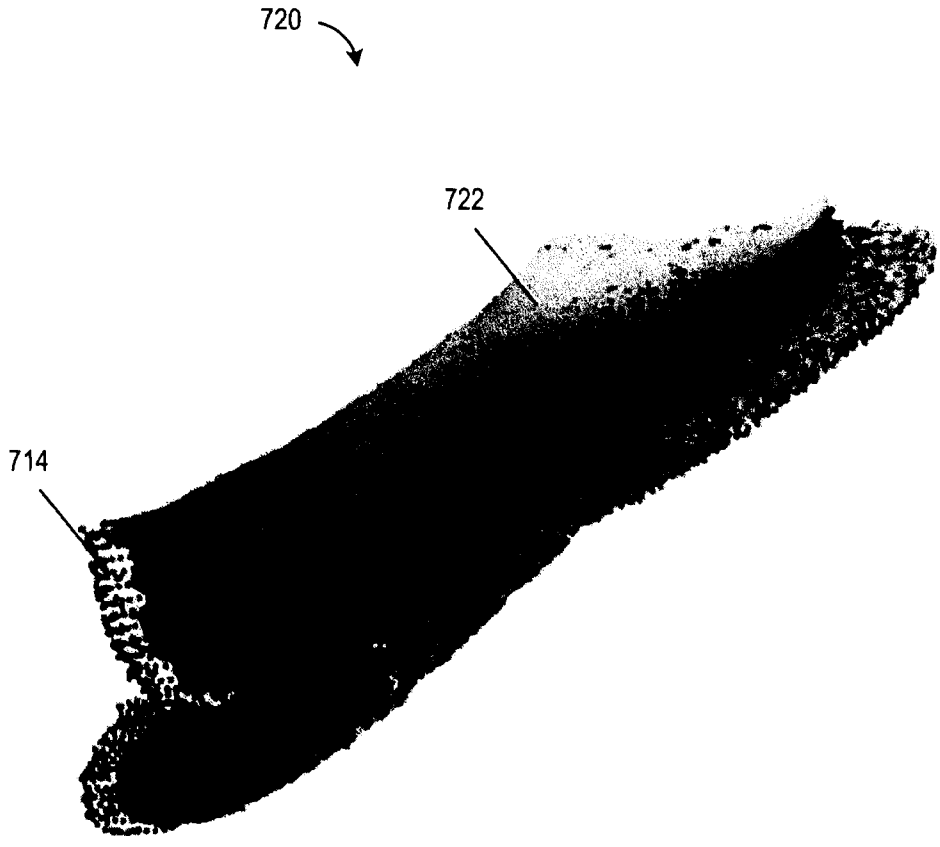


FIG. 7B

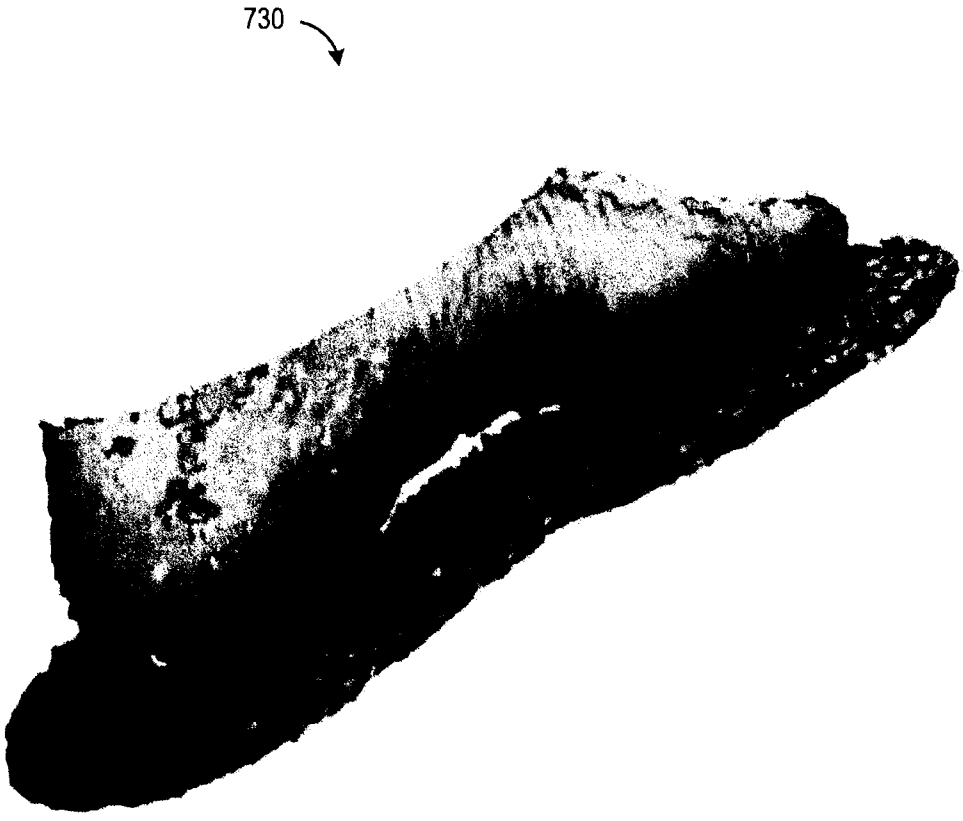


FIG. 7C

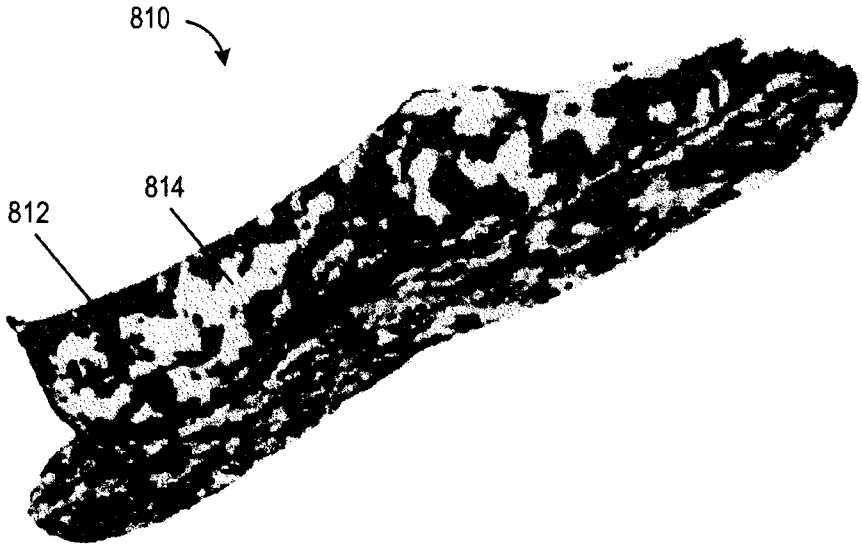


FIG. 8A

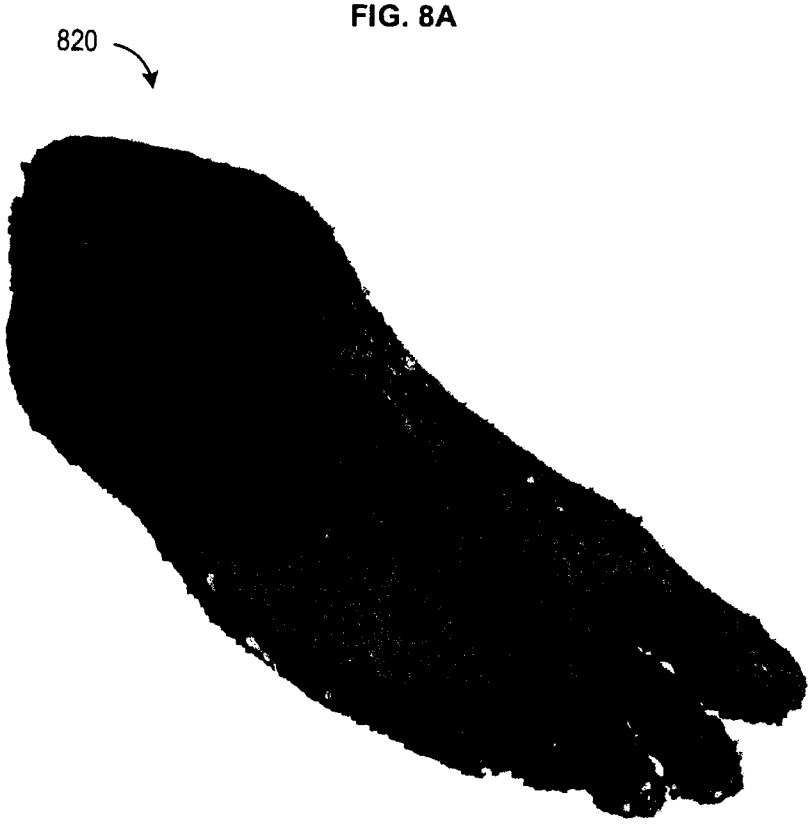


