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

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
1 December 2016 (01.12.2016)

(10) International Publication Number
WO 2016/189372 A2

(51) International Patent Classification: Not classified

(21) International Application Number: PCT/IB2016/000530

(22) International Filing Date: 26 April 2016 (26.04.2016)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
62/152,883 25 April 2015 (25.04.2015) US

(72) Inventor; and
(71) Applicant : **QUAN, Xiao** [CN/CN]; Unit 315, Building 416, BUAA (BeiHang), 37 XueYuan Road, HaiDian District, Beijing, 100083 (CN).

(81) Designated States *(unless otherwise indicated, for every kind of national protection available)*: AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, 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).

Published:
— *without international search report and to be republished upon receipt of that report (Rule 48.2(g))*
— *the filing date of the international application is within two months from the date of expiration of the priority period (Rule 26bis.3)*

(54) Title: METHODS AND APPARATUS FOR HUMAN CENTRIC "HYPER UI FOR DEVICES"ARCHITECTURE THAT COULD SERVE AS AN INTEGRATION POINT WITH MULTIPLE TARGET/ENDPOINTS (DEVICES) AND RELATED METHODS/SYSTEM WITH DYNAMIC CONTEXT AWARE GESTURE INPUT TOWARDS A "MODULAR" UNIVERSAL CONTROLLER PLATFORM AND INPUT DEVICE VIRTUALIZATION

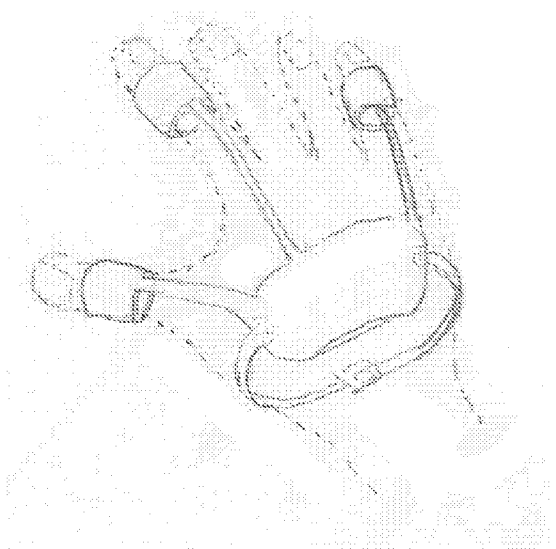

Fig. 2

(57) Abstract: This invention purposes a device physical interaction virtualization technology that allow simulating the physical interaction of user with (multiple) input device(s) with a single (universal) wearable platform. This invention also purpose a unique human centric "Hyper UI" that could serve as an integration point with multiple devices intuitive "high speed/parallel" input/interaction capabilities between human and devices. This invention also includes methods and apparatus (which can be used by other wearable for purpose of gesture related input /output) to integrate with other mobile/wearable(s) such as but not limit to smart watch / smart phone for providing natural/intuitive, context aware gesture input, single hand operation & high speed parallel input/interaction (could be b-dircctional with feedback such as tactile/pressure pattern as well as with visual/audio feedback) To facilitate more versatile usage and function extension, the proposed wearable apparatus (glove-like wearable) might also have modular design making the platform (plugable) and can be easily cleaned/maintained or extended with new hardware.

WO 2016/189372 A2

Methods and apparatus for human centric "Hyper UI for devices" Architecture that could serve as an Integration point with multiple target/endpoints (devices) and related methods/system with Dynamic context aware gesture input towards A "Modular" universal controller platform and input device virtualization

Cross Reference to Related Applications

The present application claims priority from US Provisional Patents Application Serial No. 62/152883 filed April 25, 2015, the full disclosures of which are hereby incorporated by reference herein

Technical Field and Summary

With the development of mobile, wearable computing platforms, VR(virtual reality)/AR(augmented reality) and internet of things (IOT), more and more devices are introduced around human, mobile/wearable or in homes , and each of them could have a “device based” UI (such as buttons, display screen, knobs, switches, LEDs and sounds etc.) that requires different ways of interaction (from user).

This invention purposes a device physical interaction virtualization technology that allow simulating the physical interaction of user with (multiple) input device(s) with a single (universal) wearable platform, which facilitates input/interaction across different devices, allow fast switching of input/interaction (virtual) device/methods (for cases such as VR/AR), allow high speed (nature) parallel input/interaction.

This invention also purpose a unique **human centric** “Hyper UI” that could serve as an integration point with multiple devices intuitive “high speed/parallel” input/interaction capabilities between human and devices.

This invention also includes methods and apparatus (which can be used by other wearable for purpose of gesture related input /output) to integrate with other mobile/wearable(s) such as but not limit to smart watch / smart phone for providing natural/intuitive (such as simulation technology related to realistic key pressing, knob turning, “force” like direct remote interaction/manipulation), context aware gesture input, single hand operation & high speed parallel input/interaction (could be b-directional with feedback such as tactile/pressure pattern as well as with visual/audio feedback)

To facilitate more versatile usage and function extension, the proposed wearable apparatus (glove-like wearable) might also have modular design making the platform (plug-able) and can be easily cleaned/maintained or extended with new hardware.

Embodiments of application/software (maybe 3rd party app on either the glove platform or the smart wearable platform-such as smart watch—in company) that is enabled with such technology is also included.

Introduction concepts and preferred embodiments

Some definition/concepts:

"Feelable" – some physics effects can be felt by user especially direct by skin such as mechanical pressure (patterns) , tactile/vibration (could also have patterns), temperature change, air flow/wind blow and etc. on multiple points on a surface,

A (dynamic) pressure pattern consists of a time sequence of frames of static pressure pattern "presents" to user. (for example can be used to indicate the direction or movement of something on the displaying surface)

IMU—here refer to inertia measurement unit, this is currently usually integrated circuit MEMS device/sensor that can provide multiple degree of freedom (DOF) inertia and navigational signals such as acceleration (projection on x,y,z axis), rotation (angular speed, around x,y,z axis), 3d-magnetic compass (projection on x,y,z axis) and etc.

On-demand display/screen: This could be the screen/display providing visual feedback when user interact using the wearable device for gesture/motion input which could also be a wearable display such as the display of a smart watch, the display at the back of hand of a "smart glove" as the wearable device (such as 101 in figure 1(a)) for gesture input , or it could be a VR/AR display/glass user is wearing, or even a "nearby" display screen (such as that of a laptop or tablet, a desktop computer or smart TV) that is visible to user and connected to the wearable device(s) related to gesture control when user performing such interaction.

Ergonometric simulated pose/gesture or "device simulation signature gesture": This is a pose/gesture (of limbs and possibly with palm) that user can comfortably take when using the purposed input device/input method, in some cases the pose can be kept in an extended period of time (such as more than several minutes) without getting fatigue, and such pose is similar to the pose user would use when manipulating tools/devices that is being simulated by the purposed input device/input method. For example: the "Ergonometric simulated pose" for simulating using a mouse, is "palm face down" (and maybe together with forearm considered to be "level" within some tolerances, such as plus or minus degrees, and such tolerance range might be adjustable/configurable), just like the way people would normally use a mouse on a flat surface; the "Ergonometric simulated pose" for simulating using a joystick or jet "steering stick" is palm facing side ways, just like the way people would normally use a joystick/steering stick, The "Ergonometric simulated pose" for simulating using a remote control is palm up, just like the way people would normally use a remote control, and "Ergonometric simulated pose" for simulating swipe/touch of a smartphone/tablet will be like the way people would normally use a smart phone/tablet, such as palm forward or down or with some angle, which might be configurable, (and might with 1 or 2 fingers stick to the surface), in yet another example, the "Ergonometric simulated pose" for aiming an weapon (such as a gun) could be either exactly the way user would hold such weapon (for example for rifle is using two hands, with right palm close-sideways and index finger on "trigger" if user is right handed and vice versa for left handed), or a "symbolized" pose that using one pose of holding kind of weapon (such as pistol, need to use just one hand) to represent all weapons, and also might allow optimize the pose for the purpose of comfort, such as allow elbow being supported (on the table or on the arm of chair, etc), and for related activity such as changing the clip of the weapon, the "Ergonometric simulated pose" could be some activities that similar or symbolize that activity such as rotate the palm from vertical (facing sideways) to facing up, so that it is same or close to the actual activity so it is can be easily understand or memorized and almost no need to learn in most cases. Above are just some examples, In general, if the tool being simulated have a handle, then the default "Ergonometric simulated pose" is the pose user would normally take to manipulate such handle.

The "device simulation signature gesture" can be used as kind of "hint" or "signature" gesture(such as but not limited to those mentioned in the "device simulation signature gesture" discussion earlier) for entering or existing (switching) context/operation mode of input device, and could be apart of the "context aware gesture" way of input and smart switching technology as in the following discussions.

(--Context-aware gesture and Smart switching--)

Related to **"<u>context-aware</u>" gesture (and smart switching)** way of input (and some possible embodiments):

A computer implemented method for "Context aware" gesture input/recognition/input generally including:
Determining the meaning of the user's current gesture/(changed) position or movement of user's body parts (such as but not limit to finger, hand, wrist, arm and etc.) based on a current "context" or operation/input mode; each unique command might corresponding to one or more gesture(s) that is unique under the same context;
the current "context" or operation/input mode (which can be switched dynamically in some conditions) can be determined/ is based on user's gesture/hand position/movement (for example, a zone can be defined for a control and when (user's hand/finger) entering such zone the context can be switch to the appropriate mode for such control, and when user's hand/finger moves out of zone, the context can be switched to other mode if the "virtual" control is not designed as following the hand movement), and the current (if any) operation mode/context might be based on/determined by the current control(s) or input devices being simulated that user is currently interacting with, and could also based on the "context" determining (context based or specific) "commands" or events/messages corresponding to a gesture (such as but not limit to a hand/finger gesture, could be relative/local or "global") that could indicate the begin or end of a "context"/operation mode, for example the context switch can be triggered when user's

hand/finger entering a zone(3D space) that is “exclusive” for operating a specific control/input device, for example user’s finger on (or “touched” virtually) the surface of a (simulated/virtual) button/key/knob, or a specific input/control selected could determine the only (or the best) operation/input mode, or when there’s no other available context/operation mode to switch (such as the current application/device is the only device user is interacting with and there’s only one possible way of interacting with it) such current context can be existed or change to another context(operation mode) by : leaving the “zone of operation”, using a “hint”/”symbolic”/”signature” gesture (such as but not limited to those mentioned in the “device simulation signature gesture” discussion above) that is not defined in the current context to signal the exit of such context/unselect of the current control/device, such as the gesture of “release” (all 5 fingers open and straight, not closely adjacent to each other, and the palm is basically not facing upwards (facing sideways or down)) or “thrown away” gesture similar to the “release” gesture and with arm movement from closer to body to away from body. Such exit of current context can also be triggered by user entering other context (such as select a new control/controller by entering its “exclusive operation zone” or with hints/symbolic/signature gesture)
The “automatic” switching of context based on the conditions/logic/intuitive gestures mentioned above (without additional user intervention such as pressing a button or issue a voice command is also called “smart switching”.

In some related embodiments in a given context, there are (a limited number of) unique gestures form certain parts of hand/fingers that is defined/trained/can recognized under this context, and the gesture recognize/input system might only “expects” these possible/expected inputs and filter out all others (for example for the “current” or “local” context), gestures/pose/motion input not defined in current “context” might be checked (for example by the gesture recognize/input system) for “context determining/switching” commands/signals or “short-cut” signals for some “context-less” activities (such as copy-paste). Once these “non-local context command” are found, context switch might be triggered or shortcut/global activities might be performed.

