



(19) **United States**

(12) **Patent Application Publication**  
**Eagleman et al.**

(10) **Pub. No.: US 2018/0067558 A1**  
(43) **Pub. Date: Mar. 8, 2018**

(54) **METHOD AND SYSTEM FOR PROVIDING  
ADJUNCT SENSORY INFORMATION TO A  
USER**

*G09B 21/04* (2006.01)

*A61F 11/04* (2006.01)

(52) **U.S. Cl.**

CPC ..... *G06F 3/016* (2013.01); *G08B 6/00*  
(2013.01); *G10L 15/16* (2013.01); *A61F*  
*11/045* (2013.01); *G09B 21/04* (2013.01)

(71) Applicant: **Texas Medical Center Innovation  
Institute, Houston, TX (US)**

(72) Inventors: **David M. Eagleman, Houston, TX  
(US); Scott D. Novich, Houston, TX  
(US); Danny Goodman, Houston, TX  
(US); Abhipray Sahoo, Houston, TX  
(US); Michael Perotta, Houston, TX  
(US)**

(57) **ABSTRACT**

A method for providing information to a user, the method including: receiving an input signal from a sensing device associated with a sensory modality of the user; generating a preprocessed signal upon preprocessing the input signal with a set of preprocessing operations; extracting a set of features from the preprocessed signal; processing the set of features with a neural network system; mapping outputs of the neural network system to a device domain associated with a device including a distribution of haptic actuators in proximity to the user; and at the distribution of haptic actuators, cooperatively producing a haptic output representative of at least a portion of the input signal, thereby providing information to the user.

(21) Appl. No.: **15/696,997**

(22) Filed: **Sep. 6, 2017**

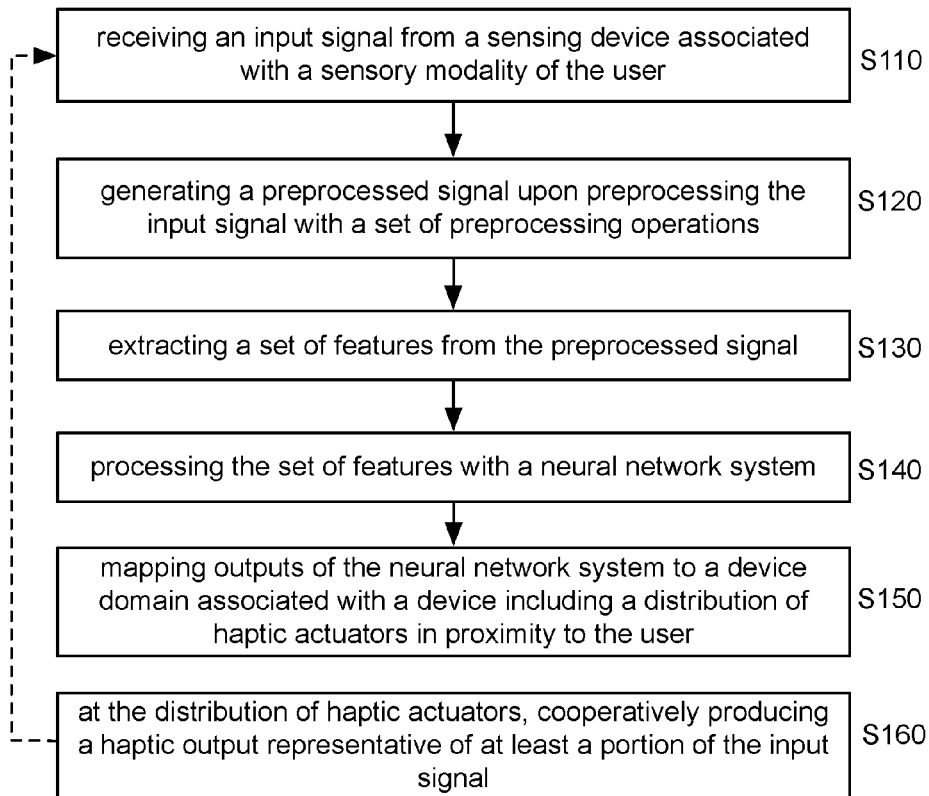
**Related U.S. Application Data**

(60) Provisional application No. 62/384,036, filed on Sep. 6, 2016.

**Publication Classification**

(51) **Int. Cl.**  
*G06F 3/01* (2006.01)  
*G08B 6/00* (2006.01)