FIG. 8B

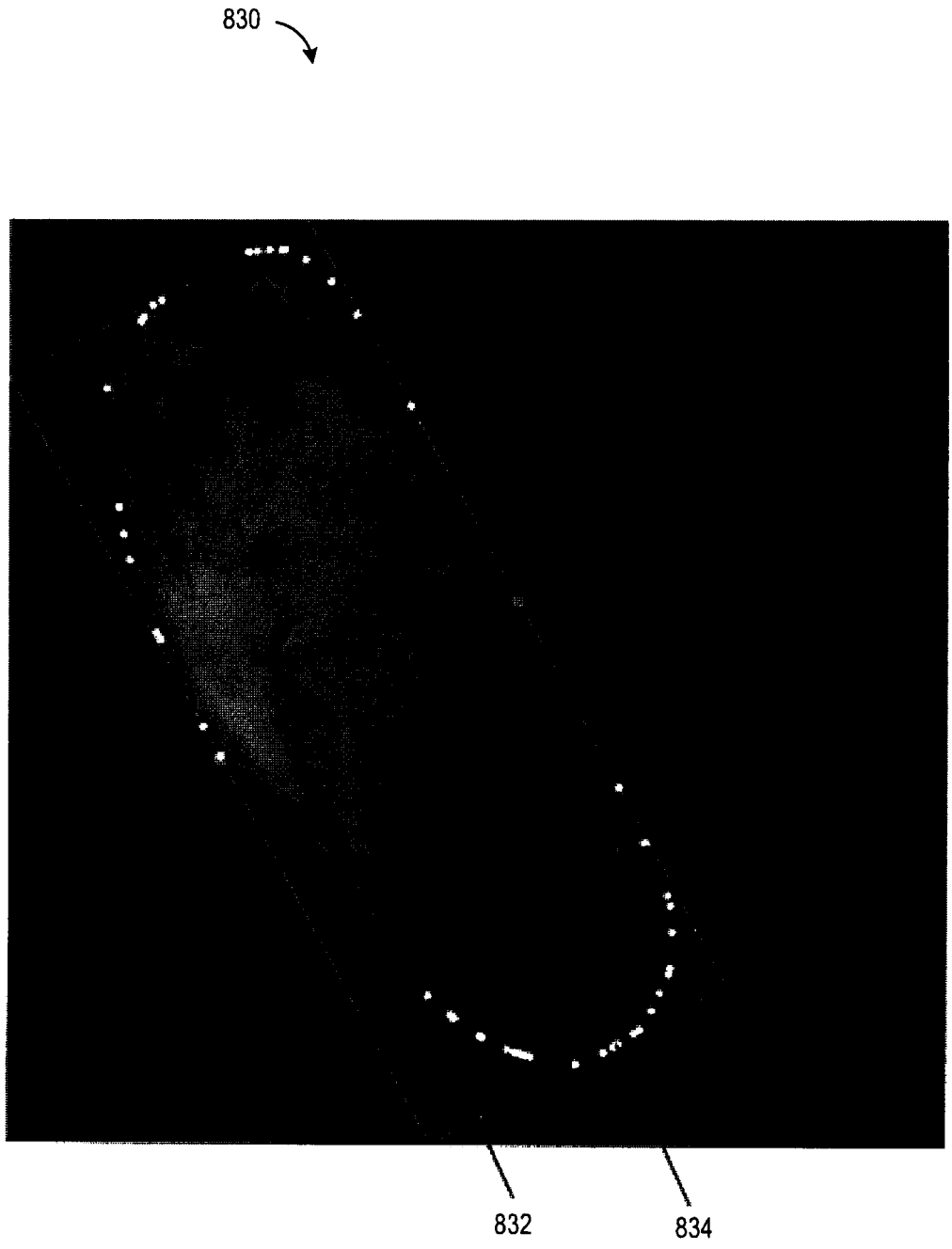


FIG. 8C

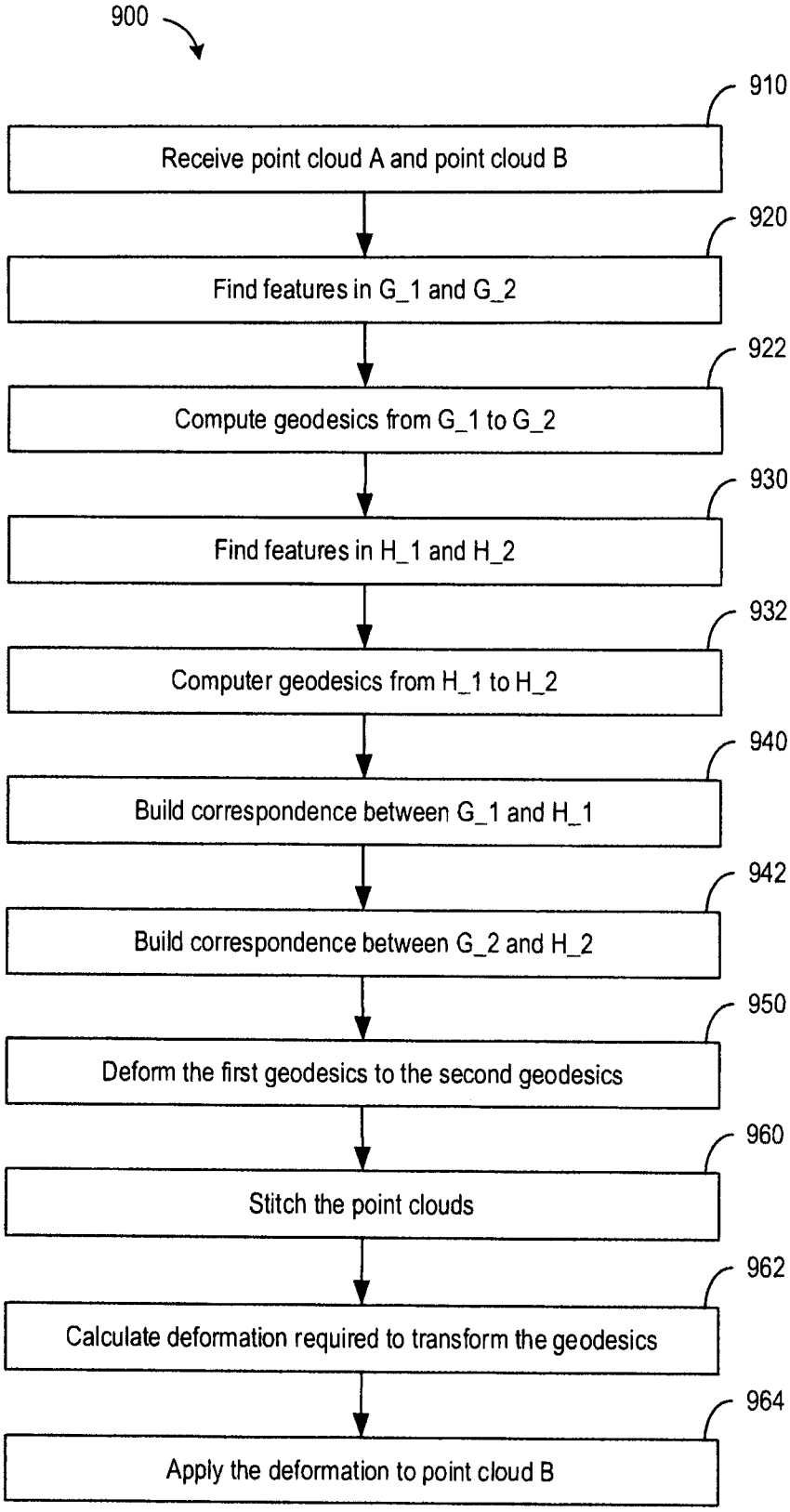

900
910
Receive point cloud A and point cloud B
920
Find features in G_1 and G_2
922
Compute geodesics from G_1 to G_2
930
Find features in H_1 and H_2
932
Computer geodesics from H_1 to H_2
940
Build correspondence between G_1 and H_1
942
Build correspondence between G_2 and H_2
950
Deform the first geodesics to the second geodesics
960
Stitch the point clouds
962
Calculate deformation required to transform the geodesics
964
Apply the deformation to point cloud B