(In some embodiments) It is desirable the context switch might be triggered by gestures from “lower order” body parts (those segments closer to body in connection sequence, such as arm) prioritizing “higher order” body parts gestures (such as those of finger)
For example, In some related embodiment, the context switch might be triggered by determining the pose or “facing direction”/”pitch” of the back of hand, for example when the back of hand is vertical (to the ground) the context might be switched to “joystick” or “pointing” mode (might depends on other finger such as index finger gesture), while when the back of hand is facing up the context might be switched to “mouse” or “keyboard” mode, depends on other finger such as index finger gesture, and when the back of hand is facing down (palm facing up) the context might be switched to “TV remote control” mode, etc.

Also, it is desirable context switch might use “hint” or “natural signature gesture”(such as but not limit to those mentioned in the “device simulation signature gesture” section earlier) which comes from typical action being simulated , such as the (typical) gesture of typing, grabbing a mouse, flipping a page on touch screen, using joy stick, grabbing a remote control, and etc.

In a related embodiment, such context determine or context switching activity might include (but not limited to) using the rotation/orientation (or pitch) of the back of hand to determine which mode/context we are , and then determine "commands" based on gesture (such as those of palm, wrist and fingers). A universal "context less" action/command might be "click" from pinky which means bring up the "context" menu, similar to right mouse click.

In a related embodiment, feedback (such as but not limited to tactile/force/feelable, visual ,audio and etc.) is given to user after context changed.

In a related embodiment, the context might also be determined by the object /environment/"topic" user picked for interaction.

In some related embodiments symbolic gestures such as the "OK" sign/Gesture can be detected and used for confirmation (such as for the "OK" button on a UI), and "Kill" gesture can be detected and used for trigger a "cancel"/"discontinue" action(such as a commands or sate change)

The "context-aware" gesture can be used in multiple different scenarios and embodiments discussed below (such as those relating to the "human centric Hyper UI" or "device virtualization" in the later portion of this spec), which could provide more natural interaction between human and machine for example "intuitively (and reliably) switch "Mode"/"Context" of gesture with gesture itself, such mode or context might include changing target/endpoint(such as device or app) for the gesture command, or determining the appropriate context to switch to base on "hint" gesture or some typical "signature" gesture related to the context or input device being simulated (such as plam facing down and fingers arched to "hint" using a mouse, or palm facing sideways and 4 fingers curved to hint a "joystick" context/mode of input.

In a related embodiment to context aware gesture and smart switching, when using context based switching based on a body part/limb/palm action/change of "pose" or pitch, the "automatic switch" is conditional, for example the context should not be switched in the middle of a "captured/protected" activity or some sort of continues activity (not finished, in the middle of some gesture). Normally it must be by itself start from relative "static" position and stabilized more than a period of time that is configurable (such as 1 second). It must be a inflection point or "turning point" and not within a continues movement or gesture (that is not finished) . Priority or more "weight" can be assigned to the "gesture/movement" of slower moving/turning part or "bigger" part of all the related parts (such as finger, palm, hand, wrist, arm and etc.) that is considered by the gesture recognition/input system, for example, the orientation of the palm is considered highly relevant in context switching, for example palm face down together with fingers naturally curved (not complete straight) may signify using a mouse while palm facing side ways with fingers curved more (like grabbing a handle) may signify using a joystick, however it is desirable the mode switch is not only rely on such gesture but also based on the "motion status" – for example in order for a "signature pose" of hand to be recognized as a "context switch signal", it must be hold relatively still for a (maybe configurable) period of time, say 1 or 2 seconds, to make sure the context change is intentional and prevent mis-operation from accidentally switching contexts.

Discussion of some **wide-applicable (or typical) aspects** that can possible fit with multiple embodiments, (which can be "Features" of dependent claims):

"Remote" selection or interaction with device or documents displayed on a on demand UI with gesture/hand motion/pose information

A. "Point and select" style of remote device selection/interaction

There are several ways for detecting/identifying a "remote" device that is being pointed (by user's hand/finger), for example:

1) "narrow beam"/ high directional style of detecting/sensing of remote device(s), by using sensors/detection mean with a "narrow beam shape"/high directional of detection/sensitivity range, such as but not limit to: optical/infra red camera/sensor array for detecting beacons or transmitting signals from the "target" device (so that it will when the wearable device is pointing at certain target device the wearable device can "detect" the "signature" of the target device, such as but not limit to wavelength, modulation signal/code emitted), or using focused (narrow beam/high directional) infra red/RF/ultra sound emitter/blaster/transmitter on the wearable device with receivers on corresponding "target" devices (so that it will receive signal when being pointed at) located on the index finger, and maybe also with corresponding locating mechanisms deployed on the device (such as maker/pattern/beacon/emitter for cameras or optical/infra red/ultrasound/RF receivers/transceivers on the finger, or receiver/ transceiver for the emitter from the finger)

2) Directional antenna (RF direction "finder" type of detection):

By using RF directional antenna (such as but not limit to a loop shaped antenna) comes with (as a part of) a wearable device such as inside a brace around the wrist of a smart watch (or of a smart glove) , basically a conduit inside the brace formed a looped shape antenna and connects to the wearable (or chip inside the brace), which is directional to RF signal in certain frequency range/wave length. A second antenna having significant different direction characteristic (such as it is an "omni direction" antenna or it is an directional antenna pointing different direction as the 1st "loop" antenna can be used on the wearable device so that when the 2nd antenna having good reception while the signal from the 1st "loop" antenna (might be on different frequencies or modulation) becoming null/weakest (comparing to levels when facing other directions), the loop antenna's axis (which parallel to the arm's pointing direction) is pointing to the direction of target device. (and such information can be used to determine/identify which device user is pointing at, and make it the selected device for the wearable device to communicate/interact with.). Other type of directional antenna can also help finding the direction in a similar way (when pointing to the direction of the RF

source the reception signal level minimized or maximized, so basically at a "extreme value" or "inflection point" which can be used to determine the direction of the RF source)
3) Using multi-model sensors (such as gravity and magnetic direction) and location information of the target devices, (and maybe together with boundary conditions such as which side of the wall the device is placed/mounted) to calculate/decide which (fixed location previously known) device user should be pointing at. Devices mounted on a wall or placed adjacent to a wall, it is only visible in around 80 degree (if in a corner) or 160 degree (in the middle of room on wall) range, so by deciding if user's pointing direction is "in range" of these devices will help to improve the accuracy for determining target devices (by for example eliminating those devices in the "impossible angle" from the candidate list of devices.)
4) Using (innovative) technologies to determine Self locating in the room to determine the location relationship (such as orientation/direction/bearing) to the known (pre-registered) device(s)

In another embodiment, a wearable device having at least one camera and IMU unit (which can tell the pitch of the watch and desirably also with the magnetic direction), by using the camera on the wearable (such as a smart watch on wrist) "looking up" and examine the picture of ceilings, by optical pattern recognition or by using infra red marking/marker/ beacons there (such as projected by a infra red laser "scanning" device), the wearable device is able to determine the approximate location of itself in the room (and possibly also with direction) and from the direction or from other sensors showing the direction user is pointing, devices in the room that is previously registered on a map can be compared to the direction (relative to user/wearable device), so that when the direction/location is a match (within certain threshold that might be configurable), the device can then be considered as "selected" for more communication/interaction (with the wearable device). To prevent "mis-operation" by accidentally (unintentionally) select an device, the selection gesture needs to be relatively stable (from previous movement) for a period of time (might be configurable) such as 1-2 seconds, and (optionally) a special pointing gesture might be required.

Some possible Scenario could be:
Selection/interaction by "Point to" --system interprets user "point" gesture as selecting a "remote" device with criteria such as: with at lest index finger, straight and parallel or within 30 deg (or configurable value no more than 40 degrees) of palm surface, the direction of the finger within +-20 degree (or configurable value no more than 30 degrees) of the direction from the that finger tip to the device, and when the criteria satisfied that device is considered "selected" for interaction (or engaged with).
The detection method/elimination method can be various and usually for example using beacon/marker/speaker/microphone/high directional sensor individually or combined (for the purpose of identifying any possible devices at the direction user pointing at. The wearable might allow extension by providing port/interfaces (such as but not limited to I2C, SPI, USB, Bluetooth and etc. and hardware port/interface like but not limited to those depicted in Fig.1b) and allow components to be attached/detached, such as in Fig 1b. in which a finger part of a glove (wearable) can be exchanged, other "finger part" with different function using the same

port/interface or attached with other components to enhance/extend the functionality of the wearable, such as in Fig1.b directed a "narrow bean" detector/ directional sensor 103 component (such as but not limited to a camera, RF antenna or IR beacon) can be attached to the index finger to perform "point and select" of devices, such sensor provide or enhance the reliability of the device selection/de-selection/communication.

B. Point and Selection (style) for a "remote" document/content located not on the wearable device that hosting the a wearable UI (such as those with VR) with gesture

In an (Independent) embodiment a method of selecting an object **on a screen of another device** including Detecting user index finger movement or pointing direction and use the movement (position or delta) for 1) determine /identify the device user intended to interact with (such as but not limit to using methods described in "A") and then on the screen, selection of objects or providing navigation of some "visual" pointing device such as cursor, pointer or representation of hand for selection of objects, so that when user's index finger is pointing at a virtual object (within error threshold such as +- 10 degrees), on the device (screen), the object can be identified.

In an embodiment, the wearable, together with a Virtual "dashboard" integrated multiple devices (such as via IoT network) running on a computing system, such as onboard the wearable, or a server computer connected to the wearable, that can generate the synthesized/integrated multi-device information for displaying (GUI), which possibly displayed on a wearable screen or "On-demand display/screen". There might be an option for user to pick (or "zoom in") a specific device via this GUI/integrated dashboard, such as but not limited to selecting an icon on the "desktop" or "tree" in the dashboard/GUI.

C. "Remote direct interaction" (such as selection/manipulation, without pointing to select)

- "Force" like direct remote interaction

A computer-implemented method performed by one or more processors for providing user with a innovative "force-like" gesture based interaction (similar to the "force" interaction depicted in the movie "star wars") with "remote" devices/targets/endpoints not seemingly within immediate reach of hands (or within haptic range) in reality or in other cases (such as on screen or in VR simulation, in Games) including:

Determining when the interaction starts:
Multiple ways, such as by software events (when entering certain mode such as but not limited to: entering "hero" mode with pickup in games such as Star wars Battle front), by events triggered from user gesture changes such as detecting a "force-like" preparation gesture and/or with detection of Myoelectric of muscle (such as hand/fingers muscle activation detected via EMG/MMG) in conjunction with said a "force-like" preparation gesture, such gesture/gesture change in closely resemble the gesture for start using force as depicted in the related movie and game (such as star wars battle front) -- for example , by detecting user's hand's short distance movement (no longer than 3 feet) in the direction that is roughly perpendicular (in a range within 25 degree of angle to the perpendicular

direction) to the plane of the palm facing away from user's body , with little or no rotation of the palm, and the palm's finger remain in a gesture that is close to the "force" usage gesture as depicted in the movie star wars (such as five fingers in an open gesture between 5 fingers all completely straight and the gesture like grabbing a baseball (curvature of each finger is within 90 degrees (angle from the base section of the finger to the tip of the finger) and the for each finger (except thumb) the curvature is considered similar within plus or minus 35 percent (of curvature angle) , and the middle finger is in a direction towards upwards (within 80 degrees of angle from the "upright" direction), when such movement is detected (by ways such as but not limit to detecting user's palm's angle/pose/gesture and movement as well as each fingers (or major fingers) curvature and/or relative angle/pose/gesture by using inertia and/or optical sensors that might be located on the glove-like wearable (in case of IMU/curvature sensors) or external (such as optical);

or by detecting a "force like selection" initial gesture by determining if the following condition(s) are met: 1) The palm is facing away from user or the back of the hand is facing towards user, 2) the hand gesture is a relatively open position like depicted in Fig.3 position 1 (solid line) where the fingers and thumb are not significantly flexed/bent or closely adjacent (closely adjacent means side by side with no space between them);