100



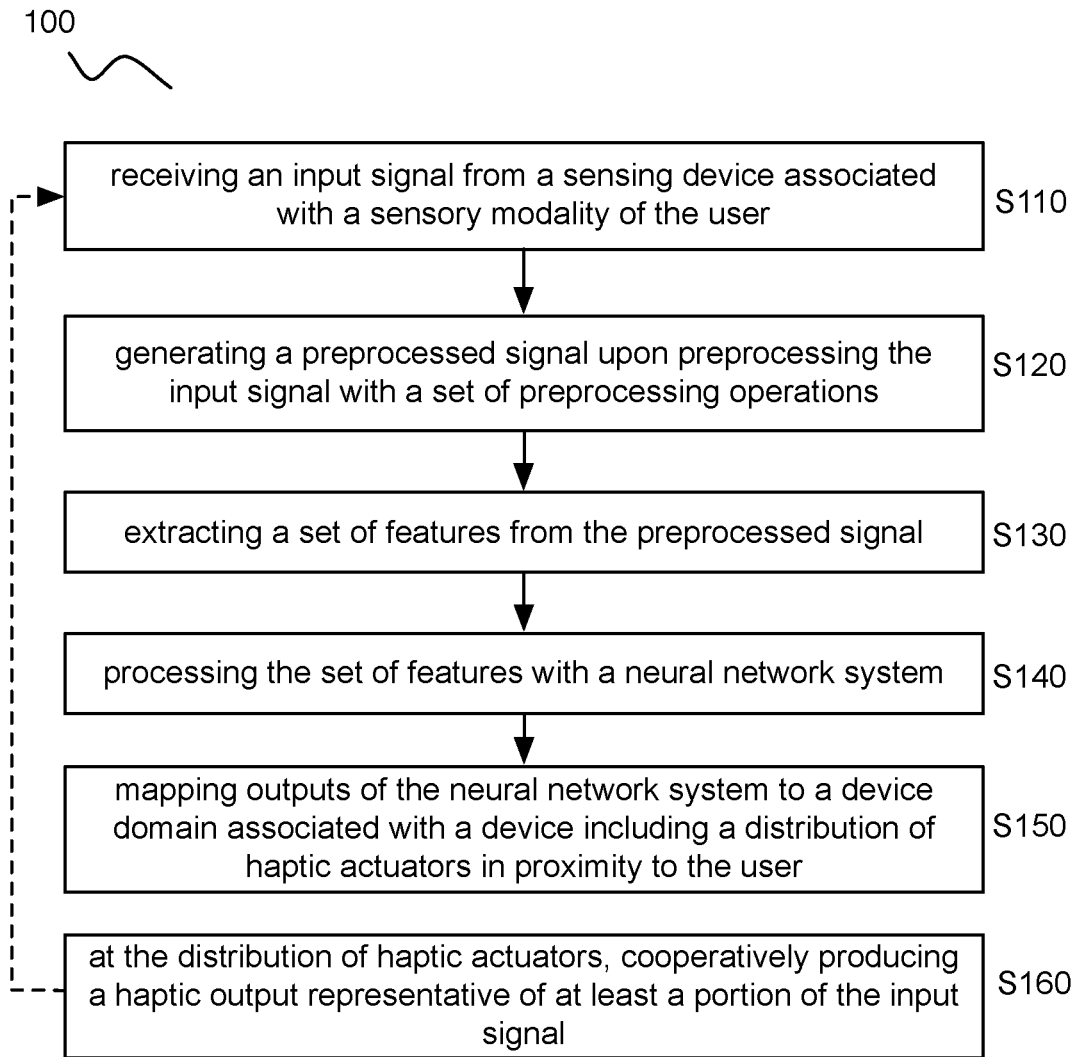


FIGURE 1A

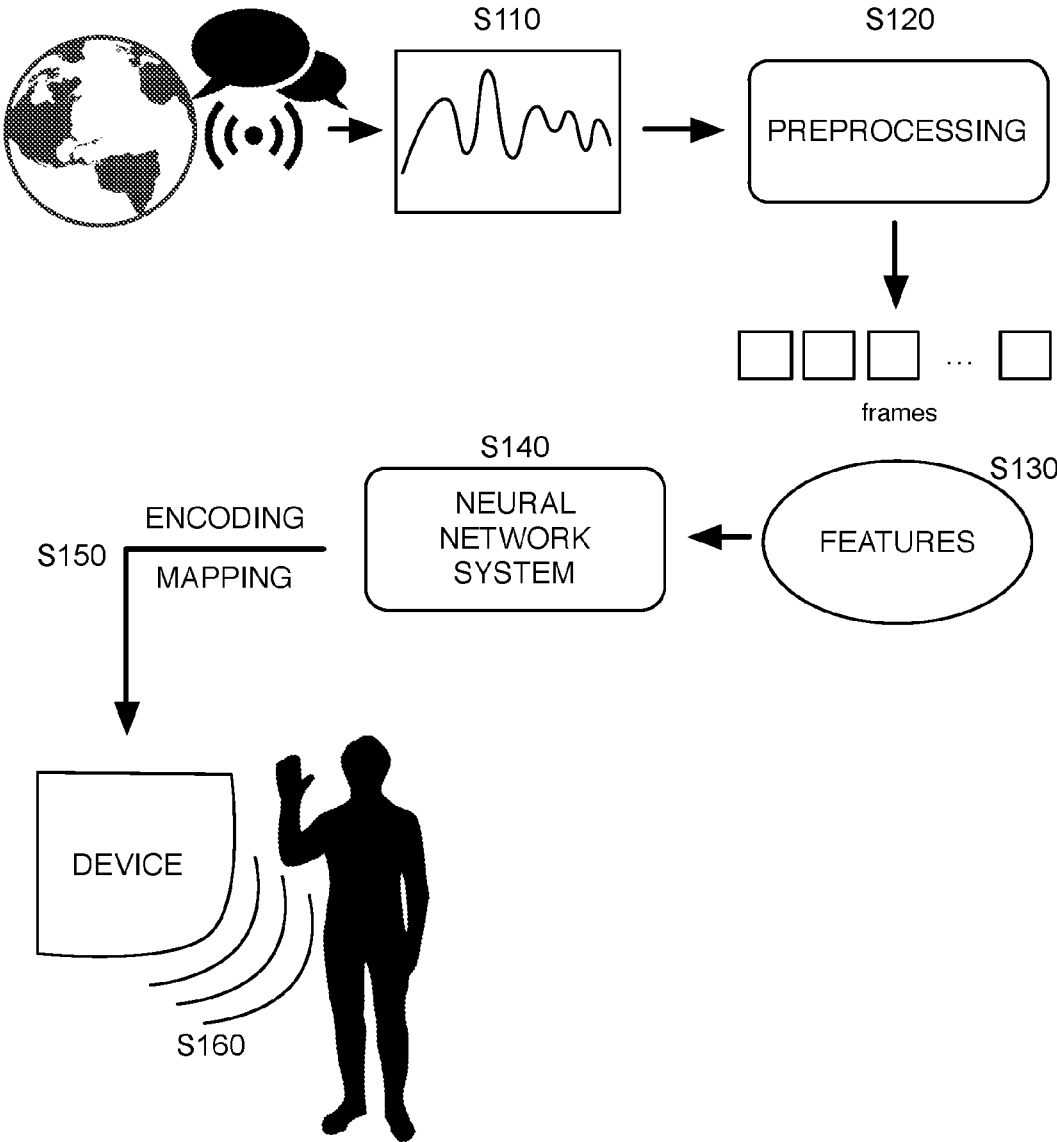


FIGURE 1B

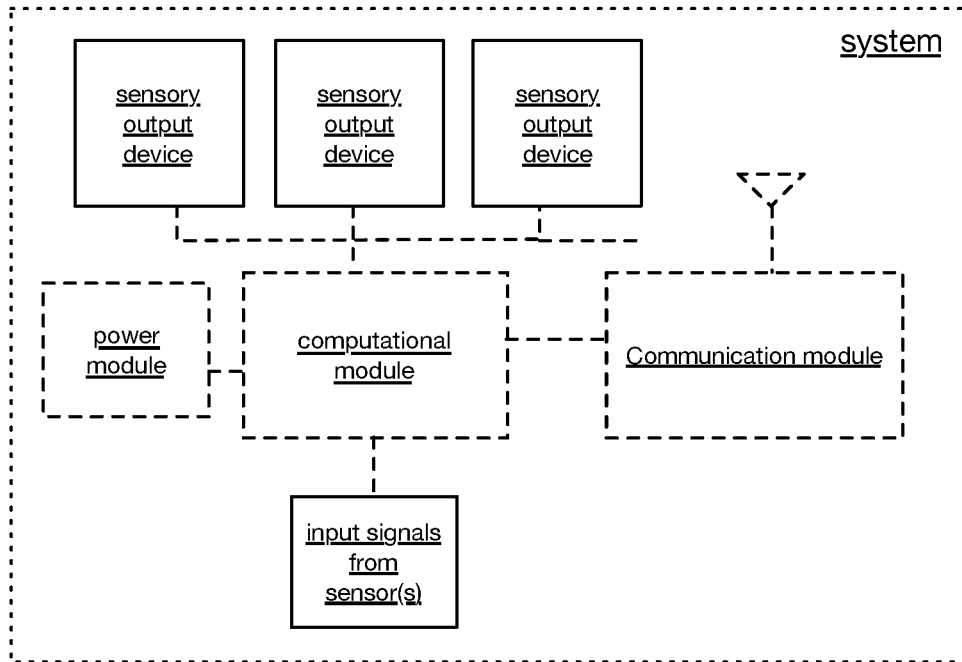


FIGURE 2A

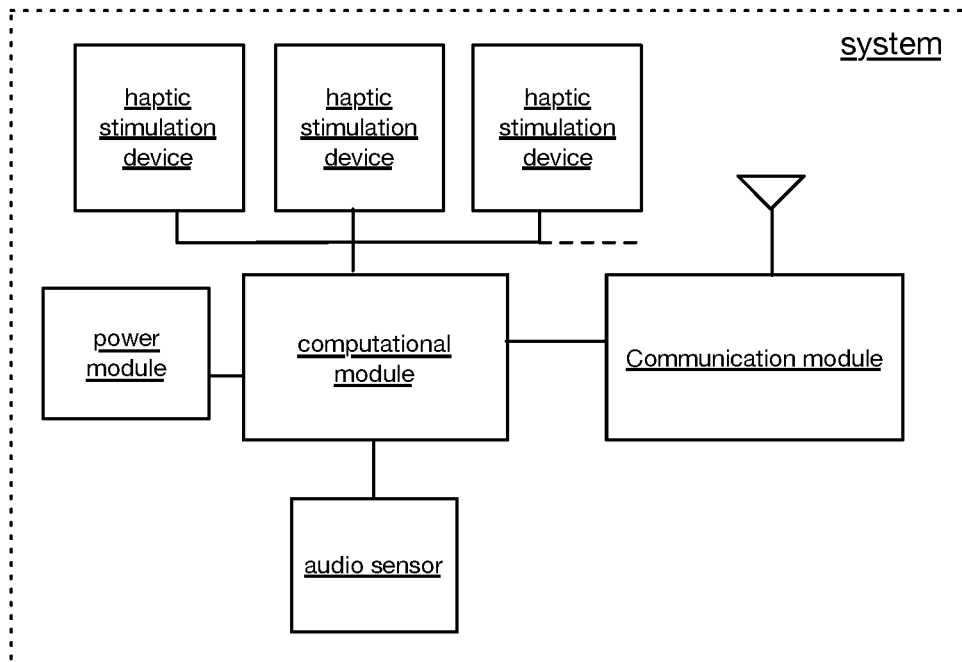


FIGURE 2B

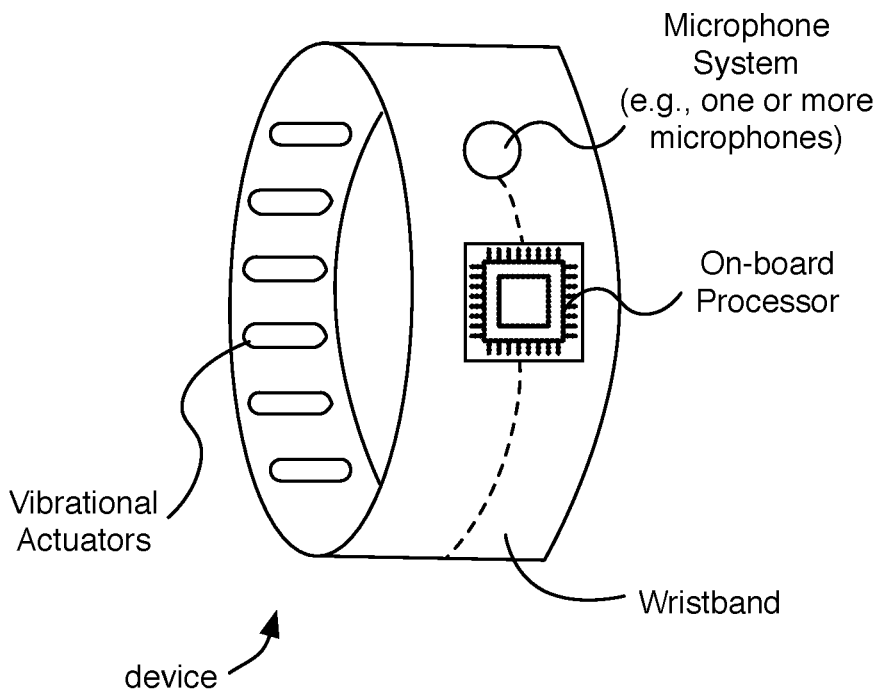


FIGURE 3A

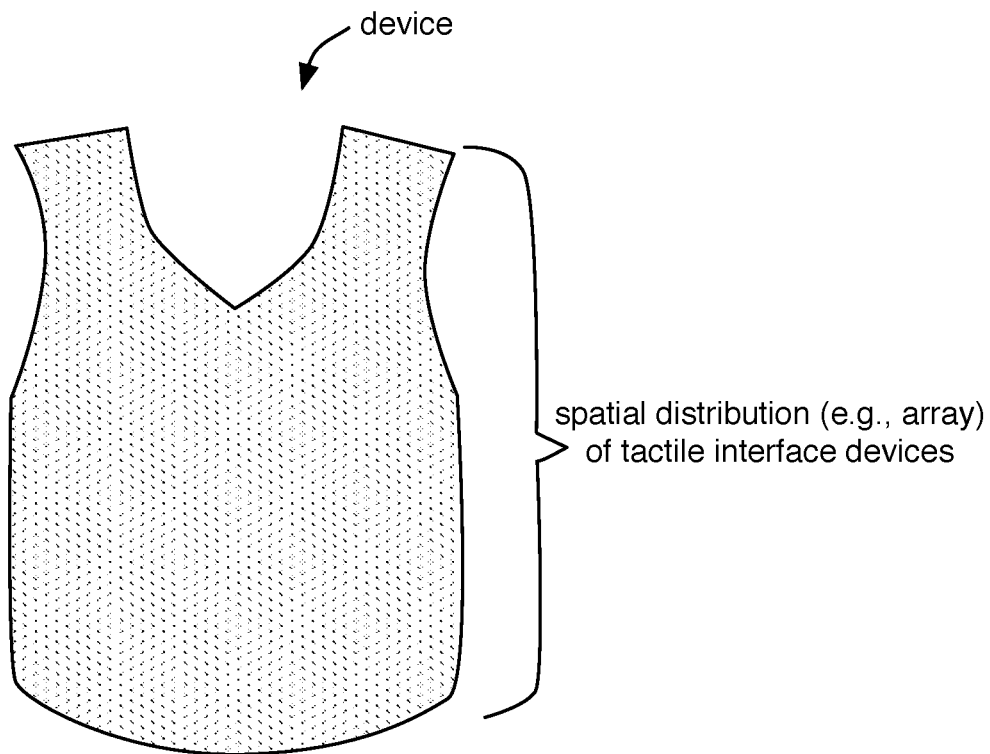


FIGURE 3B

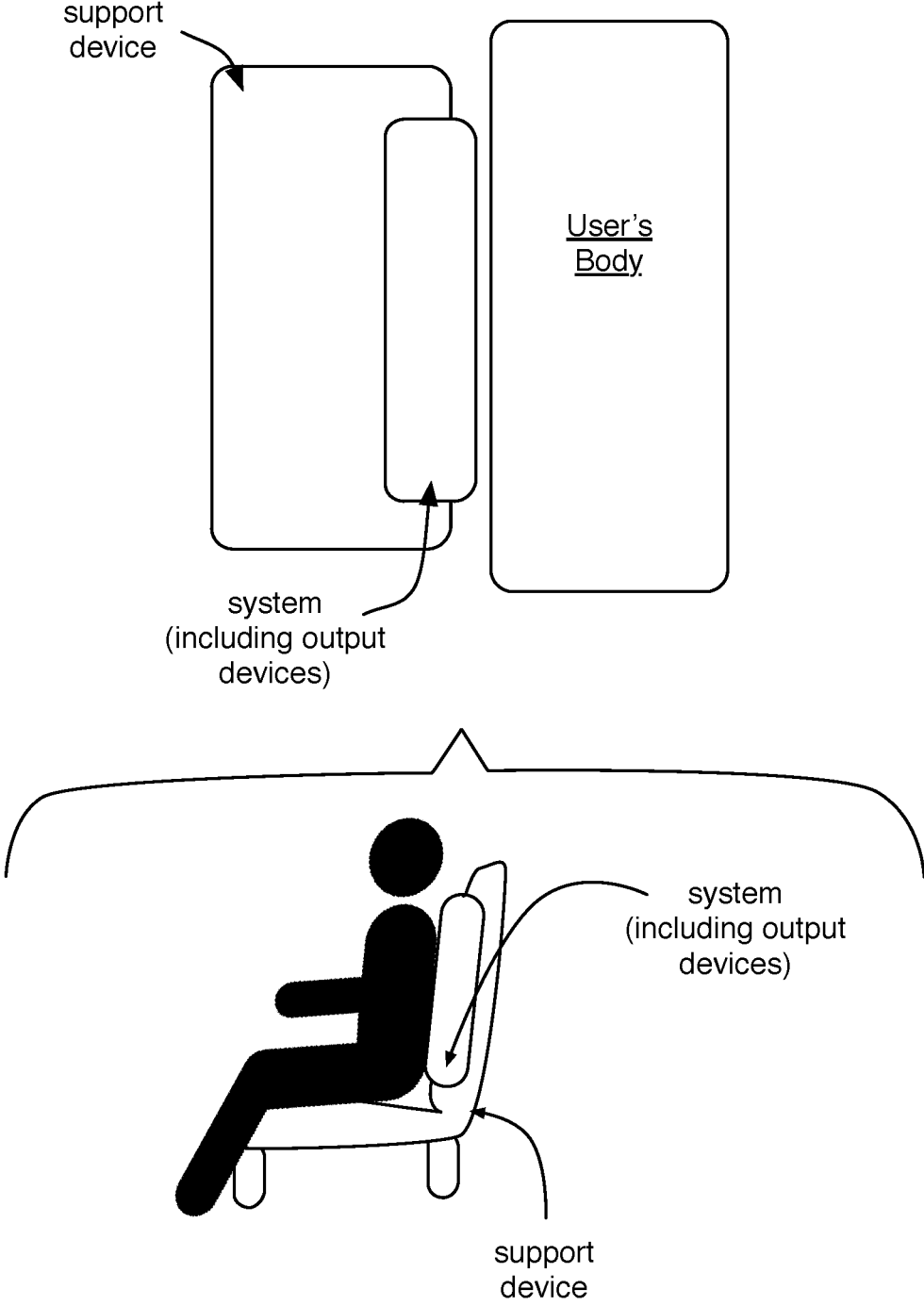


FIGURE 3C

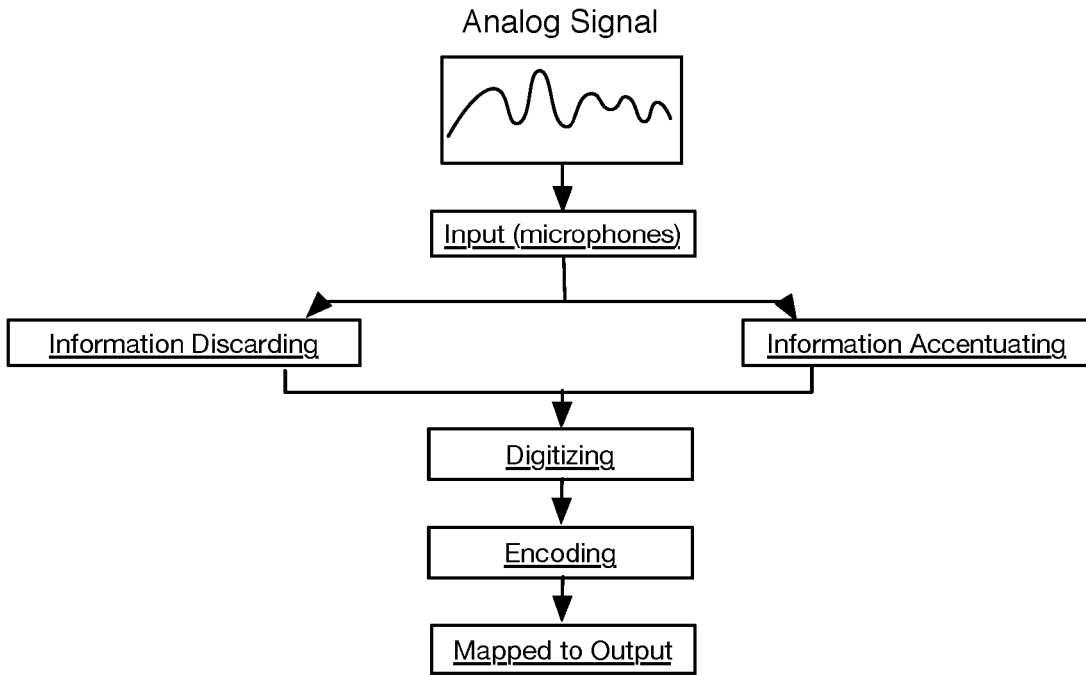


FIGURE 4A

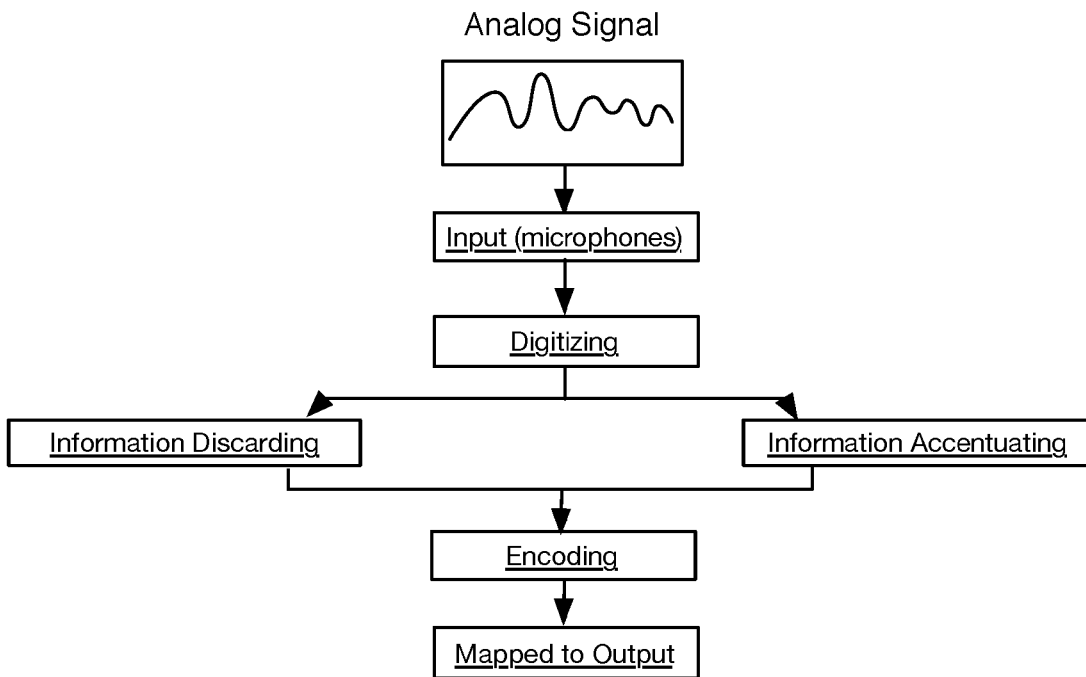


FIGURE 4B

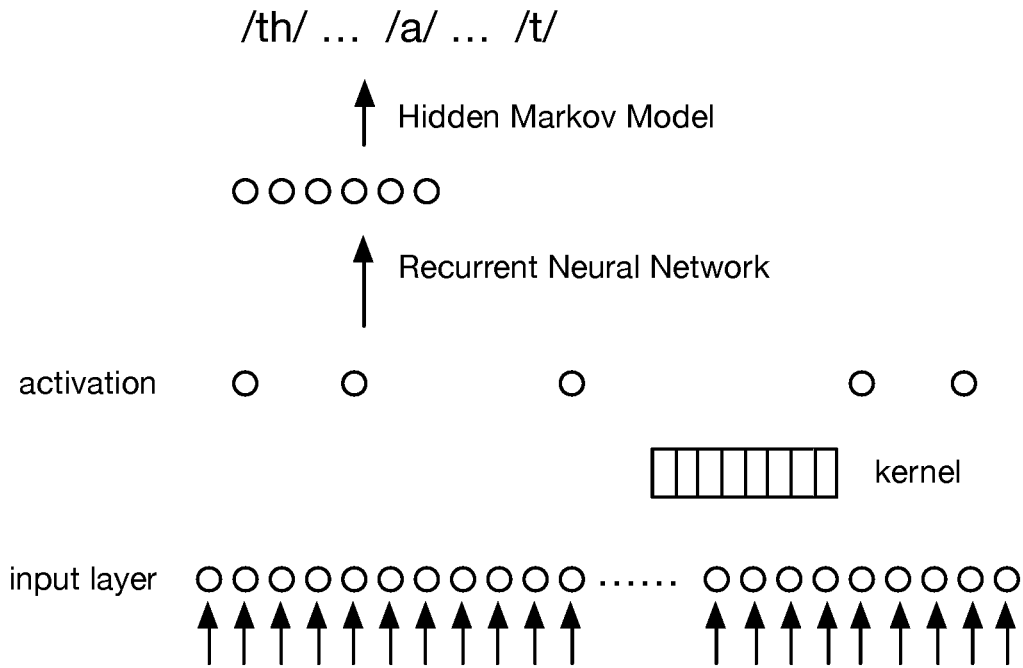


FIGURE 5

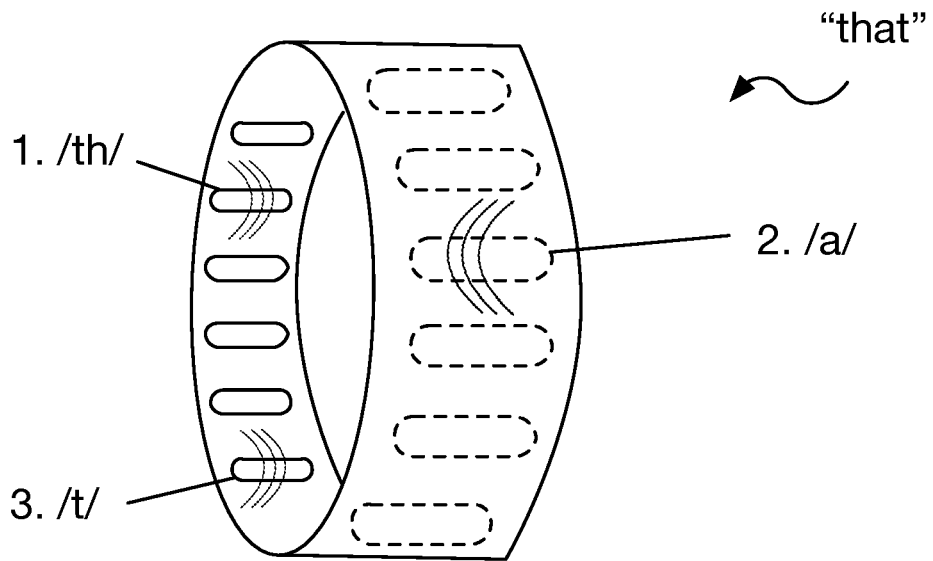


FIGURE 6



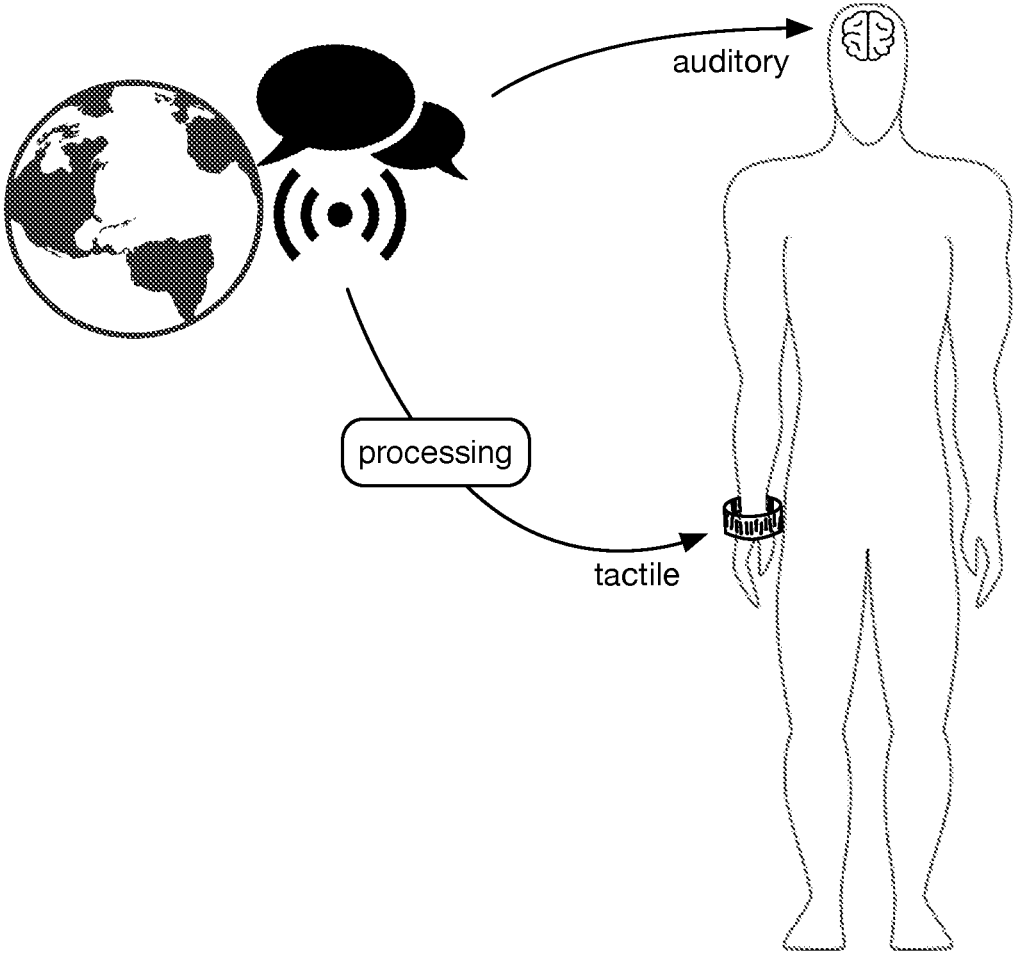


FIGURE 7

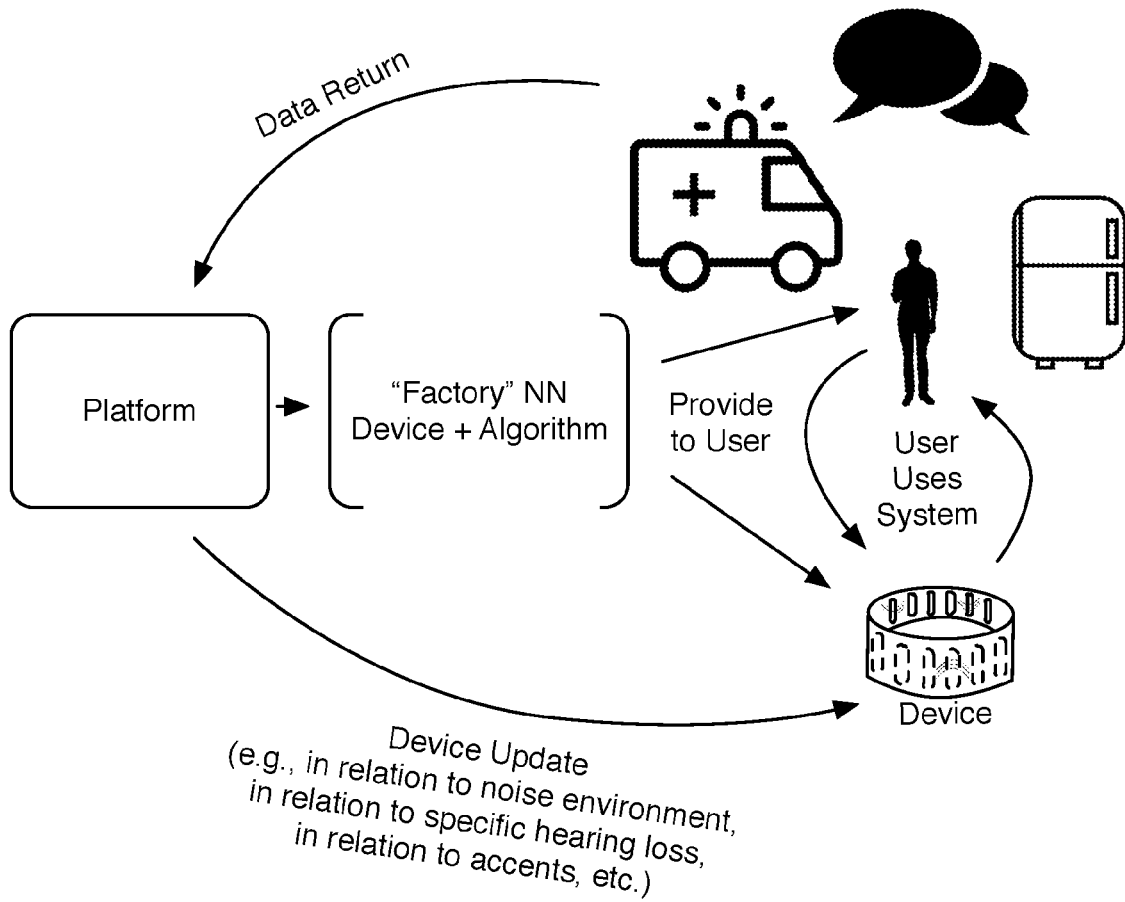


FIGURE 8A

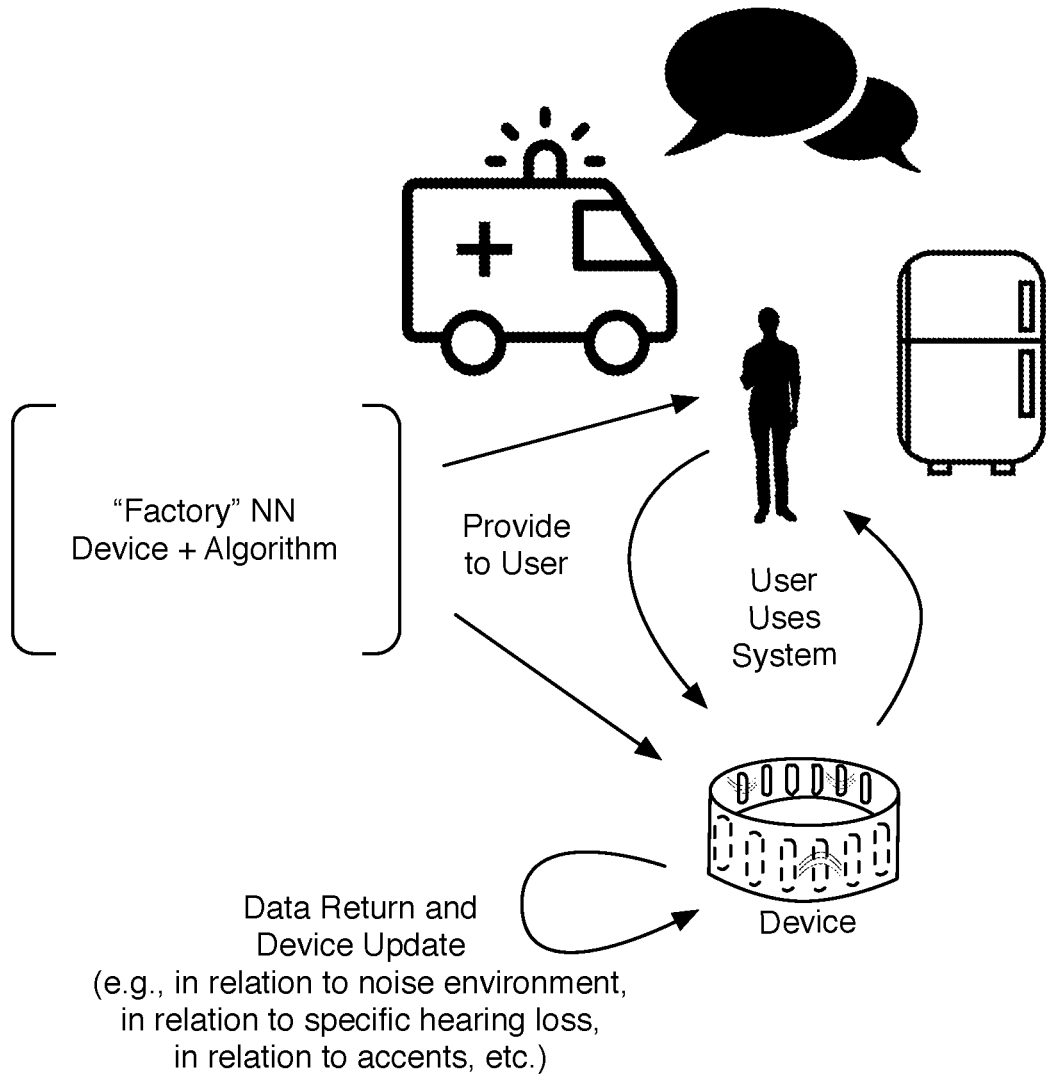


FIGURE 8B

## METHOD AND SYSTEM FOR PROVIDING ADJUNCT SENSORY INFORMATION TO A USER

### CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims the benefit of U.S. Provisional Application Ser. No. 62/384,036, filed on 6 Sep. 2016, which is incorporated in its entirety by this reference.

### TECHNICAL FIELD

[0002] This invention relates generally to the sensory stimulus technology field, and more specifically to a new and useful system and method for providing information to a user.

### BRIEF DESCRIPTION OF THE FIGURES

[0003] FIG. 1A is a flowchart diagram of an embodiment of the method for providing information to a user;

[0004] FIG. 1B is a schematic of an embodiment of a method and system for providing information to a user;

[0005] FIGS. 2A and 2B are schematics of embodiments of systems for providing information to a user;

[0006] FIGS. 3A-3C depict example stimuli providing devices associated with a method and system for providing information to a user;

[0007] FIGS. 4A-4B depict variations of methods for providing information to a user;

[0008] FIG. 5 depicts an example of a neural network architecture associated with a method for providing information to a user;