FIG. 9

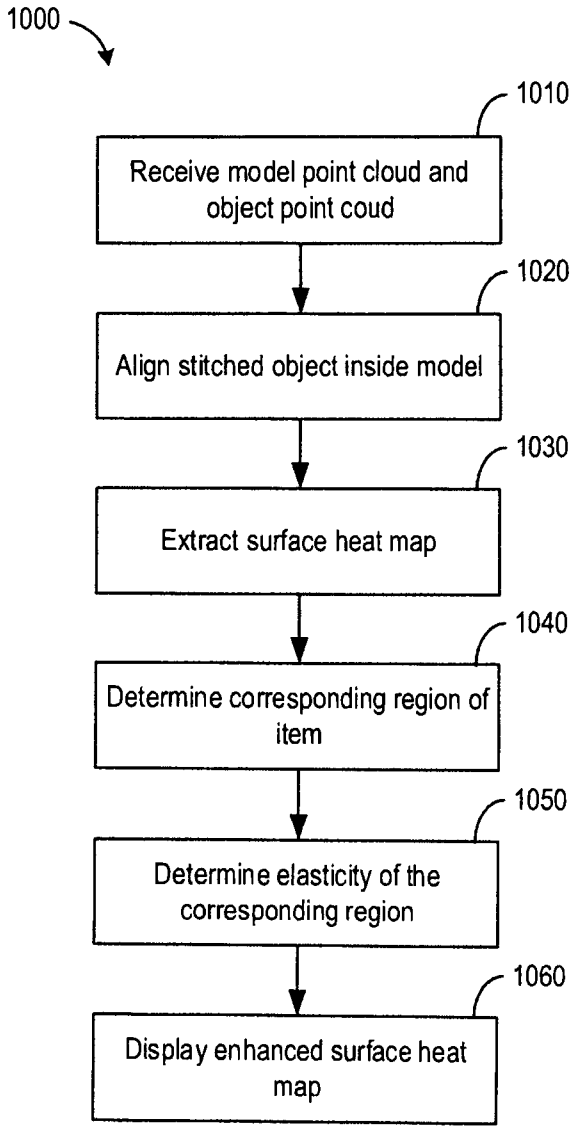

1000
1010
Receive model point cloud and object point coud
1020
Align stitched object inside model
1030
Extract surface heat map
1040
Determine corresponding region of item
1050
Determine elasticity of the corresponding region
1060
Display enhanced surface heat map