Upon such "force like selection" initial gesture is detected, determining the general pointing direction of this gesture (can be calculated using algorithms such as but not limit to by "averaging" the vector direction of the five fingers and get a general pointing direction (one way is just to use vector add to add all (normalized) directional vector of available fingers with possible different (configurable) "weight" for different fingers, and determine the direction in space), such averaging algorithm can have variations such as using different "weight" for thumb, and all other fingers, might take into account of the palm facing direction, the "degree of curving" of palm –which should be relatively not curved much, and maybe "degree of curving" of fingers, etc.);

In case there's also data for user's head facing or looking direction available, it is optionally that the control system may further filter out those situations where the general pointing direction of the this gesture is not at the direction user head is looking at (this could be an option and if related situation not filtered out, will allow user to interact with remote objects they are not facing/looking at);

Then determining which remote object(s)/target(s)/endpoint(s) user is intended to pick/select or interact with, using ways such as but not limited to:

1) In game/simulation, select objects/targets within effective range (distance from user and angle from user's palm facing direction) of the "force interaction" from user (which might be determined by the game/simulation)

or 2) Search a range in user's general pointing direction (for example in the center areas of the search "cone" defined by the general pointing direction, or defined by the individual direction of fingers, and determine if an object is in range (in this "cone") (and might provide feedback such as highlight, zoom in and/or tactile). The method for detecting/identifying a "remote" device that is being pointed can be similar (but not limit to) those methods discussed above in the A. "Point and select" style of remote device selection/interaction section (if the device being searched is in real world) or perform “virtual object” search like ray tracing or other kind of object finding methods (such as those provided by 3D/game engine) to find out the virtual object user is pointing at in case the interaction setting is in the virtual world (such as game, simulation,3D software/OS and etc)/on GUI;

If user keep this gesture of the hand and begin “scanning” (changing the hand/wrist/arm direction), the system will update related directional pointing region and determine if an remote object is in range. And might provide feedback such as highlighting (visually) or tactile/pressure feedback(tactile) to indicate an object/target is in range.

And by detecting the change from user’s “force like selection” initial gesture to “force like attracting gesture” such as depicted by the position 2 (doted) in Fig. 3, in ways such as but not limit to: determine the “openness” of the gesture from a “wider open angle” to a “narrower open angle” (the limits can be configurable or customizable). Such as in ways (but not limit to):

1) calculate the relative angle (scalar value) of each finger’s (new) pointing direction to the (new) averaged pointing direction(vector) or the “general pointing direction” of the gesture as mentioned above. If all 5(or available) values are decreasing to an extent not yet at “fully closing” level (for example at a range close to “halfway” of open pose and fully closed position, or “half grab” position and etc., and desirably the range and/or detecting mechanism configurable/customizable) in a “similar fashion” (for example not in a case one or 2 value or “change” ratio significantly different than the averaged value or ration, for example more than 30%) , or at least the values are not moving in a “contradicting way”—some of the fingers are “closing” while some other(s) are “opening”-- which signifies a change of gesture or context), then the “attracting gesture” can be determined/related message/events can be triggered; or,
2) Determined by the measurement of the “degree of curving/bent” of the 5(or available) fingers such as but not limit to from the data of the “bent/flex” sensor of the fingers, when the “degree of curving/bent” are increased for all related fingers to an extent not yet at “fully closing” level (such as “halfway” of open pose and fully closed position, or “half grab” position , and limit/range could be configurable, and desirably all finger moved in a similar fashion, for example but not limit to, all the changed ratios of bent/curving are within 25% (or a configurable value) of each other or the average value, or at least not the values are not moving in a “contradicting way”—some of the fingers are “closing” while some other(s) are “opening”-- which signifies a change of gesture or context) , then the “attracting gesture” can be determined/related message/events can be triggered;

So when the conditions mentioned above is satisfied and the “averaged” open angle of the fingers (on that hand) changed to a smaller value under a limit (compare to the “averaged” open angle of the fingers in “force like selection”) then we can determine that this is the “force like attraction” gesture, the current “remote object” in the “selection cone” becoming “selected” , and maybe events/message can be generated for subsequent processing of the remote object selected;

Similarly when detected, a “grab” gesture similar to normal grab action of the hand (all finger closed) can also be used for signaling certain meaning such as grab an object, document or content (such as in case of virtual world/UI) or in real life interacting with remote devices this might be used to signify grabbing a “UI” or document/content(when applicable) from the “remote” device user interacting with to a local wearable, mobile or “on-demand” device, the current “remote object” in the “selection cone” becoming “selected” , and maybe events/message can be generated for subsequent processing of the remote object selected;

Some subsequent processing could be for example (when in a virtual/simulated environment such as in a game or in a 3D GUI), the object might be rendered as “attracted” with the “force like attracting gesture” and appear to directly fly into user’s hand on the wearable/on demand display, or copied/moved (in case of copy the content onto the clipboard) with a grab gesture; In a real world “remote device” situation, the UI of the device or the might be picked and (selectively) display on a wearable screen or “on-demand” screen user can see, or the previously selected file/message/content on the remote device can be put to the clipboard with the “grab” gesture.

In related embodiments MMG as mentioned above to determine Myoelectic of muscles can use microphones and methods such as but not limited to the following article: http://www.academia.edu/5624561/Robust_ultra_low-cost_MMG_system_with_Brain-Machine-Interface_applications

A typical scenario will be:
In one embodiment, a (3D) interface or game setting/environment modeling a 3 dimensional “world/virtual workspace” or “sandbox” is provided for user (maybe via an VR/AR system such as a HMD or glass) to interact with, in which some objects appears to be “remote” or having a distance outside of user’s haptic range(outside of immediate hand reach) and can be manipulated using the wearable device in a “Force”-like (as depicted in the movie “star wars” series) direct remote interaction, such as (but not limit to) move/rotate the object using gesture without actually touching the object, grabbing object remotely by directly moving them into user’s “hands” (visually) or moved to appear very close by (within easy reach), and etc. An object can be brought within the haptic range for user to work on, for example to manipulating. If object is big, however user can be self motion towards the object, so that it can be within haptic range, or use a manipulator to remote manipulate it (like a joystick or mouse). The wearable control system (and also could be a hyper UI system) can determine if user is trying to “pick up” some object (using this “force-like” manner, by detecting user’s special “force like selection” (gesture and possibly other manners such as but not limited to muscle force change which can be detected by EMG or MMG), and high light such candidate objects, if object is too far, user might “reach out” to grab some object or aiming, this can make the system to “Zoom in” the area so that user can easy to pick, and when user really want to use “force” to grab it to hand, he/she will close all the fingers a little, just like “half grab” or “force like attracting gesture” mentioned above which will “attract” the object to user’s hand or within “haptic range” (within reach of hands), and user might use the other hand or this hand (when change to other gesture) to manipulate it.

And optionally when user finished, user can throw it back and it will automatically return.

In a (glove like or watch like)wearable solution (embodiment), the context can be determined by the wearable and passed as a message, or it can suggest signal on (context change) and be determined by app, also there could be "high level" messages such as (index finger flex 20 deg) passed on sides of low level data.

D. Motion/Position based gesture interaction with "local" wearable terminal and software/application (that might be connected to other devices)

A wearable information system to facilitate remote (such as those not within immediate reach of user's hands) interaction with other devices (such as IOT devices) over network (such as via TCP/IP) comprised of:
IMU/motion sensor (and other kind of input method); a display screen (might be touch screen); At least one processor and related memory/storage for executing commands and managing tasks and 3rd party apps; wireless communication units such as Wifi, Bluetooth and NFC (and maybe also with other methods); A software/app running in the processor/OS that use the communication units to connect to at least one outside devices such as IOT devices and present a UI to user on the display; The software use the sensor to capture user gesture and for a particular "recognized" gesture such as turning a knob, flip on/off a switch, point to selection or navigation, press a button, the software use the (dynamic) gesture data to determine commands (of UI control) and/or "messages/events" sent to the IOT device(s) so that user can interact/control the IOT device remotely with gesture.

In an embodiment, A computer-implemented method (for Motion/Position based gesture interaction) performed by one or more processors onboard a wearable device including:
Detecting user gesture from Motion/Position sensor (such as IMU) data to determine gesture (specific to the application/control), for example a wrist turning can be detected using the Motion/Position sensors by examining gyro scope data for rotation acceleration at the direction of wrist turning or by examining accelerometer changes to determine gravity direction change and other movement of hand, to determine how much angle have the wrist in which the wearable is on had turned (from initial position), such change can be used to drive/turn a "virtual knob" or a visual control on the screen, flip on/off a switch, point to selection or navigation, press a button (on the UI), and etc; So basically using the dynamic gesture data gathered from the wearable Motion/Position to determine commands (of UI control) and/or "messages/events"
In a related embodiment further includes establishing a connection/communication with one or more "remote" device(s) (such as but not limit to a IOT device) wirelessly using the wearable, and use the Motion/Position based gesture input method mentioned above for adjusting/making inputs to the control of the "remote" device(s) and optionally display remote device's UI interface (if any) and update any changes.
In a related embodiment said establishing a connection/communication with one or more "remote" device(s) (such as but not limit to a IOT device) wirelessly using the wearable

may contain a step of using "Point and Selection" (as described in A above) or "Remote direct interaction" (as described in C above), possibly together with hardware these methods required, to identify the device to interact with, and use the IMU based gesture input method mentioned above for adjusting/making inputs to the control of the "remote" device(s) and optionally display remote device's UI interface (if any) and update any changes.

In a related embodiment, the wearable device for communicating with other devices wirelessly might used one or more IOT protocols such as (but not limited to) the following (when communicating with IOT devices): AllJoyn, Thread(networking protocol), OIC (Open Interconnect consortium), IIC (Industrial Internet Consortium) and etc.

E. (Motion/Position) gesture based interaction using smart wearable terminal (like smart Watch) and additional IMU sensors on wearable devices(such as a ring shaped device) on the finger(s) of the same hand for interaction with "remote" device and/or "local" smart wearable/mobile terminal

A similar kind of UI, a mobile/wearable gesture manipulated UI can be considered as follows:

In an embodiment, A computer-implemented method for gesture based UI (for example but not limit to: implemented by a control/processing software/application running on either onboard the device with position/motion(such as IMU) sensor or another wearable "processing" device (such as but not limit to a smart phone, smart watch, glove-like wearable device) that connected to this device) performed by one or more processors onboard a wearable device including:

receiving data or message(s) from motion/position sensor(s) on the wearable frame/device(s) (such as a part of a glove, or an independent ring) that measure the movement/position of at least one moving parts for providing corresponding motion data (such as but not limit to a ring or a part of glove with IMU sensors), the sensor data indicative of a gesture/pose/movement of the part of user being measured (maybe relative to a reference coordinate frame);

passing/directing such data/message to related (desirably configurable) module(s) (such as but not limit to a recognition or input module, a 3rd party app or plug-in module) as input to the system;

Presenting a visual display/feedback on a display screen on a involved wearable devices(such as a smart glove with OLED display or a smart watch, or on a VR/AR display that user is wearing);

In a related embodiment, "Context-Aware" style and/or "smart switching" style gesture input methods(as described in earlier text) is(are) used for more intuitive user input.

In a related embodiment, it is desirable the control/processing software/application (for this kind of UI) also drives the (tactile) feedback to user which could be provided by the wearable(s) involved.