[0009] FIG. 6 depicts an example of a haptic output representative of speech content information in an embodiment of a method for providing information to a user;

[0010] FIG. 7 depicts a portion of a method for providing information to a user; and

[0011] FIGS. 8A-8B depict variations of method flows involving machine-learning facilitate system updates for improving provision of information to a user.

### DESCRIPTION OF THE PREFERRED EMBODIMENTS

[0012] The following description of the preferred embodiments of the invention is not intended to limit the invention to these preferred embodiments, but rather to enable any person skilled in the art to make and use this invention.

#### 1. Overview.

[0013] As shown in FIGS. 1A and 1B, a method **100** for providing information to a user comprises: receiving an input signal from a sensing device associated with a sensory modality of the user **S110**; generating a preprocessed signal upon preprocessing the input signal with a set of preprocessing operations **S120**; extracting a set of features from the preprocessed signal **S130**; processing the set of features with a neural network system **S140**; mapping outputs of the neural network system to a device domain associated with a device including a distribution of haptic actuators in proximity to the user **S150**; and at the distribution of haptic actuators, cooperatively producing a haptic output representative of at least a portion of the input signal, thereby providing information to the user **S160**. However, the

method **100** can additionally or alternatively include any other suitable elements, and can be performed in any suitable manner.

[0014] The method **100** preferably functions to transform input signals associated with at least one sensory modality (e.g., audio signals containing communication-related information, audio signals containing other information, signals containing information associated with vision, signals containing information associated with the sensation of touch, signals containing