FIG. 10

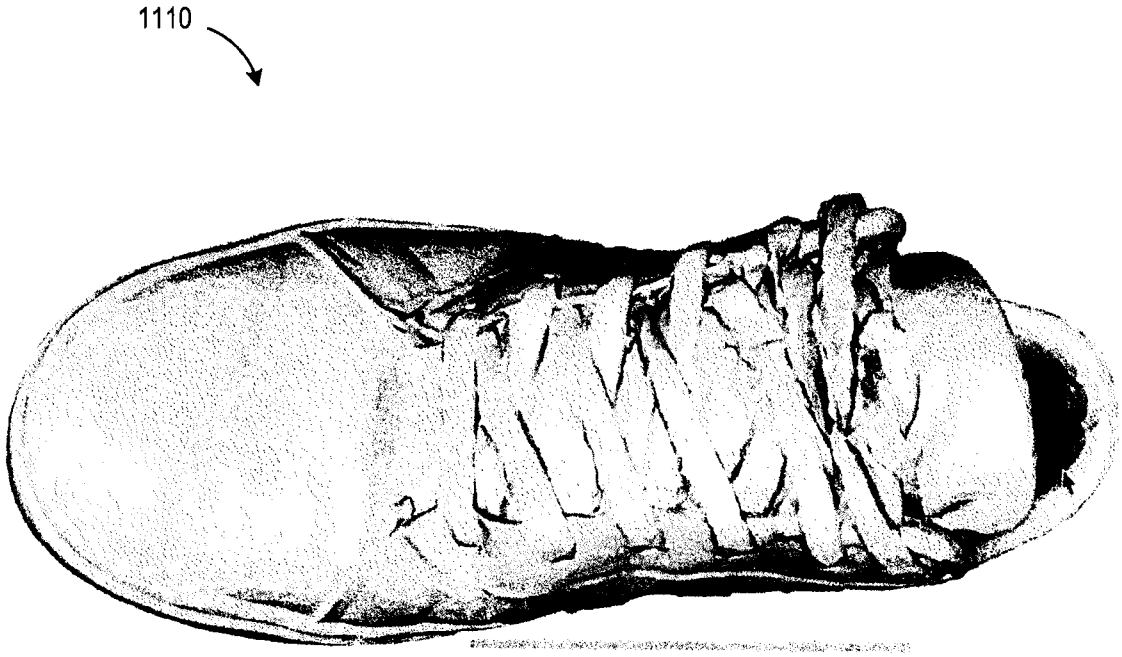


FIG. 11A

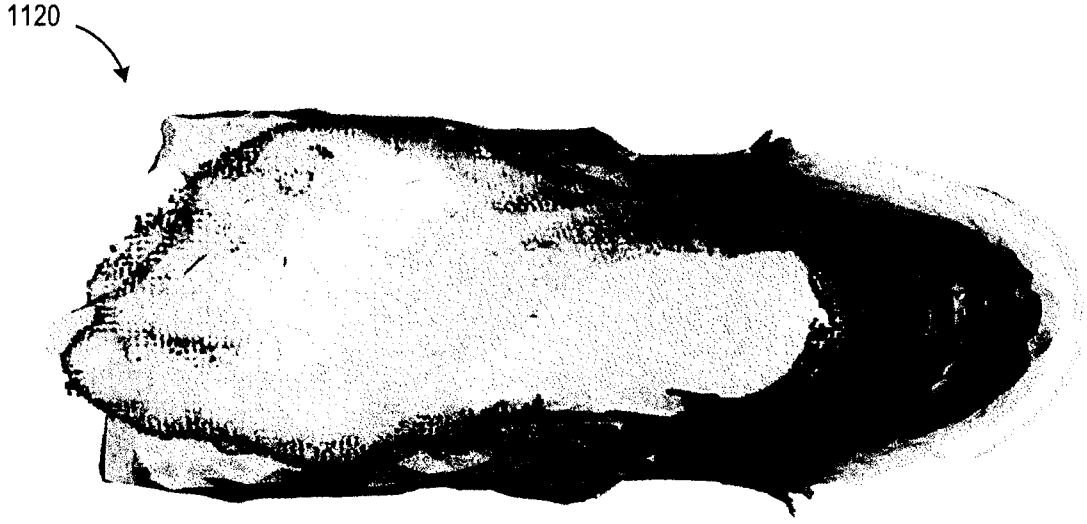


FIG. 11B

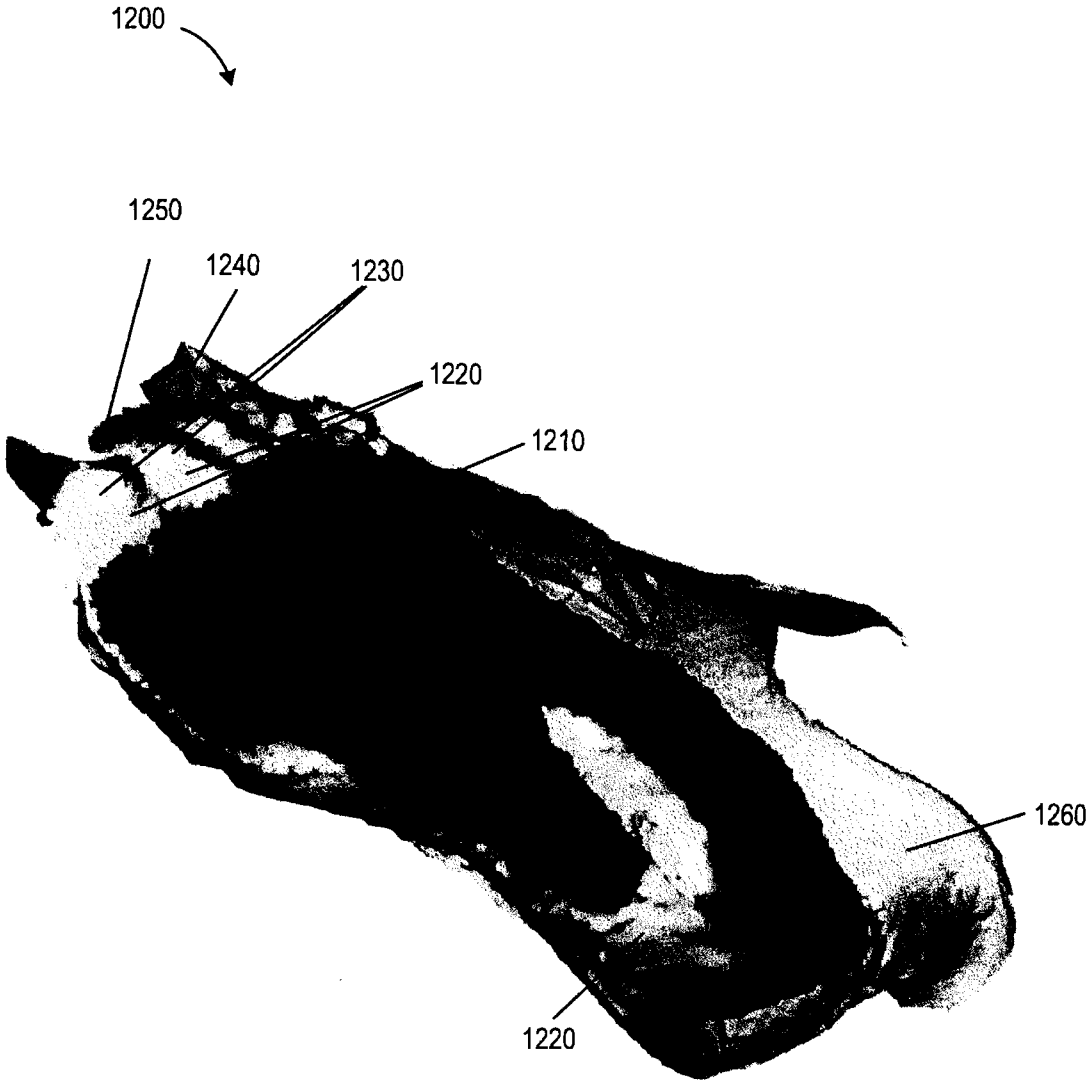


FIG. 12

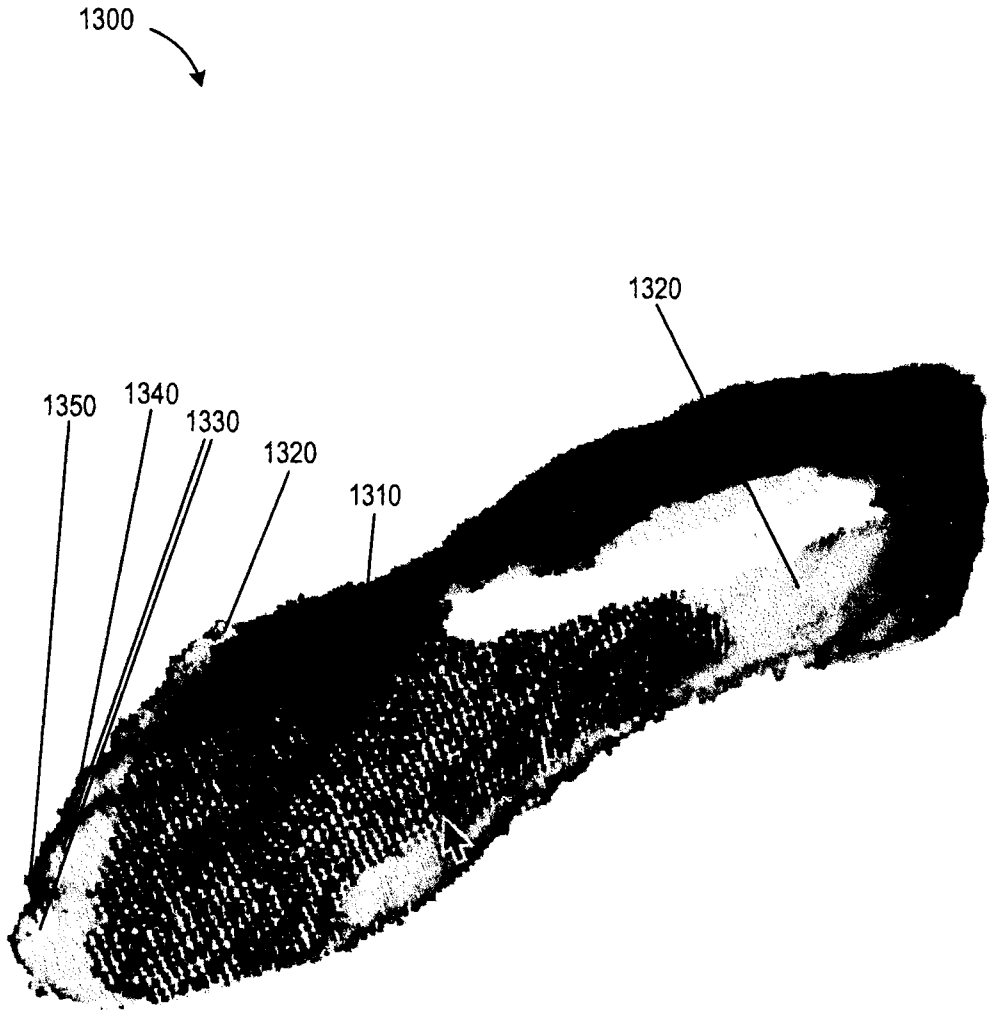


FIG. 13

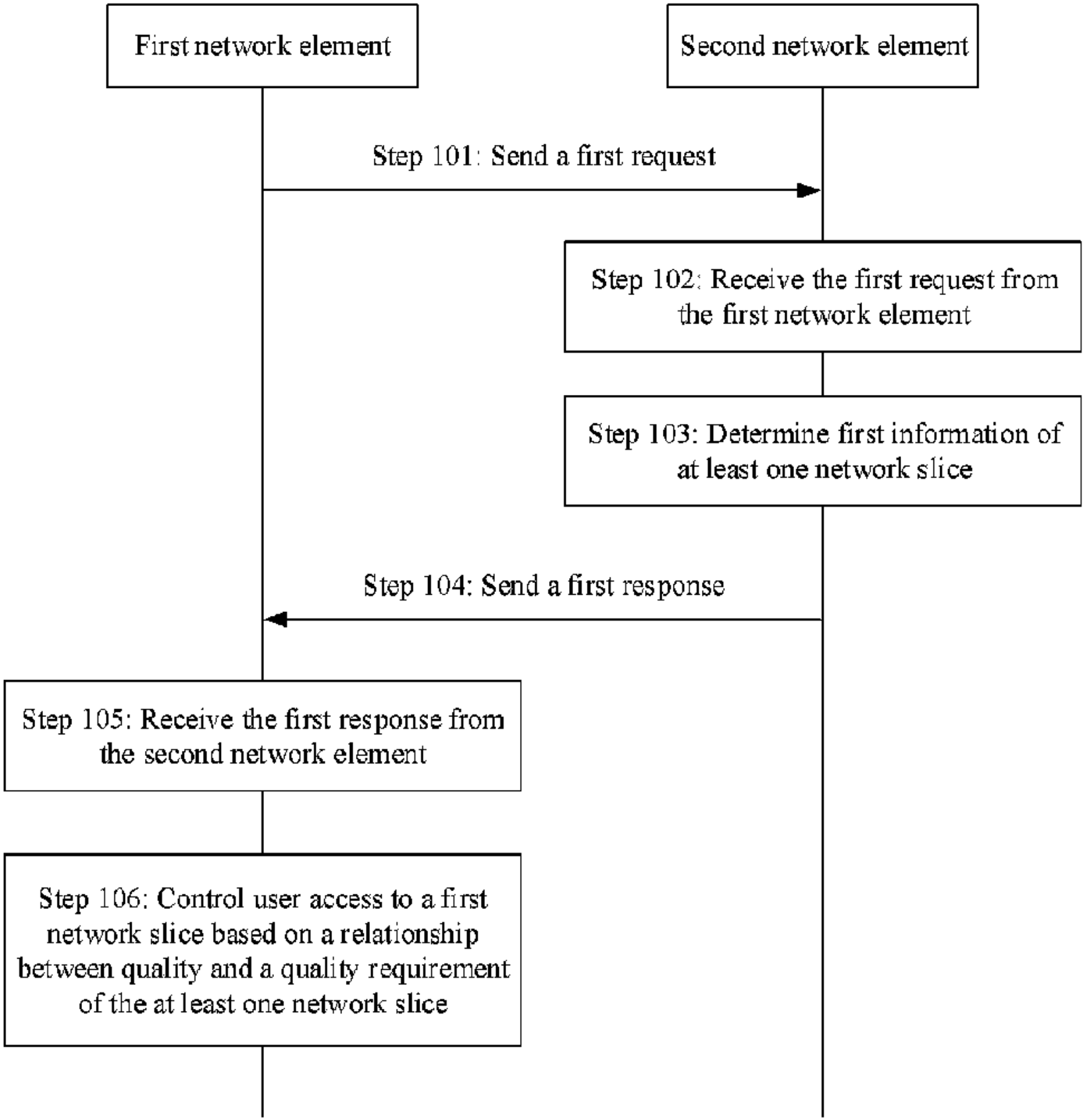

First network element
Second network element
Step 101: Send a first request
Step 102: Receive the first request from the first network element
Step 103: Determine first information of at least one network slice
Step 104: Send a first response
Step 105: Receive the first response from the second network element
Step 106: Control user access to a first network slice based on a relationship between quality and a quality requirement of the at least one network slice

FIG. 5