In a related embodiment, it is desirable motion/position data from a "second source" that measure a different (movable) part of body/limb that could have relative motion from the part body/limb being measured in the previous mentioned embodiments (such as one measuring the movement of finger and/or palm, the other measuring the movement of wrist/arm) is used to (better) determine user's gesture or providing more kind of gesture input (for the system) as more moving part now being measured. For example, a wearable/mobile device on the back of user's hand or wrist with a display and capable of running software/apps (such as a smart watch, or a part of a smart glove) could communicating with a wearable device or frame on user's finger or measuring their motion/position (such as but not limit to a part of a glove, or an independent ring) and combine the position/motion data from the "finger" sensors with the data from its own sensors (such as IMU sensors) (such as accelerator and gyroscope, magnetic measuring wrist/arm movement) to provide more detailed gesture information for purpose such as gesture recognition/input or related manipulation/control.

In one embodiment of a kind of **human centric**, IMU based UI for devices", a software/application(or computer implemented method) providing gesture input (to the UI system or OS) **combines** IMU (gesture like "pose"/pitch and acceleration/rotation as well as magnetic direction info) from a wearable device on user's arm/wrist , such as but not limited to a smart watch (together) with data from at least one IMU sensor(s) located on a wearable device(s) on at least one fingers of the hand of the same arm, (such as a located on a ring, a glove) , and performing "2D or 3D gesture recognition/input" (desirably context aware and/or smart switching based) for driving/generating UI events such as selection (pointing, click, double click, drag, drop) or navigation/zooming(such as swipe, rubber band will need 2 fingers, grabbing need 2 fingers) , so that such operation normally require the other hand touching the surface of the wearable device (such as watch) can now be performed with a single hand.

In a related embodiment, the gesture input (from the motion/position sensors located on the 1st wearable part/device that provides a motion data of a finger or fingers) is communicated to the wearable processing device (such as smart watch or a smart glove), and combined (such as but not limit to: using the acceleration/gyroscope "deferential " values from the corresponding axis of the 2 devices to get a relational motion of the finger to the wearable processing device), to provide (gesture interpreted) input commands to the (UI of) wearable processing device , such as a click , a swipe indicator, direction of navigating/moving of the view ports, turning a "knob" or flop a switch, and etc, so that user can use the single hand for operation (of applications that usually require 2 hands to operate such as one hand hold the device and the other hand touch the screen of the device)

In an example, a "dialing" gesture input can be detected by using a "ring" or finger part of a glove with motion/position sensor on it such as but not limit to IMU sensors, flex/bent sensors and etc, and with "second source" motion/position data from other part of hand/glove such as from a smart watch, a more precise/distinctive gesture input can be provided for purpose such as driving related (UI) control, generating related (UI)events/messages, and etc.

In a related embodiment to above , visual response to the single hand gesture input such as scroll, highlighting, cursor/window navigation/movement, button clicking, knob turning for the corresponding UI control (being selected/interacting with using the wearable devices), are generated by the UI system of the application/software running on the wearable processing device and displayed on the screen of the wearable processing device.

In a related embodiment, the wearable device for communicating with other devices wirelessly might used one or more IOT protocols such as (but not limited to) the following (when communicating with IOT devices)
AllJoyn, Thread(networking protocol), OIC (Open Interconnect consortium)
, IIC (Industrial Internet Consortium) and etc.

In a related embodiment, such interaction might be performed to a 3rd device with client side/remote UI running on the wearable processing device and communicating with the wearable processing device, So that user can manipulate/operate the 3rd device in a single hand gesture input way.

(--Multiple Device copy paste, can use some "point and select and force like interaction" concept above--)

Exchanging documents quickly and efficiently is generally important to a successful business and is desirable even in non-business environments.

An embodiment is directed to (computer/software implemented) systems and methods for selecting and exchanging information between different devices (such as between a source device and a target/destination device) , including the possible situation of using a device independent of the source and target device, including:

1. Determine the source (such as file/content on a "source" device) when "**Source Content Manipulation Start Signal**" is detected, such as (but not limit to) detecting "drag and drop", grab gesture, or copy/cut command from other signature/symbolic gesture input or UI control input (such as confirm a context menu item); In case no source is detected then the process will not proceed and the state/context/operation mode is not changed. If source content can be determined and is an valid one (by related validation system), then the state of "copy/paste" or "drag and drop" maybe entered (and control from gesture become "captured" means not switching to other operation mode until dropped or other exit condition are met), the source location information might be transferred and depending on implementation, contents or "short cut"/link might be put into clip board or buffer for "drag-drop" operation , and desirably sending Feedback in visual (such as change of cursor, highlighting and moving the icon of the file with

hand movement and etc.) as well as tactile (such as pressure or vibration indicating "grab" or "stickiness");

2. Detecting/Determine drop target (if not the same device) as user move hand/arm (while keeping certain gesture of grabbing/dragging in case of drag and drop) , by using "beam" like "pointing selection" manner to detect/identify possible destination target (in the area of "beam" directed by gesture of user's hand such as pointing or other ways of indicating direction) "drop" or "paste", the "beam" like "pointing selection" method might including such as (but not limit to) IR (Infra red), RF(special antenna, ultra sound) or use "NFC" (in which case user are holding or pointing right at the device within 4cm). Such target might also be indicated to user (such as visually like highlighted or change color/border, zoom in, and etc, and maybe also in tactile such as a different kind of vibration/pressure in different part of hand/arm);
3. Upon a drop target/paste destination is available (and could be validated by verification/validation system), determine if user performed the "paste"/drop gesture input. When such signature/symbolic gesture (or select related UI commands), the state of operation maybe changed (such as from "dragging" to "drop" or normal "not dragging"), and communicate with destination/target device, so that the content of the source can be transferred to destination (could be in multiple forms such as but limit to, under the control of a file/content exchange system (such as related to clip board, drag and drop controller) and copy the content directly to destination/target (if the file/content exchange has the content). Or tunneling the 2 devices (source and target), either directly or thought the relay of the wearable device which runs the file/content exchange system for exchanging/streaming of the content to the destination/target.
4. optionally (at the mean time) visual effects such as animation can be provided as a feedback for such exchange activity
5. In a related embodiment , IrDa/IR remote or NFC/RFID tagging can be used to negotiate the communication channels(such as that of RF especially bluetooth, paring passwords etc.

An embodiment is directed to systems and methods by which documents/Messages/contents may be exchanged between a computer and a portable/wearable device equipped with memory storage, network interface and appropriate software.
In one embodiment, a wearable device with a pointing/gesture means is used to identify/or select a document on a computer screen. The document's location information for retrieval (a binary version or copy of the document) is communicated to the wearable device. The binary version may then be "pasted" onto another computer or viewed on the wearable display (with such information).

Additional representative embodiments of the present teachings are directed to computer implemented methods/systems that include capturing/detecting a gesture input(such as pointing/clicking/start dragging on) with the wearable device on a first computer display/UI, wherein a document is shown on the on-demand UI or on a first computer, identifying the gesture (such as pointed location) try to find out selected file's path/location information, transmitting a request to a first computer coupled to the computer hosting the document/content using the location information, the request requesting a copy of the document from the first computer, receiving the copy at the 2nd device (such as the wearable device) from the first computer, and rendering the copy onto a on-demand UI or screen of the 2nd device.

Further representative embodiments of the present teachings are directed to wearable devices that include a processor coupled to storage memory and a gesture recognition/input application/software stored in the storage memory (that communicates with gesture input device). When executed by the processor, the operating gesture recognition/input application/software is configured to capture/detect gesture input data/message/events through(from) the gesture input device. The wearable devices further include a transceiver under control of the processor and configured to transmit and receive data and a document/message exchange application/software module stored in the storage memory. When executed by the processor, the operating document/message exchange application/ software module includes an identification module configured to examine the gesture input data from the gesture recognition/input application/software (provided by the gesture input device) for the document/content (selected/to be exchanged) on the On demand UI display (or 1st computer screen), wherein the gesture input data is examined to identify the location information associated with the document/content and its computer “container”, and a communication module configured to transmit a request to the computer “container” of the document/content through the transceiver and using the location information, wherein the request requests a copy of the document associated with the location information, wherein the communication module is further configured to receive the copy from the first computer.

Still further representative embodiments of the present teachings are directed to specially programmed computers that include a processor coupled to storage memory, a network interface under control of the processor and configured to send and receive data from one or more networks, a computer identifier (ID) stored in the storage memory representing an identity and location of the specially programmed computer, and a document/message exchange application stored in the storage memory. When executed by the processor, the document exchange application includes a communication module configured to receive a request from a wearable/mobile device through the network interface and using location information contained within the request for retrieving a copy of a document/or send out related message associated with the location information, wherein the communication module is further configured to send the copy to the wearable/mobile device.

Still further representative embodiments of the present teachings are directed to computer program products having computer readable medium with computer program logic recorded thereon. The computer program products include code for capturing/detecting input data/message(s) provided by the gesture input device wherein a document is shown on the first computer or On demand UI, code for examining associated gesture input data/message(s) to identify location information related to the document(selected/to be exchange), code for transmitting a request to the computer containing the document/content and coupled to the first computer display (or On demand UI) using the location information, the request requesting a copy of the document from the computer containing the (selected) document/content, code for receiving the copy at the portable device from the first computer, and code for transforming a state of ones of a plurality of pixels of a portable display of the wearable/mobile device to render the copy onto the device display (or On demand UI).

Additional representative embodiments of the present teachings are directed to computer implemented methods that include executing instructions on a computing platform so that binary digital electronic signals detect (with sensors on a wearable device) a user gesture pointing/interacting a document/content on a first computer display, wherein a document is shown on the first computer display, executing instructions on the computing platform so that binary digital electronic signals identify from the gesture of user, the location information related to the document, executing instructions on the computing platform so that binary digital electronic signals transmit a request to a first computer coupled to the first computer display using the location information, the request requesting a copy of the document from the first computer, executing instructions on the computing platform so that binary digital electronic signals receive the copy at the portable/wearable device from the first computer and executing instructions on the computing platform so that binary digital electronic signals render the copy onto a portable display of the portable/wearable device.

(--A human centric "Hyper UI", **can use some "point and select and force like interaction" and Multiple Device copy paste concept above --)**

A human centric "Hyper UI" that could serve as an integration point with multiple devices comprised of:
#1) A mobile/wearable means/device that user worn or mobile with user that can communicate with other devices/systems wirelessly (such as but not limited to: RF, IR, ultrasound and etc.);
#2) An "On-demand" based visual display (system) that can [use said mobile/wearable means as input devices (and might also as output device)] to provide user with a visual/graphic UI for the device(s) he/she is interacting with(could be such as but not limited to: some other devices via the said mobile/wearable means) , visual display (system) may include mobile, wearable (such as VR/AR helmets. Glasses, visors etc) or an "on demand" display on external display system (which not mobile with user or worn by user) coordinating with a system that is mobile with/worn by user that is visible to user (with the mobile/wearable system);
#3) At least one (3D) gesture/hand pose/motion sensor system/means as (primary) input, this could be for example a series of motion/position and related sensors (such as but not limit to IMU sensors and flex-bent sensors and etc) located on a frame that moves together with user's finger and possibly also hand(palm), wrist, arm, such sensor system(s) might be part of the wearable device in 1) or could be an independent wearable. // both situation

It is also desirable that a “tactile/pressure”(or other “feelable” feedback type) output system is also provided in the UI system (for the purpose such as but not limit to direct feedback, control-location indicating, control surface simulation and etc.) ;

so that when a user interacting with multiple “remote” devices (could be at the same time) which can not be very easily reached by user directly (for example out of haptic range of user or when user’s hand is occupied for other tasks) and/or with objects displayed on the #2 on demand screen, the gesture/(hand) pose/motion of user/human to machine direction of interaction can be pick up by #3, and interpreted as input to UI system displayed on #2 as well as related modules/devices that expecting user input, and possibly communicating with other device(s) via #1 when necessary (such as when user interacting with other “remote” devices, and desirably also drive the tactile feedback (of device to human direction) can be provided by tactile/pressure “output” means on the wearable (if available)

So that an intuitive human-machine interaction such as (but not limit to) 3D gesture manipulated 3D user interface (maybe provided in a immersive 3D display system by the on demand display system) can be achieved;

In a related embodiment, A controller/processing unit(s) of the mobile/wearable device (such as gesture input “smart” glove, smart watch) -- may further coordinating “Task management/scheduling” and “communication” for devices as well as “copy/paste” with multiple endpoints and the related software running on it, which could include 3rd party app could provide further integration of the devices such as using a “unified GUI” instead of individual UIs.