information associated with tastes, signals containing information associated with smells, etc.) into stimuli provided using a device in proximity to or otherwise worn by a user, wherein the stimuli are associated with signal processing outputs of a different domain, dimensionality, and/or rate than the input signals. The method **100** can be used for users without sensory conditions (e.g., having sensory sensitivity within a typical range) but for whom receiving of information from multiple sensory sources is desired (e.g., to enhance perception and/or enjoyment of the information). The method **100** can additionally or alternatively allow a user with one or more sensory conditions (e.g., reduced sensory sensitivity, enhanced sensory sensitivity, lacking sensitivity in one or more sensory modalities, etc.) to receive information that would otherwise be received through the one or more senses.

[0015] In variations, the method **100** can thus operate to provide a means for sensory cross-boosting to allow a user to receive and process information, whereby sensory cross-boosting includes a conversion of a subset of information from one sensory modality (e.g., auditory information for hearing) into another sensory modality (e.g., touch). As such, sensory cross-boosting can be used as a sensory adjunct for a user or for partial sensory substitution (e.g., in relation to a user with partial sensory loss), whereby information typically associated with one sensory modality can be provided to the user by way of multiple sensory modalities of the user. Additionally or alternatively, the method can be used for sensory substitution, whereby information typically associated with one sensory modality (e.g., auditory information) is converted to stimuli associated with another sensory modality (e.g., the sensation of touch), such that the user can perceive the content of the information through a sensory modality not typically used to perceive the content of the information.