In a embodiment related to all of the above, the wearable device for communicating with other devices wirelessly might used one or more IOT protocols such as (but not limited to) the following (when communicating with IOT devices)
AllJoyn, Thread(networking protocol), OIC (Open Interconnect consortium)
, IIC (Industrial Internet Consortium) and etc.

(-- Software (method) integration (OS,app, --)

A computer-implemented method (for human centric “Hyper UI”) performed by one or more processors onboard a wearable device including:
receiving sensor data from one or more sensors on at least one wearable gesture/pose/motion input device that can accommodate and move together with at least one (section of) moving part of user’s body or limbs such as (but not limit to): finger(s), palm(s), hand(s), wrist(s),arm(s) and etc , the sensor data indicative of a gesture/pose/movement of the part of user being measured (maybe relative to a reference coordinate frame);

Providing On-demand visual display of a graphic user interface, such as on a display screen on the wearable device(such as a smart glove), on a wearable processing device (such as a smart watch/smart phone) companion with the wearable device for gesture input (such as a gesture input glove) or on a VR/AR display that user is wearing;

determining a visual feedback on the On-demand display environment based on the sensor/gesture data; and desirably also drive the tactile feedback (of device to human direction) can be provided by tactile/pressure “output” means on the wearable (if available)

communicate/connecting with other (necessary) devices/systems (user desired/selected to interact with or related to an object displayed on the on-demand display) wirelessly;

In a related embodiment, the computer-implemented method/system includes using “context aware and smart switch” method (as mentioned in earlier text) in gesture recognition/input process (or in processing of sensor data for gesture/motion recognition/input) to allow intuitive control for user and fast switching of modes.

In a related embodiment, the computer-implemented method/system (further?) includes using “force like” remote direct interaction method enable user to select and manipulate an “remote” device or an virtual object displayed on the “on-demand” display.

In a related embodiment, the computer-implemented method/system further includes using “point and select” like remote interaction method enable user to select and manipulate an “remote” device or an virtual object displayed on the “on-demand” display.

In a related embodiment, the computer-implemented method/system includes using “multi-device copy-paste” way of interaction/inegration method (as described in text earlier) to enable content/documents to be transferred between 2 (or more) devices conveniently (in a way similar to the actions required for user to do this within one device).

In a related embodiment, A controller/processing unit(s) of the mobile/wearable device -- may include (one or more of) further tasks such as (but not limit to): coordinating “Task management/scheduling” ; managing 3rd party apps installed on the platform; coordinating “copy/paste” with multiple endpoints/devices (as mentioned in the **Multiple Device copy paste section above)** and the related software running on it;

In a related embodiment, the computer-implemented method could further integrate with multiple devices, for example by providing additional services such as (but not limit to) one or more of the following:

1. State/Context management ;
2. Gesture interpretation services, maybe with plug-in (could be 3rd party software)
3. Input method / Virtual Input Device management

4. Feedback (visual) and tactile management, coordination 3D calculation, collision detection and model management (for example for plug-in)
5. UI/Display management;
6. Communication/Connection/ access control (connected device) management
7. Hardware extension management (such as plugin, drivers)

So that the (3rd party) app running on this platform just need to specify their desired input device type (rather than implement the detailed code for doing so), and/or how to switch in and out of a specific input device/input mode; facilitate communication from app to other devices (such as for drag drop), and/or support for hardware extension. These services work together to support the app and allowing a "Hyper UI" style of interaction between the app and user.

(--smart watch or a Display (VR, AR) integration and related control system, run app)
A computer-readable storage medium having instructions stored thereon, which, when executed by one or more processors, causes the one or more processors to perform operations, comprising:
generating a display environment on a display of a wearable device (such as a VR, AR system or smart watch, smart glove);
wearable gesture input device, such as glove like frame with sensors on the part covering/move with finger(s) and/or palm;

receiving sensor data from one or more sensors on the wearable gesture input device (as mentioned above in 1), the sensor data indicative of a gesture of user (maybe relative to arm or a reference coordinate frame);
providing feedback on the UI based on user's input/command, and desirably also with other channels such as tactile/haptic capabilities provided by the wearable gesture input device;

In a related embodiment, further including manging/running a (3rd party) app
in a related embodiment said display environment is three-dimensional (3D) ;

In a related embodiment, further including connecting to another device ("remote device") wirelessly and allow control/interaction with the "remote device" with the instructions stored or installed/managed (such as a 3rd party app), for example including using the data from sensor(s) --such as interpreting them as commands/messages, maybe based on gesture or context aware gesture-- to interact with /control the remote device.

(--integrating architecture perspective --)
A human centric "Hyper UI" that could serve as an integration point with multiple devices having features such as:
An architecture includes means for storage, manage and run for apps;
Communicating with other "remote devices" wirelessly;
Integrating the accepted gesture data from sensor with control system, remote device and feedback system, for example interpreting gesture data (for example using

context aware input method) with OS/Software or application stored or and display the UI(feedback) and optionally drive "tactile" feedback of the output device;
In a related embodiment a software/app stored on a wearable device and can be executed, having such architecture features.

(--software platform providing service for the above applications--)

Some key services provided by this framework/related software methods are:

State/Context management;
Gesture interpretation services (plugin, could be 3rd party);
Input method / Virtual Input Device management;
Feedback (visual) and tactile management, coordination 3D calculation, collision detection and model management
UI/Display management;
Communication/Connection/ access control (conntected device) management
Hardware extension management (plugin, drivers)

So that on the app/plugin level, they just need to specify their desired input device type, how to switch in and out, and the input method and input state are managed by the framework, the icon/hand skeleton/cursor can be changed according to the current mode, and the feedback management system provide feedback, the communication to other devices (such as for drag drop) are managed by communication/access control, and hardware plug and play are handled by hardware extension management. These services work together with the app they support to allow a "Hyper UI" kind of interaction application.

In a embodiment related to all of the above, the wearable device for communicating with other devices wirelessly might used one or more IOT protocols such as (but not limited to) the following (when communicating with IOT devices)
AllJoyn, Thread(networking protocol), OIC (Open Interconnect consortium)
, IIC (Industrial Internet Consortium) and etc.

(--A revolutionary word processor--)
In one possible scenario/application, A "direct manipulating" word processor can be created by allowing user to use multiple finger as different tools, and context-aware smart switch of different modes so that when user palm is facing down, the input mode is (automatically, according to context) changed to "keyboard" simulation, so that user can directly type into the word processor editor without the need to look or select from a "visual only virtual keyboard", when user's palm is facing front and one of the fingers pointing at certain text, depends on which finger (which can be assigned to different tools, such as selection/highlight, underline or bold) user is using to interact with the text, and maybe further depends on the "depth" of user's finger interaction with the text (for example a "deep press" by index finger that appears to "penetrate" the depth of the displayed text might indicate "cross out" of the text interact with), there are many other interactions using multiple finger separated or combination, symbolic gestures (such as

"Ok" gesture for a confirm button or "kill" gesture for a cancel button on a dialog box). Such natural and multi-finger "parallel" style of editing (rather than having to select one tool at a time) could make editing/changing more directly, intuitively and high efficient.

Sample scenario: In a related embodiment/example, Allowing user to use index finger as "pointer" (indicating the device) and interpret "left click"/"left(button) down" like events (similar to the meaning and effects of mouse left click/mouse left button down) as (detection of) the flexing of thumb or middle finger over a certain (configurable) threshold so that system could know the action is intentional, and such "left click" can be assign meanings such as selection or other meanings like corresponding mouse events, and interpret "right click" like events (similar to the meaning and effects of mouse right click/mouse right button down) as (detection of) the flexing of pinky finger over a certain (configurable) threshold (so that system could determine the action is intentional), and right click can be assigned meaning for (or triggering) context menu/commands/status information

In an embodiment of VR/AR UI system, including combining the IMU data from a wearable on user's finger such as a ring (wirelessly), IMU data from a wearable on user's arm/wriest such as watch, to form a **gesture-manipulated** system for the "wearable UI" displayed on Glass/VR helmet of the user (simultaneously), such UI provides feedback to user's gesture input (that is interpreted from the combination of the IMU data)

(And in a related embodiment,) it is desirable that in some embodiments the software/app might run on a wearable device with a display (such as smart watch) that communicate with this glove/helmet

(--A input/output device virtualization)

An input/output device virtualization system simulating a hardware input device that allow user to operate it with similar gesture/hand movement/pose to comprised of:

1) One or a group of cooperating wearable means/device that user worn on the part of the body related to the operation of one or multiple input device being simulated, such as hand/arm/wrist with user that can communicate with other devices/systems desirably wirelessly such as but not limit to RF(such as Bluetooth, NFC, Wifi and etc), IR, ultrasound and etc.;

2) At least one (3D) gesture/hand pose/motion sensor system/means as (primary) input, this could be for example a series of motion/position and related sensors (such as but not limit to IMU sensors and flex-bent sensors and etc) located on a frame that moves together with user's finger and possibly also hand(palm), wrist, arm, such sensor system(s) might be part of the wearable device in 1) or could be an independent wearable. // both situation

It is also desirable that a “tactile/pressure”(or other “feelable” feedback type) output system is also provided in the UI system (for the purpose such as but not limit to direct feedback, control-location indicating, control surface simulation and etc.) ;

optionally a visual representation of the “physical device/instrument” being simulated (or a part of it such as (vicinity) area that user’s hands/fingers and etc. are interacting with) is presented to user (for example for calibration or correction of key stroke);

So that the (spatial) relationship between fingers (especially areas with feelable feedback actuators such as finger tip) to the key(s)/instrument operating area(s) can be determined from the sensor system data (for example those from IMU’s acceleration and pose data, magnetic “direction” data (or possibly from the “hand skeleton model” which capture and integrate these data), and maybe with other data from additional bent/flex sensors, for (optionally) optical sensors, (or the combine of such data to get a more reliable location); such data might determine if a control is touched, pressed or other situations for example half pressed, pressed multiple controls/keys at the same time, and etc., and certain events/messages can be generated or corresponding procedures/process is executed;

so that to the message/event recipient software/User Interface system or application(s) the output of such interaction of user with virtual device is the same as with a physical (real) device, and from user’s perspective the interaction with the simulated input device is similar/close to that of the real device in aspect of functionality, the skills/gesture required for interaction (moving the hand/arm/wrist/finger the same or similar way), and feedback (visually and maybe together with tactile such as pressure/tactile pattern).

In a related embodiment, said spatial relationship data could be from relative positional or movement data, or it could be calculated from the location of each moving section or from the position object interacting with.