[0016] In specific examples, the method **100** can be used to provide real-time (or near real-time) understanding of information acquired using audio sensors (e.g., microphones) to a user by way of a wearable device (e.g., torso-coupled wearable device, limb-borne wearable device, head-coupled wearable device, etc.) operable to provide haptic stimuli to the user. In more detail, processing of the input signals (e.g., audio signals) can be adapted to specific demographics (e.g. age demographics, gender demographics, species demographics, etc.) or sensory conditions, and in a specific application, can include processing input audio signals to allow users with high frequency hearing loss to perceive high frequency auditory information using their sensations of touch.

[0017] The method **100** can thus be implemented using system components described in more detail below, and/or using an embodiment, variation, or example of the system described in U.S. application Ser. No. 14/750,626, titled "Providing Information to a User Through Somatosensory Feedback" and filed on 25 Jun. 2015, which is herein

incorporated in its entirety by this reference. However, the method **100** can additionally or alternatively be implemented using any other suitable system or system components for providing information to users through feedback devices.

## 2. Benefits.

**[0018]** The method **100** can confer a number of benefits. First, embodiments of the method **100** can function to transform input signals with characteristic dimensionality and/or speed into stimuli (e.g., stimuli provided to a user using one or more devices as sensory inputs to the user) with different dimensionality and/or speed (e.g., transforming low-dimensionality, high-speed input information such as speech information into higher-dimensionality, lower-speed outputs such as tactile outputs; transforming high-dimensionality and/or high-speed input information into lower-dimensionality and/or lower-speed outputs; etc.). In examples, the relationship can be inversely proportional (e.g., as in sound-to-touch for speech processing, which processes a low dimensional and high speed stream into a high dimensional and low speed stream).

**[0019]** Second, embodiments of the method **100** can implement haptic stimuli-providing devices coupled to one or more users to provide information-rich stimuli to users, based on transformation of input signals associated with one sensory modality to stimuli associated with another sensory modality. In some embodiments, such cross-sensory transformations can aid in perception and/or understanding of the input signals, such as by supplementing and/or replacing the input signals (e.g., associated with a sensory modality for which a user has diminished sensitivity) with related (e.g., complementary, representative, etc.) information delivered via a second sensory modality (e.g., tactile sense). For example, if the input signals include information that cannot easily be provided to the user via the original sensory modality (e.g., due to sensory limitations, output device constraints, etc.), the method **100** can include providing information via the second sensory modality, enabling user perception and/or understanding of the information. In examples, such transformations can be adapted to user-specific needs associated with demographic factors (e.g., age, gender, ethnicity, etc.), health conditions, and/or any other suitable factor(s), whereby components of the information (e.g., high frequency components of speech) associated with the input signals are less readily perceived by users of interest, and stimuli provided by the devices can be modulated to fit the sensory sensitivity (e.g., haptic sensitivity) of the user. Further, in some embodiments, systems associated with such stimuli (e.g., stimulus output devices) can have beneficial form factors and/or modalities of interaction with the user(s). In a specific example, in which the method **100** includes transforming auditory input signals into haptic outputs, the haptic outputs can be provided to the user by a discretely-worn system (e.g., worn under clothing, worn on a body extremity, etc.), whereas some systems configured to provide auditory outputs may include less discrete elements.

**[0020]** Third, embodiments of the method **100** can decrease latency in transmitting information to a user by selecting a subset of information contained in input signals for further processing and, ultimately, transformation into stimuli provided to users, thereby improving performance of processing systems associated with the method **100**.

**[0021]** Thus, embodiments of the method **100** can function to provide platforms and devices for supplementing and/or substituting functions of one or more sensory modalities, in a manner that is adapted to specific user needs. However, the method **100** can additionally or alternatively confer any other suitable benefits.

## 3. System.

**[0022]** The system preferably receives or generates one or more input signals from appropriate sensors, and provides stimuli (e.g., through sensory output devices in proximity to the user), and can optionally include one or more communication modules, power modules, and/or computational modules (e.g., as shown in FIGS. 2A-2B). The device components associated with the stimuli are preferably disposed in a single device, but can additionally or alternatively be disposed across a plurality of devices, and/or be disposed in any other suitable manner.

**[0023]** The stimuli can be provided by a plurality of tactile interface devices (e.g., haptic actuators, electrical stimulators, etc.) in a spatial distribution (e.g., multidimensional spatial distribution), each of which can provide a variety of available output stimuli with different stimulus parameters (e.g., as shown in FIGS. 3A-3B). The device(s) can provide haptic stimuli through the tactile interface devices, and in specific examples, can include an array of tactile interface devices operable to provide configurable haptic stimuli to a user. The tactile interface devices can include vibration motors (e.g., eccentric rotating mass (ERM) devices), Linear Resonant Actuators (LRAs), piezoelectric devices, and/or any other suitable devices (and/or combinations thereof, such as hybrid devices incorporating both ERM and LRA elements).

**[0024]** The device(s) can additionally or alternatively be operable to provide one or more of: auditory stimuli, electrical stimuli (e.g., peripheral stimuli, etc.), olfactory stimuli, taste stimuli, and any other suitable form of stimulus.

**[0025]** The spatial distribution (e.g., array) of tactile interface devices can have a density from 5 devices per  $\text{cm}^2$  to 50 devices per  $\text{cm}^2$ , or any other suitable density. Furthermore, the spatial distribution of tactile interface devices can be configured with any suitable morphological aspects. The tactile interface devices are preferably arranged in one or more arrays (e.g., high-density arrays) but additionally or alternatively arrays of any suitable density. The arrays can include multidimensional arrays (e.g., planar array, 3-dimensional volumetric array, array defined substantially along one or more device surfaces, etc.), single-dimensional arrays (e.g., linear array, curvilinear array, etc.), and/or any other suitable arrays. For example, the device can include a two-dimensional array (e.g., defined substantially on a plane, defined on a curved and/or bent surface, etc.). The arrays can be configured as one or more of: a circular array, an ellipsoidal array, a polygonal array (e.g., a triangular array, rectangular array, a pentagonal array, a hexagonal array, etc.), a circumscribing array, an amorphous array, an array substantially spanning the support structure with which the array is integrated, and any other suitable array type. Additionally or alternatively, the device can include an irregular distribution of tactile interface devices (e.g., arranged substantially on a surface and/or within a volume of the device) and/or any other suitable arrangement of tactile interface devices. Furthermore, the spatial distribu-

tion (e.g., array) can be configured across different layers of the overarching device coupled to the user.

**[0026]** In a first embodiment, as shown in FIG. 3A, the array of tactile interface devices is integrated with a wrist-region wearable band device, wherein the array is distributed circumferentially about the band surface and coupled to electronics that facilitate provision of haptic stimuli. In this embodiment, the system comprises a housing operable to 1) contain electronics for powering the tactile interface devices and transitioning the tactile interface devices between different modes and 2) support the array of tactile interface devices while positioning the array of tactile interface devices in a manner such that the user can sense stimuli provided by the array. The housing can thus be coupled to or otherwise include a fastener that couples the system to a user. The fastener and housing can be of unitary construction or otherwise physically coextensive, or can be otherwise connected, coupled, or coupleable. The fastener is preferably operable to be easily and/or repeatably fastened and unfastened manually by the user, and in specific examples, can include a latch, snap, buckle, clasp, hook-and-loop fastening mechanism, and/or any other suitable fastening mechanism, and/or can be operable to expand and contract (e.g., including an elastic element, such as an expansion band; including a deployment clasp, butterfly clasp, or other clasp that is physically coextensive when unclasped; etc.).

**[0027]** In a second embodiment, the tactile interface devices are configured to be carried with a user (e.g., worn by the user, in proximity to the user). In this embodiment, the tactile interface devices are preferably integrated into a wearable garment, wherein the garment can comprise a top (e.g., shirt, vest, etc.), a bottom (e.g., pants, shorts, skirt, etc.), a headpiece (e.g., headband, earmuffs, hat, etc.), a backpack, an undergarment, socks, and any other suitable form of garment. Additionally or alternatively, the tactile interface devices can be configured to be mechanically coupled to the wearable garment (e.g., retained in one or more pockets of the garment, attached by fasteners such as buttons, clips, magnets, and/or hook-and-loop fasteners, attached by adhesive, etc.). Additionally or alternatively, the tactile interface devices can be configured to attach directly to a user (e.g., by suction, adhesive, etc.), preferably to one or more skin surfaces of the user. Additionally or alternatively, the tactile interface devices can be incorporated into one or more wearable devices (e.g., a head-mounted wearable device, etc.) and/or implanted devices. Additionally or alternatively, the tactile interface devices can be incorporated into prosthetic devices (e.g., lower limb prosthetics, upper limb prosthetics, facial prosthetics, etc.). In an example, such as shown in FIG. 3B, the array of tactile interface devices can be integrated with a vest garment operable to be worn by a user as the user moves about in his/her daily life.

**[0028]** In a third embodiment, such as shown in FIG. 3C, the tactile interface devices are configured to be mechanically coupled to the user by a support device that supports the user (e.g., by a support element of the support device). For example, the tactile interface devices can be integrated into the support element and/or arranged between the user and the support element (e.g., resting on top of the support element). The support devices can include seats, couches, beds, platforms (e.g., for sitting and/or standing on), walls, inclined surfaces (e.g., configured to support a leaning user), and/or any other suitable support devices, as described in

U.S. application Ser. No. 15/661,934 titled “Method and System for Determining and Providing Sensory Experiences” and filed on 27 Jul. 2017, which is herein incorporated in its entirety by this reference.

**[0029]** Additionally or alternatively, the tactile interface devices can be disposed in a device configured to be held by the user (e.g., hand-held, held between an arm and torso of the user, held between the legs of the user, etc.). Additionally or alternatively, the tactile interface devices can be disposed in a device configured to rest on the user (e.g., retained against the user by gravity), such as a blanket. However, the tactile interface devices can additionally or alternatively be coupleable to the user (and/or otherwise configured to interact with the user) in any other suitable manner.

**[0030]** Each tactile interface device (and/or other output unit) is preferably controlled by independent signals and configured to actuate independently from the other output units. Alternatively, a group of output units (e.g., a cluster or subset of the output units) can be independently controlled, such that the group of output units can operate independently from the other output units. Each controlled subset (e.g., individual output unit or cluster) can include one or more output units of the same or different types. In variations, in addition to or in alternative to controlling subsets of actuators (e.g., overlapping and/or disjoint subsets) to convey information as a function of features (e.g. in a first group for a first phoneme; in a second group, including only actuators not included in the first group, for a second phoneme; in a third group, including a subset of actuators of the first and second groups, for a third phoneme; etc.), subsets can be used to map a numerical input to a multi-actuator output. In an example, to make the impression of “sweeps” (e.g., turning actuators on and off in quick succession), one could analyze a frame of music and track the strongest/loudest frequency and control the actuators to produce upward/downward “sweeps” as a function of whether the frequency increased or decreased from a previously analyzed frame.

**[0031]** Each controlled subset is preferably individually identified, such that it has a locally unique identifier (e.g., index value), but can alternatively share an identifier with a second controlled subset of the device, or be otherwise identified. Each controlled subset (or the respective identifier) is preferably associated with a known, stored spatial position on the device (controlled subset position). The controlled subset position can include an arcuate position, radial position, position along an axis (e.g., lateral axis, longitudinal axis, etc.), set of coordinates, grid position, position relative to another device component (e.g., sensor, different output unit, etc.), or be any other suitable position. The controlled subset positions can be stored by the device (e.g., on volatile or non-volatile memory), can be encoded (e.g., implicitly, explicitly) via a re-indexing module (e.g., reindexing array), and/or stored (and/or otherwise made available) by any other suitable system. However, indexing and/or storing can additionally or alternatively be implemented in any other suitable manner.