In a related embodiment, providing “direct”/multi-modal feedback including providing (“specific” or “corresponding”)“response” or “feedback” (such as but not limited to: pressure/tactile patterns, vibration patterns/strength/frequency changes etc.) to user’s finger tip with **tactile means** that located on the part of the wearable accommodating finger tip (and other parts) based on the (“specific”) spatial relationship between finger and the virtual object(keys), for example , one activated “spot”/”fixel” in the pattern could means (barely) contacting the virtual object (at such location, might be off centered) but not pressed;

In a related embodiment, optical system can be used in addition to IMU for tracking the hand-based wearable (so that the accuracy can be improved and errors specific to each individual methods can be reduced by referring to the other methods, using ways such as but not limited to Kalman filtering) such as but not limit to glove-like wearable, comprised of:

A least one camera that can be placed on users HMD/Goggle/glasses/AR glasses, or utilizing the camera already on those devices weared by user, or at least one camera weared by user such as around user's neck or in front of user's body (like a body cam) that is able to "see" user's hand in most cases;

At least 2 desirably infrared beacons (such as IR LEDs) on wearable (such as on the sides of the glove-like wearable) that might be controlled/modulated to work together with the camera (which might equip filters allowing light signals from the beacon such as IR lights to pass and might filter out other lights such as in "visible" specturm);
Using optical motion/position tracking algorithms on the signals acquired by the camera to determine the direction and distance of the wearable with the beacon to the camera, thus determine the relative position to user's body/head -- this can use for example but not limited to using H.264 motion vector detection (could e native by the camera) for fast detemination of movement of the beacon or other image processing/motion detection and capturing/tracking algorithm for the images;
It is also possible when using stereo camera, pairs of image can be captured to determine the relative position of the wearable (or the beacons on it);
In related embodiment each beacon light might be
1) modulated or
2) have different pattern/shape
3) having different color/wave length in IR (such as one is 850nm, an other is 900 nm) or other "visual" differences detectable by the camera,
so that from the image or sequence of images captured by the camera it is easy to determine which part of image is corresponding to which beacon (since there could be more than one);

and from size of the becon and the distance between them the the distance and relative position from beacons to camera can be established, which can be used for capturing motion or position of the wearable (such as but not limited to accommodating user's hand fingers).

A computer-implemented method (for human centric "Hyper UI") performed by one or more processors onboard a wearable device including:
receiving sensor data from one or more sensors on at least one wearable gesture/pose/motion input device that can accommodate and move together with at least one (section of) moving part of user's body or limbs such as (but not limit to): finger(s), palm(s), hand(s), wrist(s),arm(s) and etc , the sensor data indicative of a gesture/pose/movement of the part of user being measured (maybe relative to a reference coordinate frame);

Determining the (spatial) relationship between fingers (especially areas with feelable feedback actuators such as finger tip) to the key(s)/instrument operating area(s) from the sensor system data (for example those from IMU's acceleration and pose data, magnetic "direction" data (or possibly from the "hand skeleton model" which capture and integrate these data), and maybe with other data from additional bent/flex sensors, for (optionally) optical sensors, (or the combine of such data to get a more reliable location); such data might determine if a control is touched, pressed or other situations for example half pressed, pressed multiple controls/keys at the same time, and etc., and certain events/messages can be generated or corresponding procedures/process is executed;

desirably also (optionally) presenting a visual representation of the "physical device/instrument" being simulated (or a part of it such as (vicinity) area that user's hands/fingers and etc. are interacting with) is presented to user (for example for calibration or correction of key stroke);

communicate/connecting with other (necessary) devices/systems (user desired/selected to interact with or related to an object displayed on the on-demand display) wirelessly;

In a related embodiment, said spatial relationship data could be from relative positional or movement data, or it could be calculated from the location of each moving section or from the position object interacting with.

In a related embodiment, providing "direct"/multi-modal feedback including providing ("specific" or "corresponding")"response" or "feedback" (such as but not limited to: pressure/tactile patterns, vibration patterns/strength/frequency changes etc.) to user's finger tip with **tactile means** that located on the part of the wearable accommodating finger tip (and other parts) based on the ("specific") spatial relationship between finger and the virual object(keys), for example , one activated "spot"/"fixel" in the pattern could means (barely) contacting the virtual object (at such location, might be off centered) but not pressed,

1. **Application, run other devices such as iwatch simulating a music instruments (most threaten are those that do not need a tactor)**
   It is desirable a wearable device such as a smart watch employ such virtulization technology to simulate physical devices that interact with hands such as but not limit to music instruments, keyboard/key pad, remote control, joystick, mouse and etc.

   It is also desirable a software/application stored and executed by a wearable device (such as a smart watch) use/integrate with virtulization technology to simulate physical devices that interact with hands such as but not limit to music instruments, keyboard/key pad, remote control, joystick, mouse and etc.

In a related Methods for iPhone/iWatch keyboard/game controller to see the virtual keyboard: Identify a key-reaching(-over) gesture of a finger by comparing the gesture(static) or movement of the 10 fingers (find out which one is moving towards which direction), and when detecting a "press" gesture, first give out a "touched" feedback(tactile) and display on the screen the key user is touched but not pressed, then when user press down that key then give more feedback on tactile and the visual.

(--A (software system, hardware platform use input/output device virtualization for integration multiple device, switching --)

A input/output device virtualization technology that allow multiple different input devices being simulated and capable of instant switching between these simulated devices/methods, and high speed parallel input (that could also serve as an integration point with multiple devices) comprised of:

(so that) to allows user to interact with multiple (simulated) devices (some could be at the same time) and optionally provide fast switching between the simulated input devices (that fits for the UI or user picked) and without the need for any physical controller changing. This could includes, 1) establishing communication channels and 2) providing visual/tactile feedback to simulate an appropriate/designated device (device to human direction) or "output" and 3) provides ("high speed/parallel" intuitive) human to machine direction of interaction (input). A controller/processing unit(s) on board of the mobile/wearable -- or in smart watch that collaborating such wearable -- may further coordinating "Task management/scheduling" and "communication" for devices as well as "copy/paste" with multiple endpoints and the related software running on it, which could include 3rd party app (running on each) could provide further integration of the devices such as using a "unified GUI" instead of individual UIs.

It is desirable a computer system (such as but not limited to wearable device such as a smart watch, smart glove or mobile device and etc.), employing such virtualization technology to communicate with multiple remote devices that and provide interaction for user with hands with **corresponding** (could be multiple different one) input methods with the devices user is interacting with. For example, mouse simulation for a desktop, touch screen interaction with a tablet, joystick simulation for a game console, or gesture input (such as simulating turning a knob) for a IOT device such as a smart thermostat.

(-- Hard ware features –specific to glove)

In an embodiment, a wearable device such as a glove or watch have a brace around the wrist (or part of hand if is a small glove) and a conduit inside the brace formed a looped shape antenna and connects to the wearable (or chip inside the brace), which is directional. Inside the wearable device there's another antenna with its plan perpendicular to the plane of this loop, so that when the "internal" antenna have good reception while the signal from this antenna (might be 2 different frequencies) become weakest (null) it means the loop is pointing to the direction, and such information can be used to determine/identify which device user is pointing at, and make it the selected device for the wearable device to communicate/interact with.

In a related embodiment, such wearable could be modular design and could be dissembled, and certain parts (such as index finger part) can be switched.
For example, the "finger tip" or "finger" portion of the glove can be removed or replaced. There could be multiple designs for modularity such as but not limit to: 1) A 3 part design, with removable finger (tip) portion (with sensors and possibly actuators) with

connectors connecting to the "base"/palm part of the glove(with sensors, controllers and power source), a tactile "plate" that is basically the "palm" side of the glove with actuators on it (and could be optional), and a base portion accommodating the palm with controllers, communication, power on the back of the hand, and capable to connect to the other 2 part. Or an optional design could be 2 part design like depicted in Fig 1 (a) and (b) : the "finger" portion could be removed have sensors (such as motion/position sensors such as but not limit to IMU sensors, flex-bent sensors) and (optionally) also with actuators (such as tactile actuator array) in them and could have long connectors (electronic, and could also with pneumatic tube/hoses depends on the actuator type in the finger portion), and might also have flex sensors. By changing to different finger tip with different configuration of sensors and actuators, the "interactive" type of the glove can be changed from non-tactile to tactile/pressure type and etc. Yet another 2 part design like depicted in Fig.1 (c) is also possible where the glove can be "split" into an upper and a lower portion, connect with connectors where the upper portion could house most sensors (and the controller on the back of the hand), the lower portion are more devoted to "feedback" and could house mst of the actuators.

In a related embodiment the "skin" or lining of the glove can be changed.

Like depicted in Fig2, the wearable could adopt a "partial glove" design where motion/position data are only gathered for a few fingers (such as 3 including the index finger) with "ring" like (could be flexible/elastic) "brace" accommodated the finger and connects to a controller on the central "patch" located on the back of the hand (which contain motion/sensors such as IMU sensors for hand position/motion information); and desirably the rings/brace or patch could have button/touchable areas on it to allow additional UI function in interaction.
In a related embodiment the connector between the central "patch" and the "brace"/ring on finger could contain additional flex/bent sensors (such as resistors) to provide additional gesture information.

In an embodiment of a wearable for gesture input related to all embodiments above, further including an earphone/speaker located on the part covering thumb, and a microphone located on the part accommodating either picky or the finger next to it (the "4th" finger from thumb), so that when user make a "telephone gesture" (curved fingers with thumb sticking up and possible picky finger also out and pointing downwards) close to user's face, the earphone/speaker and the microphone are close enough (to be effective for phone conversation) to user's ear and mouse respectively; the earphone/speaker and microphone is connected to the control system (that may further connect to a communication system) and they maybe activated upon a "telephone" like gesture (optionally the gesture recognition maybe ignoring the pinky finger pose for possible wider adoptions of such gesture)
In a related embodiment, such wearable connects to a mobile communication system (such as on a smart watch or a smart phone) desirably wirelessly to provide voice services (pickup and output of audio signal) for voice based communication (such as for placing a phone call, an internet based voice chat)

In an embodiment of mobile communication system, wearable for gesture input (such as a smart glove) housing abovementioned earphone/speaker and microphone, also equipped with wireless communication system such as but not limit to base band cellular communication system, 3G, 4G mobile communication system (maybe including voice and maybe also together with data services), wifi, and etc.

In a related embodiment, such wearable with micro phone –speaker combination may connect to other devices for voice related services or control.
In a related embodiment, the wearable might also have means for other communication such as but not limit to NFC, Infra-Red, ,Bluetooth, means for supporting IOT protocols .

In a embodiment related (to all of the above), the wearable device could include means/modules wireless communicating with other devices with one or more IOT protocols such as (but not limited to) the following (when communicating with IOT devices) AllJoyn, Thread(networking protocol), OIC (Open Interconnect consortium) , IIC (Industrial Internet Consortium) and etc.

As the suitable systems, means, methods here (such as but not limited to sensors, actuators, controllers etc) may be embodied in a wide variety of forms, some of which may be quite different from those of the disclosed embodiment. Consequently, the specific structural and functional details disclosed herein are merely representative;

Brief description of Drawings
Fig 2 depicted a kind of "minimalist" style of design where only motion/position of 3 fingers are gathered by ring like (could be flexible/elastic) "brace" that connects to a controller on the central "patch" located on the back of the hand (which contain montion/sensors such as IMU sensors for hand position/motion information). The connector between the central "patch" and the "brace"/ring on finger could contain additional flex/bent sensors (such as resistors) to provide additional gesture information, and the rings/brace or patch could have button/touchable areas on it to allow additional UI function.