**[0032]** Each controlled subset is preferably wired in parallel relative to other controlled subsets of the device, but can alternatively be wired in series, wired in a combination of in parallel and in series, or be wired in any other suitable manner (or not be wired). The controlled subsets of the device are preferably controlled by the processor, but can additionally or alternatively be controlled by a remote

computing system (e.g., server system), external device (e.g., mobile device, appliance, etc.), and/or any other suitable computing system.

**[0033]** The input signals associated with the method **100** can be derived from sensors of the system **100** (e.g., wherein sensors are included with the same device(s) that provide the stimuli, wherein sensors are distinct from the devices that provide the stimuli, etc.). The input signals can be derived from local sensors (e.g., sensing an environment of the device and/or user), remote sensors (e.g., sensing a separate environment), virtual inputs (e.g., associated with a virtual environment), and/or any other suitable sensors in any other suitable configuration.

**[0034]** The input signals preferably include audio and/or music input signals. For example, the input signals can be derived from microphones (e.g., multiple microphones, which can be used to perform beam forming operations to remove environmental artifacts, as described below), and/or other audio sensors, sources of audio data streams (e.g., analog electrical connectors such as audio line in connectors; digital electrical and/or optical connectors configured to receive audio and/or music information such as HDMI, TOSLINK, MIDI, etc.; generic computer data connectors such as USB, Ethernet, etc.; wireless connections such as those enabled by a wireless communication module of the system; etc.), and/or any other suitable audio inputs. The sensors associated with input signals can additionally or alternatively include sensors associated with other sensory experiences (e.g., visual, tactile, olfactory, taste, etc.), other environmental information (e.g., location, location type, velocity, temperature, humidity, etc.), and/or any other suitable information.

**[0035]** The sensors can additionally or alternatively include one or more: cameras (e.g., CCD, CMOS, multi-spectral, visual range, hyperspectral, stereoscopic, etc.), spatial sensors (e.g., inertial measurement sensors, accelerometer, gyroscope, altimeter, magnetometer, etc.), location sensors (e.g., GPS, GNSS, triangulation, trilateration, etc.), audio sensors (e.g., transducer, microphone, etc.), barometers, light sensors, temperature sensors, current sensor (e.g., Hall effect sensor), air flow meter, voltmeters, touch sensors (e.g., resistive, capacitive, etc.), proximity sensors, force sensors (e.g., strain gauge meter, load cell), vibration sensors, chemical sensors, sonar sensors, and/or any other suitable sensors. However, the system can additionally or alternatively include any other suitable sensors.

**[0036]** The communication modules can include wired communication modules (e.g., configured to communicate by wired data connections, such as Ethernet, USB, power line, etc.) and/or wireless communication modules (e.g., radios). The wireless communication modules preferably support (e.g., enable communication using) one or more wireless communication protocols (e.g., WiFi, Bluetooth, BLE, NFC, RF, IR, Zigbee, Z-wave, etc.). However, the system can additionally or alternatively include any other suitable communication modules.

**[0037]** The power module can include one or more power input elements, power storage elements, and/or any other suitable elements. The power module is preferably an electrical power module with an electrical input (e.g., electrical power connection such as a wired connector or inductive loop) and/or electrical storage element (e.g., battery, super-capacitor, etc.), but can additionally or alternatively include any other suitable power input and/or storage elements. The

power module can include a battery that is preferably electrically coupled (e.g., connected by conductive wires) to the powered system components, wherein the computational module preferably controls power provision (e.g., as described below), but power provision and/or battery management can additionally or alternatively be performed by any other suitable components.

**[0038]** The computational module can include one or more processors (e.g., CPU or other microprocessor, control circuit, relay system, etc.), computer memory modules (e.g., RAM), computer storage modules (e.g., hard disk drive, flash memory, etc.), and/or any other suitable elements. The computational module is preferably configured to control and/or receive information from the outputs, inputs, communication modules, power modules, and/or any other suitable elements of the system. The computational module can be distributed across multiple systems (e.g., remote server, personal computing device, wearable computing device, mobile computing device, etc.) and/or in the cloud, or can alternatively be implemented in a single computing system.

**[0039]** The computational module is preferably configured to control the controlled subsets (e.g., output units such as tactile interface devices, groups of output units, etc.) individually. In a first example, the processor is configured to provide control signals to each controlled subset (e.g., to a control element of each controlled subset, such as an actuator control circuit). Additionally or alternatively, in a second example, the processor is configured to selectively provide power from the power module to each controlled subset (e.g., by regulating the current provided to each output unit) or to selectively command each controlled subset to enter a mode or attain a set point parameter value (e.g., by communicating a command to an integrated controller of each output unit). However, the computational module can additionally or alternatively be configured to control the controlled subsets in any other suitable manner, or can be configured to not control the controlled subsets.

**[0040]** As described earlier, the system can include embodiments, variations, and examples of the device(s) described in U.S. application Ser. No. 14/750,626, titled "Providing Information to a User Through Somatosensory Feedback" and filed on 25 Jun. 2015; however, the system can additionally or alternatively include any other suitable devices and/or device elements.

#### 4. Method.

##### 4.1 Receiving Input Signals

**[0041]** As described in relation to the method **100** above, Block **S110** recites: receiving an input signal from a sensing device associated with a sensory modality of the user. Block **S110** functions to provide signals that can be transformed into stimuli provided to the user in downstream steps of the method **100**. The input signals are preferably received by and/or derived from one or more sensors of the system described in Section 3 above. In particular, the input signals can be received from sensors of the system (e.g., information sampled by the sensors), other sensors (e.g., sensors connected to the system, such as by a communication module), computing systems (e.g., from computer storage, generated by the computing system, etc.), other systems, and/or any other suitable source.

**[0042]** The input signals can capture information associated with one or more sensory modalities (e.g., audio, visual,

tactile, olfactory, etc.), and/or any other information associated with (e.g., indicative and/or representative of, sampled based on, etc.) a sensory experience (e.g., associated with the environment of the system and/or user, a remote environment, a virtual environment, etc.). The input signals can additionally or alternatively capture non-sensory information, such as information associated with an environment (e.g., the environment of the system and/or user, a remote environment, a virtual environment, etc.). For example, the input signals can include location, speed, acceleration, orientation (e.g., relative to a reference orientation and/or position of the user or of a vehicle occupied by the user), electric and/or magnetic field information (e.g., intensity, orientation, gradient, curl, etc.), navigation information (e.g., turn-by-turn directions), and/or any other suitable information, as described in relation to system components in Section 3 above.

**[0043]** The input signals received in Block S110 preferably include current information (e.g., sent in near real-time to the system, such as streamed substantially concurrent with sampling). The input signals received in Block S110 can additionally or alternatively include advance information (e.g., associated with a sensory experience that is expected to occur at a later time, preferably along with the expected time of occurrence), but can additionally or alternatively include historical information and/or information associated with any other suitable time (or no time).

**[0044]** In a first variation, Block S110 includes receiving audio signals (e.g., from a microphone system, etc.) derived from entities and/or objects in a user's environment. In this variation, receiving the audio signals can include sampling audio information at one or more microphones of a system coupled to a user or otherwise in proximity to the user, as described in relation to the system embodiments of Section 3 above, wherein using signals from multiple microphones can be used in downstream processing steps, as described in more detail below.

**[0045]** In one example, the audio signals can capture human communication content, from which speech components (e.g., phonetic components, etc.) can be processed, extracted, and transformed to provide stimuli to users, wherein the stimuli convey speech information (or information derived from the communication information) to the user through another sensory modality. In another example, the audio signals can capture communication from a non-human entity, from which communication components can be processed, extracted, and transformed to provide stimuli to users, wherein the stimuli convey information to the user through another sensory modality. In another example, the audio signals can capture non-communication information associated with entities or objects in the user's environment, from which features can be processed, extracted, and transformed to provide stimuli to users, wherein the stimuli convey information (or information derived from the speech information) to the user through another sensory modality. In a first specific example of this example, information can be associated with environmental hazards (e.g., alarms, vehicle traffic, hazardous conditions in proximity to the user), notifications (e.g., door bell rings, appliance notifications, etc.), and/or any other suitable environmental information that would be useful for the user to perceive. In a second specific example, information can be associated with a continuous representation of environmental sounds (e.g., associated with features such as sound frequency).

**[0046]** In variations, the audio signals can be received as an analog electrical representation (e.g., line in), a digital audio encoding (e.g., computer audio stream or file), and/or in any other suitable form. In these variations, Block S110 can additionally or alternatively include receiving timing information associated with the audio information (e.g., synchronization information, such as the time at which a portion of the audio information was captured from the user's environment, etc.). However, Block S110 can additionally or alternatively receiving any other suitable input information (e.g., olfactory stimuli, taste stimuli, etc.) in any other suitable manner.

#### 4.2 Preprocessing Input Signals

**[0047]** Block S120 recites: generating a preprocessed signal upon preprocessing the input signal with a set of preprocessing operations, which functions to process out undesired signal components from the input signal(s) received in Block S110, and/or perform suitable preprocessing operations on the input signal to facilitate downstream portions of the method 100. Block S120 is preferably implemented at the processing system components in communication or otherwise coupled to the sensor components associated with Block S110 and described in Section 3 above; however, Block S120 can additionally or alternatively be implemented using any other suitable system components.

**[0048]** In variations, Block S120 can include filtering out signal components associated with noise and/or attenuating or accentuating signal components in any other suitable manner, in order to increase the signal-to-noise ratio (SNR) of input signal(s). Block S120 can additionally or alternatively include filtering or otherwise processing frequency components of the input signal associated with or not associated with specific hearing conditions of the user(s) involved. For example, Block S120 can include isolating high frequency components of the input signals for downstream processing according to subsequent blocks of the method 100, in order to provide information to users with high frequency hearing loss. In more detail, isolation of frequency components (or ranges of frequency components) can implement an algorithm that prioritizes features of interest to the user (e.g., based on sensory condition details, based on typical user preferences, based on personal interest, based on professional interests, etc.) or de-prioritizes features not of interest to the user (e.g., for a user lacking auditory sensitivity in a high-frequency band but having strong auditory sensitivity in a low-frequency band, including many or all features associated with inputs in the high-frequency band but excluding some or all features associated with inputs in the low-frequency band). However, Block S120 can include post-processing or pre-processing the feature sets in any other suitable manner.

**[0049]** In variations, Block S120 can implement one or more digital techniques including one or more of: of an infinite impulse response (IIR) filter, a finite impulse response (FIR) filter, high pass filter, a low pass filter, and a band pass filter operation for processing signals in frequency ranges of interest.

**[0050]** In relation to noise reduction, Block S120 can implement one or more noise reduction techniques including gating operations, expanding operations, operations to remove hum (e.g., in relation to a mains supply frequency and associated harmonics), operations to remove buzz (e.g., in relation to a mains power supply, waveform distortion,



and/or harmonics), operations to remove clicks, operations to remove crackle, operations to remove hiss, and any other suitable noise reduction operation. In specific examples, Block S120 can implement one or more of a spectral subtraction operation and an adaptive filtering operation to de-noise incoming audio signals. In variations related to other types of input signals, Block S120 can use any other suitable filtering technique/noise reduction technique. As shown in example FIGS. 4A and 4B, digitizing can occur upstream or downstream of filtering/attenuation operations.

**[0051]** Block S120 can additionally or alternatively implement a separation operation to separate the input signal(s) into multiple signals associated with different sources. In the context of signals capturing speech content, separation can include separating the input signals into a first portion that includes the speech content and at least one other portion that includes non-speech content. In the context of signals capturing environmental audio, separation can include separating the input signals into a first portion associated with a first classification of objects (e.g., transportation-related objects), a second portion associated with a second classification of objects (e.g., pets/animals), a third portion associated with a third classification of objects (e.g., appliances), a fourth portion associated with a fourth classification of objects (e.g., people), and any other suitable number of portions. In the context of signals capturing musical content, separation can include separating the input signals into a first portion that includes a first portion (e.g., associated with melody), a second portion (e.g., associated with vocals), a third portion (e.g., associated with instrumentals), and any other suitable number of portions. Separation operation(s) associated with any other suitable type(s) of input signal can, however, be implemented in any other suitable manner.

**[0052]** Additionally or alternatively, in some variations, Block S120 can implement a beam forming operation with multiple microphones to remove artifacts (e.g., reverberation artifacts) introduced by the environment of the user. The beam forming operation can include a spatial filtering technique that produces constructive interference of desired signal components (e.g., associated with speech) from multiple sensors (e.g., microphones, accelerometers, etc.), and/or produces destructive interference of undesired signal components (e.g., from the ambient environment) from multiple sensors (e.g., microphones, accelerometers, etc.). The beam forming operation can include control of phasing, relative amplitude and/or any other suitable aspect of signals from each sensor/microphone involved in the process. The beam forming operation can be conventional (e.g., using fixed or switched beam beamformers), or can be adaptive (e.g., using a phased array). The beam forming operation can, however, be implemented in any other suitable manner using any suitable number or type(s) of sensors/microphones.

**[0053]** Block S120 can include implementing a framing operation with the input signal, in order to facilitate latency aspects, and in order to facilitate processing of incoming input signals in an efficient and highly accurate manner. The framing operation can implement any suitable segmenting process to generate frames from the input signal, such that the frames can be processed to identify features of interest (i.e., from Block S130), and/or frames can be labeled accordingly in relation to machine learning operations that improve efficiency of input signal processing for specific user demographics. The frames can be uniformly sized, or

can alternatively be non-uniformly sized. Furthermore, the frames can overlap, or alternatively, the frames may not overlap. The method 100 can include performing elements of the method for each frame (e.g., repeating Blocks S120-S160 for each frame), and/or performing analysis on the frames in any other suitable manner. In an example, the framing operation processing a 16 kHz input audio signal into 20 ms frames, wherein the 20 ms frames have 10 ms of overlap with adjacent frames. However, in variations the input signal can have any suitable characteristic frequency, the frames can have any other suitable size, and the frames can have any other suitable overlapped or non-overlapped configuration.

**[0054]** Pre-processing in Block S120 can additionally or alternatively include implementing a windowing operation (e.g., subsequent to the framing operation). In specific examples, the windowing processing includes an AKB window, Hann window, and/or Kaiser window. Pre-processing in Block S120 can additionally or alternatively implement one or more of: zero-padding, normalization, subsampling within frames or across frames, and any other suitable pre-processing step, in order to facilitate downstream processing steps according to the method 100.

**[0055]** Pre-processing in Block S120 can additionally or alternatively include implementing transformation operations (e.g., functional decomposition operations such as Fourier-related transforms) with outputs of the framing operation and/or any other pre-processing steps including one or more of: a discrete cosine transform (DCT) operation, a modified discrete cosine transform (MDCT) operation, a fast Fourier transform (FFT) operation, a discrete Fourier transform (DFT) operation, a Mel-frequency Cepstral Coefficient (MFCC) operation, a wavelet transformation operation, a McAulay-Quatieri (MQ) operation, a Fast Hartley transformation operation, and any other suitable transformation operation, from which features of interest can be extracted in Block S130.

#### 4.3 Extracting Features

**[0056]** Block S130 recites: extracting a set of features from the preprocessed signal, which functions to generate features that can be processed with one or more neural networks, as described in relation to Block S140 below. In more detail, Block S130 extracts features from the pre-processing operation, which can be provided to the neural network systems of Block S140 to generate or facilitate recognition of components of the input signal efficiently, for eventual encoding and mapping to haptic device outputs. Block S130 is preferably implemented at the processing system components in communication or otherwise coupled to the sensor components associated with Block S130 and described in Section 3 above; however, Block S130 can additionally or alternatively be implemented using any other suitable system components.

**[0057]** In the context of speech information from the input signals, Blocks S130 and S140 can facilitate generation of features that can ultimately be used for transmitting of information from speech components (e.g., phonemes, words, senones, subphones, diphones, triphones, quin-phones, utterances, fillers, etc.) to be perceived by the user in Blocks S150 and S160. However, in the context of other input signals, Blocks S130 and S140 can be used for recognition of any other suitable information components

from the input signal(s), for transmission of appropriate stimuli to users in Blocks S150 and S160.

**[0058]** Block S130 can include generating derived features associated with spectral/frequency from the pre-processing operations of Block S120. In variations, derived features can be associated with one or more of: spectral peak frequencies, spectral peak magnitudes, spectral rolloff aspects, spectral centroid aspects, frequency band features (e.g., energy ratios), gammatone transforms (e.g., gammatone transform coefficients), measures of spectral deformation, measures of spectral width, zero-crossing features (e.g., zero crossing rates), moments of crossing intervals, lacunarity, whitening transforms (e.g., elements of a whitening transform matrix), and any other suitable derived features.

**[0059]** Additionally or alternatively, Block S130 can include generation of any other suitable features, in relation to any other suitable input signal type and/or encoded information associated with the stimuli provided in subsequent blocks of the method 100.

#### 4.4 Neural Network Operations

**[0060]** Block S140 recites: processing the set of features with a neural network system, which functions to transform the features of Block S130 into outputs that can be encoded and mapped in a manner for provision of stimuli to the user as described in relation to Blocks S150 and S160 below. Block S140 is preferably implemented at the processing system components described in Section 3 above. In one variation, Block S140 is implemented on embedded hardware (e.g., of the stimulation device, of the device(s) including sensors for generation of the input signals) and/or on a mobile computing device in communication with one or more devices of the system, etc.), with or without supplemental digital signal processors (DSPs). However, Block S130 can additionally or alternatively be implemented using any other suitable system components.

**[0061]** In Block S140, the neural network operations can include one or more of fully connected configurations, convolutional configurations, and recurrent configurations, combinations of multiple network configurations (e.g., as in batch normalization designs, skip connection designs, multiplicative integration designs, etc.). In variations, the neural network(s) implemented in Block S140 can include one or more of: recurrent neural networks (e.g., with a long short term memory design, with a gated recurrent unit design, etc.), convolutional neural networks, feedforward neural networks, deep neural networks, and any other suitable variant of neural network. Additionally or alternatively, the neural network(s) implemented in Block S140 can be supplemented with statistical models for reinforcement learning and pattern recognition of components of interest from input signals, wherein such statistical models can include one or more of: Hidden Markov Models (HMMs), language models, mixture models (e.g., Gaussian mixture models), and/or any other suitable statistical models. In a specific example, represented in FIG. 5, Block S140 can implement a convolutional neural network with a convolutional layer having a dimension smaller than that of the input layer, with a recurrent neural network design and a Hidden Markov Model for learning/recognizing speech components (e.g., phonemes, words, senones, subphones, diphones, triphones, quinphones, utterances, fillers, etc.). In relation to the specific example, the neural networks can include additional inputs to any layer and/or modification of weights associated

with network components. Additionally or alternatively, the neural networks can incorporate one or more of: a network selector, a volume normalizer, information from additional sensors for learning of a noise environment associated with the input signal(s), and any other suitable data/information.

**[0062]** In relation to additional sensors to enhance learning and recognition facilitated by the neural network(s), Block S140 can leverage data from one or more of: motion sensors (e.g., accelerometers, gyroscopes) for beam forming to learn signal source aspects; location identifying modules (e.g., GPS systems) for selecting networks (e.g., localizing to accents associated with speech content); barometers for learning environmental pressure characteristics; light sensors for learning environmental light characteristics; temperature sensors for learning environmental temperature characteristics; air flow meters; current sensors (e.g., Hall effect sensor); voltmeters, touch sensors (e.g., resistive, capacitive, etc.); proximity sensors; force sensors (e.g., strain gauge meter, load cell); vibration sensors; chemical sensors; and/or any other suitable sensors for learning signal components associated with speech content, the environment of the user, and/or any other suitable input signal components.

**[0063]** The neural networks can be trained (e.g., with dropout and stochastic gradient-based techniques), and in specific examples, can be trained using one or more of: Adam-based methods for efficient stochastic optimization with first-order gradients; second order quasi-Newton methods (e.g., Broyden-Fletcher-Goldfarb-Shanno methods, limited-memory Broyden-Fletcher-Goldfarb-Shanno methods, KFAC methods, etc.), and other methods (e.g., gradient descent methods, Newton's method, conjugate gradient methods, Levenberg-Marquardt methods, etc.). Training data can be augmented with features of the input signal(s), wherein the features can be derived or learned. In examples, the training data features can include or be derived from one or more of: spectral peak frequencies, spectral peak magnitudes, spectral rolloff aspects, spectral centroid aspects, frequency band features (e.g., energy ratios), gammatone transforms (e.g., gammatone transform coefficients), measures of spectral deformation, measures of spectral width, zero-crossing features (e.g., zero crossing rates), moments of crossing intervals, lacunarity, whitening transforms (e.g., elements of a whitening transform matrix), and any other suitable derived features.

**[0064]** In relation to network configurations, Block S140 can implement compression techniques (e.g., in order to allow such neural networks to run efficiently without supplemental signal processing hardware, as indicated above). Network compression techniques can include one or more of: binarization of weights, trained quantization, Huffman coding, pruning, hashing to exploit redundancy in neural networks to achieve reductions in model sizes, and any other suitable compression technique. In a specific example, compression of the neural network(s) used in Block S140 involves a deep compression process including pruning, trained quantization, and Huffman coding to reduce storage requirements without affecting accuracy. Such a method provides increased efficiency and speed in processing, and increases flexibility in where such neural network(s) can be run.

**[0065]** In variations of the method 100 for speech recognition, encoding, and mapping, outputs of the neural networks can include frames labeled with speech component

features associated with one or more of: phonemes, sub-phoneme components, super-phoneme assemblies, non-phoneme components (e.g., phones, diphones, triphones, diphthongs, triphthongs, etc.), words, sentences, and/or any other suitable language components. In related to derived features, outputs can include frame-wise probabilities of speech components identified, accuracy metrics associated with identified speech components, and any other suitable derived features (which can be processed in a machine learning feedback loop, described in more detail in Section 4.6 below). Outputs of the neural networks can additionally or alternatively include features of environmental objects (e.g., motion characteristics, morphological characteristics, classification aspects, state characteristics), features associated with indications of hazards to the user, features associated with other environmental information (e.g., in relation to changes in environmental states that may not be associated with hazards), and any other suitable features. Outputs of the neural networks can additionally or alternatively include any other suitable neural network outputs.

**[0066]** Block S140 can additionally or alternatively include other feature-output transformation techniques (e.g., not using neural networks)

#### 4.5 Mapping and Producing Haptic Outputs

##### 4.5.1 Encoding and Mapping

**[0067]** Block S150 recites: mapping outputs of the neural network system to a device domain associated with a device including a distribution of haptic actuators in proximity to the user. Block S150 functions to encode and map feature-and/or parameter value-derived outputs of the neural network system of Block S140, for use in controlling outputs of the stimulation devices of Block S160. As such, Block S150 facilitates transformation of speech component outputs (or other outputs associated with input signals) from the neural network system of Block S140, into stimuli that can be delivered to the user at a device in proximity to the user.

**[0068]** In specific examples, Block S150 preferably includes encoding and mapping of speech components (e.g., frequency coefficient amplitudes, peak frequency values, phonemes, words, etc.) to specific device elements and/or stimulation parameters. Mappings between speech components (or other components of interest extracted from the input signals) can be associated with actuators of the array of tactile interface devices in a 1:1 manner or alternatively, not in a 1:1 manner. As such, Block S150 can include encoding and/or mapping complex waveforms associated with stimulation parameters of a subset of the array of tactile stimulation devices (e.g., to create stimulation “textures” associated with different components of interest). However, Block S150 can include encoding and/or mapping in any other suitable manner.