Fig 4 depicts several scenarios of the hyper UI with on-demand (wearable) displays:
In (a) shows a user's hand interacting with a simulated "knob", which could represent a control of a "remote" device such as a smart (IOT) thermostat 5 feet away, and with it (alternative/wearable) interface displayed on the smart watch user is wearing; User's motion/gesture of turning the "virtual knob" can be captured by the wearable device (the glove) user wearing at the same hand, and it is connected to the smart watch, maybe wirelessly such as via Bluetooth, user's interaction of the control such as turning of the "virtual knob" can change the setting of the remote device and reflected on the smart watch UI; alternatively user could just interact with an app running on this watch and use similar interaction to interact with its "tuning knob" for example, this allows convenient single hand operation and do not need the help from the other hand for related input.
(b) shows an alternative situation where the UI is displayed on a AR glasses that serves as the on-demand screen of "Hyper UI" instead of smart watch screen. (c) shows a VR

display image (pair) that can be shown on a VR HMD (right side image omitted), in which instead of UI a simulated control and representation (or the image) of the hand is shown to user in the VR environment so that user can directly see the knob being adjusted when his/her hand turns.

Claims

1. A computer-implemented method performed by one or more processors for providing user with a “force-like” remote interaction (similar to the “force” interaction depicted in the movie “star wars”) with “remote” objects/targets/endpoints outside of haptic range (immediate reach of hands) in reality or in other simulated cases (such as on screen or in VR/AR/MR environment, in video games/simulation) including:
Determining when the interaction starts;
Then determining which remote object(s)/target(s)/endpoint(s) user is intended to pick/select or interact with.

2 In a method related to claim 1, said "determining when the interaction starts" including multiple ways, such as triggered by software events (when entering certain mode such as but not limited to: entering "hero" mode with pickup in games such as Star wars Battle front), by events triggered from user gesture changes such as detecting a "force-like" preparation gesture and/or with detection of Myoelectric of muscle (such as hand/fingers muscle activation detected via EMG/MMG) in conjunction with said a "force-like" preparation gesture, such gesture/gesture change in closely resemble the gesture for start using force as depicted in the related movie and game (such as star wars battle front) -- for example , by detecting user's hand's short distance movement (no longer than 3 feet) in the direction that is roughly perpendicular to the plane of the palm facing away from user's body , with little or no rotation of the palm, and the palm's finger remain in a gesture that is close to the "force" usage gesture as depicted in the movie star wars, when such movement is detected (by ways such as but not limit to detecting user's palm's angle/pose/gesture and movement as well as each fingers (or major fingers) curvature and/or relative angle/pose/gesture by using inertia and/or optical sensors that might be located on the glove-like wearable (in case of IMU/curvature sensors) or external (such as optical);
or by detecting a “force like selection” initial gesture by determining if the following condition(s) are met: 1) The palm is facing away from user or the back of the hand is facing towards user, 2) the hand gesture is a relatively open position like depicted in Fig.3 position 1 (solid line) where the fingers and thumb are not significantly flexed/bent or closely adjacent (closely adjacent means side by side with no space between them);
Upon such “force like selection” initial gesture is detected, determining the general pointing direction of this gesture (can be calculated using algorithms such as but not limit to by “averaging” the vector direction of the five fingers and get a general pointing direction (one way is just to use vector add to add all (normalized) directional vector of available fingers with possible different (configurable) “weight” for different fingers, and determine the direction in space), such averaging algorithm can have variations such as using different “weight” for thumb, and all other fingers, might take into account of the palm facing direction, the “degree of curving” of palm –which should be relatively not curved much, and maybe “degree of curving” of fingers, etc.);
In case there’s also data for user’s head facing or looking direction available, it is optionally that the control system may further filter out those situations where the general pointing direction of the this gesture is not at the direction user head is looking at (this

could be an option and if related situation not filtered out, will allow user to interact with remote objects they are not facing/looking at);

3. In a method related to claim 1, determining which remote object(s)/target(s)/endpoint(s) user is intended to pick/select or interact with including using ways such as but not limited to:
1) In game/simulation, select objects/targets within effective range (distance from user and angle from user's palm facing direction) of the "force interaction" from user (which might be determined by the game/simulation)
or 2) Search a range in user's general pointing direction (for example in the center areas of the search "cone" defined by the general pointing direction, or defined by the individual direction of fingers, and determine if an object is in range (in this "cone") (and might provide feedback such as highlight, zoom in and/or tactile). The method for detecting/identifying a "remote" device that is being pointed can be similar (but not limit to) those methods discussed above in the A. "Point and select" style of remote device selection/interaction section (if the device being searched is in real world) or perform "virtual object" search like ray tracing or other kind of object finding methods (such as those provided by 3D/game engine) to find out the virtual object user is pointing at in case the interaction setting is in the virtual world (such as game, simulation,3D software/OS and etc)/on GUI;
If user keep this gesture of the hand and begin "scanning" (changing the hand/wrist/arm direction), the system will update related directional pointing region and determine if an remote object is in range. And might provide feedback such as highlighting (visually) or tactile/pressure feedback(tactile) to indicate an object/target is in range.

4. In a method related to claim 1, by detecting the change from user's "force like selection" initial gesture to "force like attracting gesture" such as depicted by the position 2 (doted) in Fig. 3, in ways such as but not limit to: determine the "openness" of the gesture from a "wider open angle" to a "narrower open angle" (the limits can be configurable or customizable). Such as in ways (but not limit to):

3) calculate the relative angle (scalar value) of each finger's (new) pointing direction to the (new) averaged pointing direction(vector) or the "general pointing direction" of the gesture as mentioned above. If all 5(or available) values are decreasing to an extent not yet at "fully closing" level (for example at a range close to "halfway" of open pose and fully closed position, or "half grab" position and etc., and desirably the range and/or detecting mechanism configurable/customizable) in a "similar fashion" (for example not in a case one or 2 value or "change" ratio significantly different than the averaged value or ration, for example more than 30%) , or at least the values are not moving in a "contradicting way"—some of the fingers are "closing" while some other(s) are "opening"-- which signifies a change of gesture or context), then the "attracting gesture" can be determined/related message/events can be triggered; or,
4) Determined by the measurement of the "degree of curving/bent" of the 5(or available) fingers such as but not limit to from the data of the "bent/flex" sensor of the fingers, when the "degree of curving/bent" are increased for all related fingers to an extent not yet at "fully closing" level (such as "halfway" of open

pose and fully closed position, or "half grab" position , and limit/range could be configurable, and desirably all finger moved in a similar fashion, for example but not limit to, all the changed ratios of bent/curving are within 25% (or a configurable value) of each other or the average value, or at least not the values are not moving in a "contradicting way"—some of the fingers are "closing" while some other(s) are "opening"-- which signifies a change of gesture or context) , then the "attracting gesture" can be determined/related message/events can be triggered;

So when the conditions mentioned above is satisfied and the "averaged" open angle of the fingers (on that hand) changed to a smaller value under a limit (compare to the "averaged" open angle of the fingers in "force like selection") then we can determine that this is the "force like attraction" gesture, the current "remote object" in the "selection cone" becoming "selected" , and maybe events/message can be generated for subsequent processing of the remote object selected;

Similarly when detected, a "grab" gesture similar to normal grab action of the hand (all finger closed) can also be used for signaling certain meaning such as grab an object, document or content (such as in case of virtual world/UI) or in real life interacting with remote devices this might be used to signify grabbing a "UI" or document/content(when applicable) from the "remote" device user interacting with to a local wearable, mobile or "on-demand" device, the current "remote object" in the "selection cone" becoming "selected" , and maybe events/message can be generated for subsequent processing of the remote object selected;

Some subsequent processing could be for example (when in a virtual/simulated environment such as in a game or in a 3D GUI), the object might be rendered as "attracted" with the "force like attracting gesture" and appear to directly fly into user's hand on the wearable/on demand display, or copied/moved (in case of copy the content onto the clipboard) with a grab gesture; In a real world "remote device" situation, the UI of the device or the might be picked and (selectively) display on a wearable screen or "on-demand" screen user can see, or the previously selected file/message/content on the remote device can be put to the clipboard with the "grab" gesture.

5. An input/output device virtualization system simulating a hardware input device that allow user to operate it with similar gesture/hand movement/pose to comprised of:

1) One or a group of cooperating wearable means/device that user worn on the part of the body related to the operation of one or multiple input device being simulated, such as hand/arm/wrist with user that can communicate with other devices/systems desirably wirelessly such as but not limit to RF(such as Bluetooth, NFC, Wifi and etc), IR, ultrasound and etc.;

2) At least one (3D) gesture/hand pose/motion sensor system/means as (primary) input, this could be for example a series of motion/position and related sensors (such as but not limit to IMU sensors and flex-bent sensors and etc) located on a frame that moves together with user's finger and possibly also hand(palm), wrist, arm, such sensor system(s) might be part of the wearable device in 1) or could be an independent wearable. // both situation

It is also desirable that a "tactile/pressure"(or other "feelable" feedback type) output system is also provided in the UI system (for the purpose such as but not limit to direct feedback, control-location indicating, control surface simulation and etc.) **;**

optionally a visual representation of the "physical device/instrument" being simulated (or a part of it such as (vicinity) area that user's hands/fingers and etc. are interacting with) is presented to user (for example for calibration or correction of key stroke);

So that the (spatial) relationship between fingers (especially areas with feelable feedback actuators such as finger tip) to the key(s)/instrument operating area(s) can be determined from the sensor system data (for example those from IMU's acceleration and pose data, magnetic "direction" data (or possibly from the "hand skeleton model" which capture and integrate these data), and maybe with other data from additional bent/flex sensors, for (optionally) optical sensors, (or the combine of such data to get a more reliable location); such data might determine if a control is touched, pressed or other situations for example half pressed, pressed multiple controls/keys at the same time, and etc., and certain events/messages can be generated or corresponding procedures/process is executed;

so that to the message/event recipient software/User Interface system or application(s) the output of such interaction of user with virtual device is **the same as with** a physical (real) device, and from user's perspective the interaction with the simulated input device is similar/close to that of the real device in aspect of functionality, the skills/gesture required for interaction (moving the hand/arm/wrist/finger the same or similar way), and feedback (visually and maybe together with tactile such as pressure/tactile pattern).

6. In a system according to Claim 5, said spatial relationship data could be from relative positional or movement data, or it could be calculated from the location of each moving section or from the position object interacting with.

7. In a system according to Claim 5, providing "direct"/multi-modal feedback including providing ("specific" or "corresponding")"response" or "feedback" (such as but not limited to: pressure/tactile patterns, vibration patterns/strength/frequency changes etc.) to user's finger tip with **tactile means** that located on the part of the wearable accommodating finger tip (and other parts) based on the ("specific") spatial relationship between finger and the virtual object(keys), for example , one activated "spot"/"fixel" in the pattern could means (barely) contacting the virtual object (at such location, might be off centered) but not pressed;

8. In a system according to Claim 5, optical system can be used in addition to IMU for tracking the hand-based wearable (so that the accuracy can be improved and errors specific to each individual methods can be reduced by referring to the other methods, using ways such as but not limited to Kalman filtering) such as but not limit to glove-like wearable, comprised of:
A least one camera that can be placed on users HMD/Goggle/glasses/AR glasses, or utilizing the camera already on those devices weared by user, or at least one camera weared by user such as around user's neck or in front of user's body (like a body cam) that is able to "see" user's hand in most cases;
At least 2 desirably infrared beacons (such as IR LEDs) on wearable (such as on the sides of the glove-like wearable) that might be controlled/modulated to work together with the camera (which might equip filters allowing light signals from the beacon such as IR lights to pass and might filter out other lights such as in "visible" specturm);
Using optical motion/position tracking algorithms on the signals acquired by the camera to determine the direction and distance of the wearable with the beacon to the camera, thus determine the relative position to user's body/head -- this can use for example but not limited to using H.264 motion vector detection (could e native by the camera) for fast detemination of movement of the beacon or other image processing/motion detection and capturing/tracking algorithm for the images;
It is also possible when using stereo camera, pairs of image can be captured to determine the relative position of the wearable (or the beacons on it);
In related embodiment each beacon light might be
1) modulated or
2) have different pattern/shape
3) having different color/wave length in IR (such as one is 850nm, an other is 900 nm) or other "visual" differences detectable by the camera,
so that from the image or sequence of images captured by the camera it is easy to determine which part of image is corresponding to which beacon (since there could be more than one);
and from size of the beacon and the distance between them the the distance and relative position from beacons to camera can be established, which can be used for capturing motion or position of the wearable (such as but not limited to accommodating user's hand fingers).

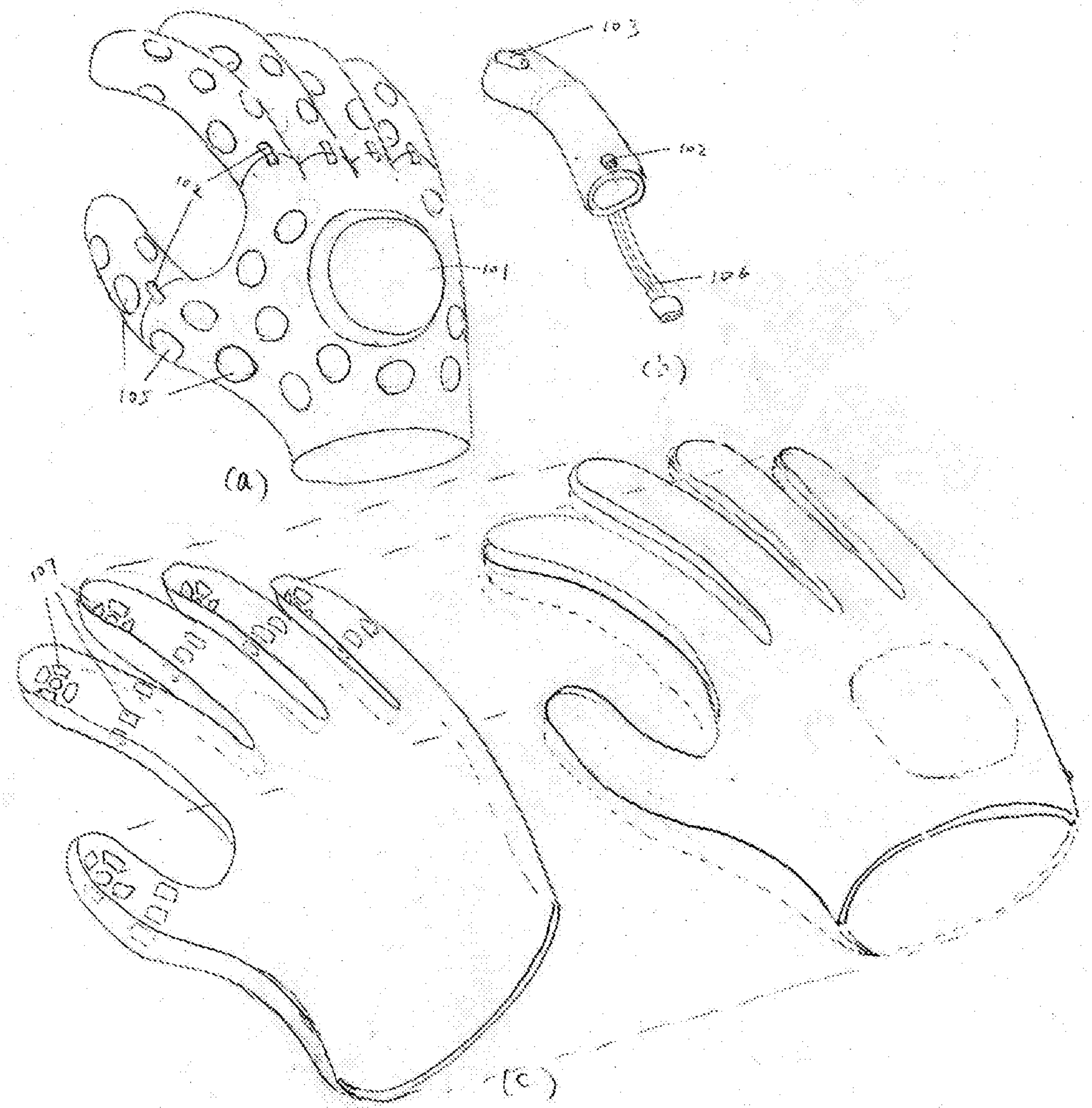

Fig. 1

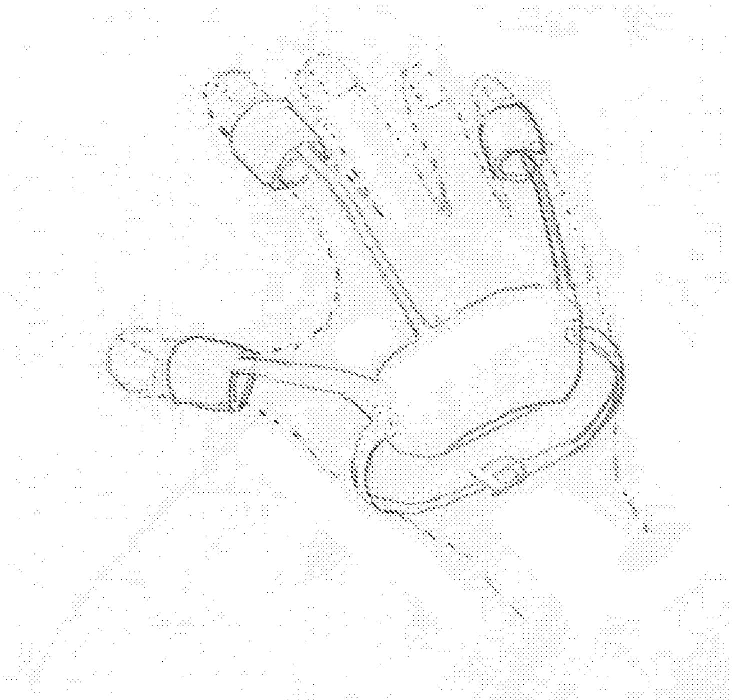

Fig. 2

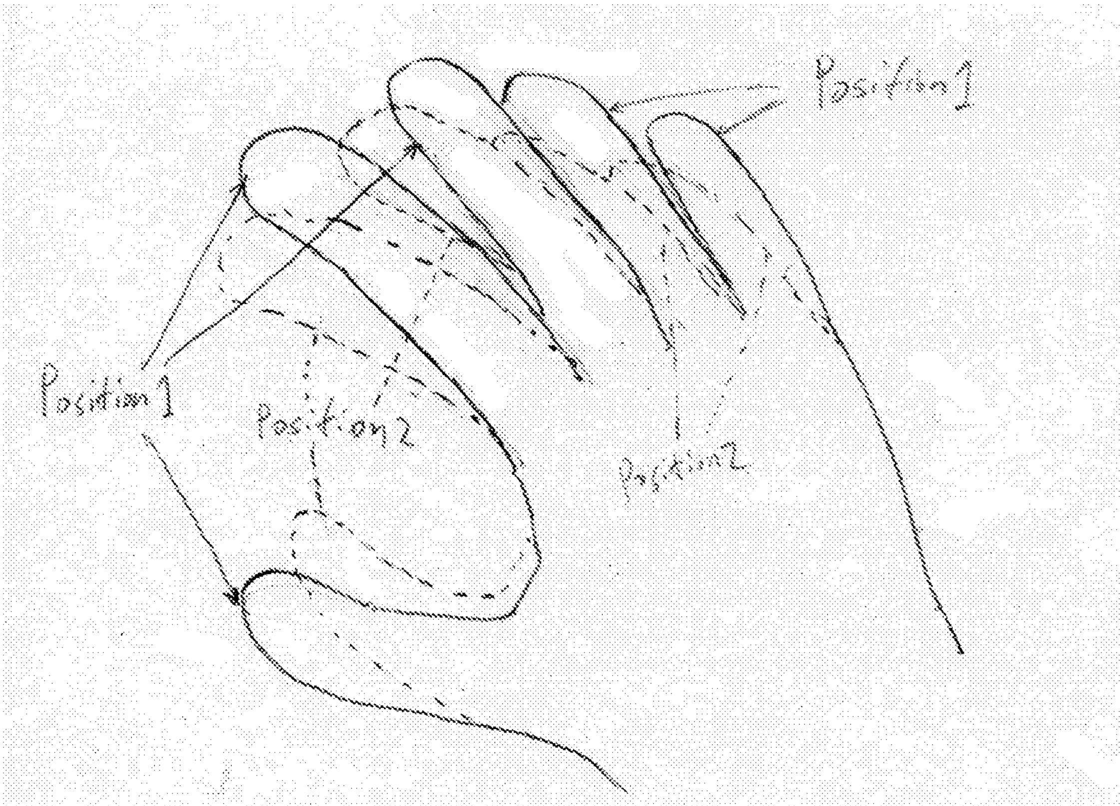

Fig. 3

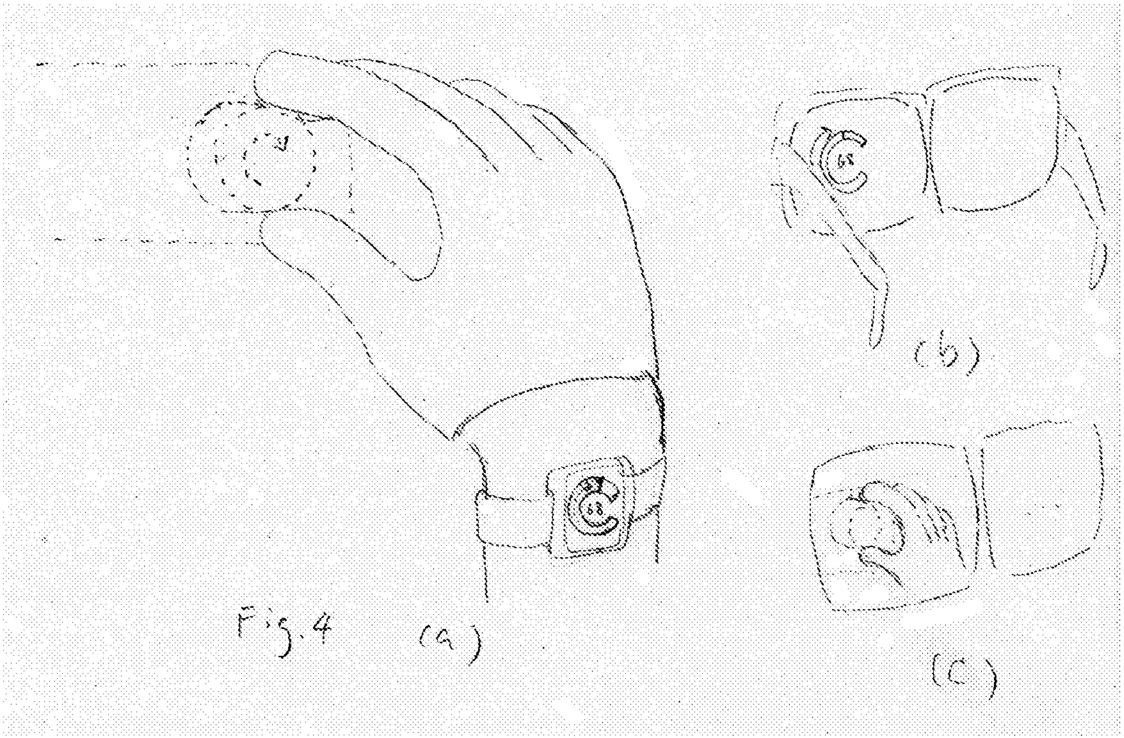

Fig. 4