**[0069]** In one variation, Block S150 encodes the outputs of the neural network system of Block S140 to haptic patterns executable using the array of tactile interface devices described in Section 3 above, wherein the haptic patterns are associated with control signals that activate devices of the array of tactile interface devices, and wherein the control signals can be executed in Block S160 of the method 100. In variations of Block S150 associated with speech component-associated outputs, Block S150 can include can encoding each of the outputs with a spatial aspect and/or device stimulation parameter of the array of

tactile interface devices. In relation to speech components described, the Block S150 can transform, encode, or otherwise map outputs derived from the neural network operation (s) of Block S140 to one or more of: subdomains (e.g., subregions, sub-clusters, sublayers, etc.) of the array of tactile interface devices; different stimulus aspects associated with the array of tactile interface devices; and any other suitable aspects of the array of tactile stimulus devices.

**[0070]** Speech components of Block S140 can be mapped to domains (e.g., layers, regions, clusters, areas, etc.) of the array of tactile interface devices, examples of which are shown in FIG. 6. As such, different speech components (e.g., frequency locations of formants, peak frequency amplitudes, phonemes, sub-phoneme components, super-phoneme assemblies, phones, diphones, triphones, diphthongs, triphthongs, etc.) can be mapped to different domains of the array of tactile interface devices. For instance, in a device variation that has a distribution of the array of tactile stimulation devices configured about a wristband wearable: each device of the array of tactile stimulation devices can be associated with a corresponding phoneme or word, such that different devices of the array of tactile stimulation devices can “play” phonemes/words in a specific pattern that corresponds to the message of the communication data of Block S110, and can be detected at a wrist region of the user.

**[0071]** In another example, in a device variation that has a distribution of the array of tactile stimulation devices integrated with a wearable system or other support system in proximity to the user: each device of the array of tactile stimulation devices can be associated with a corresponding phoneme or word, such that different devices of the array of tactile stimulation devices can “play” phonemes/words in a specific pattern that corresponds to the message of the communication data of Block S110, and can be detected at a body region of the user that is receptive to touch sensations. However, variations of the example can associate devices of the array of tactile stimulation devices with speech components in any other suitable manner.

**[0072]** In the examples and variations described above, the domains/regions of the array of tactile stimulation devices can be fixed or dynamically modifiable. For instance, the subdomain can be dynamically modified, according to the encodings performed in Block S150, in order to convey a wider variety of information and/or more complex information to the user with a limited physical device space.

**[0073]** Additionally or alternatively, in variations related to speech components described in Block S140 above, the transformation model can additionally or alternatively transform, encode, or otherwise associate speech component labels with a set with stimulus parameters of the array of tactile interface devices. In variations, the transformation operation can map different speech components (e.g., phoneme pitch, phoneme energy, phoneme tone, phoneme emphasis, etc.) to a range of stimulus types. In variations, stimulus parameters can include one or more of: output type (e.g., intermittent, pulsed, continuous, etc.); pulse pattern; pulse waveform characteristics (e.g., sinusoidal, square wave, triangular wave, wavelength, etc.), output amplitude, output intensity; output duration; out pulse duration, etc.), device domains involved in an output (e.g., a sweeping pattern using multiple devices), and any other suitable stimulus parameter. For instance, in a device variation that has a distribution of the array of tactile stimulation devices configured about a wristband wearable: each device of the

array of tactile stimulation devices can output a specific stimulus parameter corresponding to a speech component, such that the devices can relay information not only to speech component labels (e.g., phoneme labels), but also more complex language aspects (e.g., phoneme pitch, phoneme energy, phoneme tones, phoneme emphasis, etc.) and can be detected at a wrist region of the user.

**[0074]** In another example, in a device variation that has a distribution of the array of tactile stimulation devices integrated with a wearable device or other support structure in proximity to the user: each device of the array of tactile stimulation devices can output a specific stimulus parameter corresponding to a speech component, such that the devices can relay information not only to speech component labels (e.g., phoneme labels), but also more complex language aspects (e.g., phoneme tones, phoneme emphasis, etc.) and can be detected at a body region of the user that is sensitive to touch. However, variations of the example can associate stimulus parameters of the array of tactile stimulation devices with speech components in any other suitable manner.

**[0075]** In variations related to speech components described in Block S140 above, the transformation model can additionally or alternatively transform, encode, or otherwise associate speech component outputs of Block S140 with complex or combined outputs of the array of tactile interface devices. In variations, the transformation operation can result in generation of encodings related to both subdomains and stimulus outputs available using the array of tactile stimulus devices.

**[0076]** In a first example, associated with the English language, the Block S150 can assign each of 47 phoneme labels to specific devices of the array of tactile interface devices (e.g., distributed about a wristband, distributed across a vest, etc.), such that the array of tactile interface devices can “play back” stimuli associated with different phonemes in an order corresponding to portions of the input signal(s) received in Block S110. As shown in FIG. 6, a wrist band can include an array of tactile interface devices, each mapped to a specific phoneme, which can be signaled to play the phoneme according to a pattern corresponding to the input signals in Block S110. In a second example, associated with spoken Mandarin, Block S150 can assign each of 41 phoneme labels to specific devices of the array of tactile interface devices and four stimulus patterns (e.g., pulse patterns) associated with four spoken tones, such that the array of tactile interface devices can “play back” stimuli associated with different phonemes and different phoneme tones in an order corresponding to portions of the input signals(s) received in Block S110.

**[0077]** Encoding and mapping in Block S150 can additionally or alternatively be implemented in any other suitable manner, such as described in U.S. application Ser. No. 14/750,626, titled “Providing Information to a User Through Somatosensory Feedback” and filed on 25 Jun. 2015.

#### 4.5.2 Stimulation Delivery

**[0078]** Block S160 recites: at the distribution of haptic actuators, cooperatively producing a haptic output representative of at least a portion of the input signal, thereby providing information to the user. Block S160 preferably includes executing control signals operable to deliver stimulation through the array of tactile interface devices (e.g., the output parameters determined in Block S150, etc.) with the

distribution of tactile interface devices coupled to the user. Block S150 preferably functions to enable outputs to be delivered to the user through the tactile interface devices, according to the transformation and encoding algorithms of Blocks S120-S140. The control signals can be executed according to methods described in U.S. application Ser. No. 14/750,626, titled “Providing Information to a User Through Somatosensory Feedback”; however, the control signals can additionally or alternatively be executed in any other suitable manner. For instance, Block S160 can include receiving one or more user inputs operable to adjust a gain level, equalizer levels, and/or other aspect(s) of output stimuli, such that the user can customize the intensity of stimuli provided through the array of tactile interface devices (e.g., of all stimuli, of stimuli associated with particular input and/or output characteristics, etc.). However, Block S150 can additionally or alternatively include controlling the output devices in any other suitable manner.

**[0079]** In the variations and examples described above, phoneme outputs generated in Block S140 and corresponding to the input signal(s) of Block S110 can be encoded, mapped, and delivered through the array of tactile interface devices in a manner similar to the natural timing of speech. In more detail, stimulus provision in Block S160 preferably occurs in real-time or near real-time (e.g., within a time threshold, such as 100 ms, 90 ms, 75 ms, 50 ms, 110 ms, 125 ms, 150 ms, 200 ms, 300 ms, etc.), as represented in FIG. 7, such that the user perceives tactile feedback substantially simultaneously with reception of input signals in Block S110 (e.g., is unable to discern a delay between the input signal and tactile feedback), with minimal delay. However, delivery of haptic stimuli can alternatively be implemented in a manner that does not mimic natural speech timing. As such, delivery of haptic stimuli can be implemented with any suitable speed, frequency, cadence, pauses (e.g., associated with grammatical components of language), gain (e.g., amplitude of stimulation corresponding to “loudness” or punctuation), pattern (e.g., spatiotemporal pattern played using subarrays of the array of tactile interface devices, etc.), and any other suitable output component.

**[0080]** Stimulus provision in Block S160 can, however, be implemented in any other suitable manner.

#### 4.6 Additional Machine Learning Aspects and Feedback Loops

**[0081]** While neural network operations are described in Block S140 above, the method 100 can additionally or alternatively implement additional or alternative machine learning techniques for optimization of extraction of speech components or other components from the input signal. The method 100 can additionally or alternatively implement machine learning algorithms having an objective function for optimizing power consumption of the array of tactile interface devices, thereby improving overall device and/or system performance. Additionally or alternatively, machine learning algorithms can be used to optimize stimulation outputs for user specific conditions (e.g., user specific hearing loss), as described in more detail below.

**[0082]** In one variation, given N tactile interface devices in the array of tactile interface devices and one-hot encoding of speech components (e.g., phonemes), N+1 classes can be represented (including a “BLANK” state associated with inactivation of all of the array of tactile interface devices). In this variation, an objective function for machine learning

operations can include an optimization of frame-wise cross-entropy among predicted speech component (e.g., phoneme) annotations; however, optimization of parameters in relation to frames of input signal data can be implemented in any other suitable manner. Additionally or alternatively, an objective can include generation of accuracy metrics associated with speech component annotations, wherein in a specific example, log-likelihoods of a phoneme-level accuracy metric can be determined across a set of contiguous frames of input signal data, based on an analysis of correct annotation for each of the set of frames. In more detail, an entire phoneme captured in the set of frames of input signal data can exhibit a high level of the accuracy metric if every frame is predicted to be either correct or BLANK, and at least one frame is predicted to be correct. Such accuracy analyses can then be used to refine annotations of phoneme (or other speech components) in processing subsequent frames of input signal data. Additionally or alternatively, accuracy analyses can be used to modulate stimulus output aspects of the array of tactile interface devices, in relation to Block S160 above. For instance, tactile feedback can be scaled to represent accuracy level in relation to predictions of phonemes in frames of the input signal data. Generation and usage of accuracy metrics for optimization can, however, be implemented in any other suitable manner.

**[0083]** Additionally or alternatively, in some variations, an objective function for machine learning operations can include optimization of power consumption characteristics associated with the array of tactile interface devices (e.g., in relation to specific device constraints). In one specific example, such operations can optimize for sparsity of an encoded representation of a speech component output with respect to delivery of stimulus parameters upon activation of one or more of the array of tactile interface devices, wherein sparsity is optimized to reduce power consumption while still allowing perception of the stimuli by the user. In relation to these variations, a dynamic range of motor activations can be learned from psychophysical analyses of users (e.g., with respect to minimum sensory perception thresholds, with respect to maximum sensory perception thresholds); however, optimizing power consumption under a constraint of stimuli perception by a user can be implemented in any other suitable manner.

**[0084]** Additionally or alternatively, in relation to optimization of stimulation delivery for specific user or environment conditions (e.g., in relation to user specific hearing loss, in relation to noise environments, in relation to accents, etc.), an example method flow is shown in FIG. 8A. In more detail, an original system state (e.g., in “factory” conditions) including a device (e.g., wristband) and original-state neural network-based algorithm can be provided to a user. Then, as the user uses the device according to his/her conditions and in his/her environment, data gathered during use can be transmitted back to remote processors (e.g., of a cloud-based system for aggregating and processing user data) to update the original-state neural network-based algorithm to a user data-informed algorithm. The device (e.g., wristband) can then be updated (e.g., in a remote firmware update operation) and the cycle can continue as the user continues to use the device. In an alternative configuration, as shown in FIG. 8B, the remote processors can be omitted, and the device can be configured to update itself as the user uses the device.

**[0085]** Furthermore, the method 100 can include any other suitable Blocks operable to promote provision of informa-

tion to users, through the array of tactile interface devices, in any other suitable manner. For instance, the method 100 can include rapidly training a user to learn to correctly identify speech components associated with haptic outputs provided through an array of tactile interface devices, by providing pre-recorded speech with time-locked haptic outputs through the array of tactile interface devices. In a specific application, such a training protocol can allow users with high-frequency hearing loss to help them discriminate between commonly confused higher frequency phonemes (e.g., /th/, /f/, /s/, /h/, /k/, /z/, /b/, /dh/, /t/, /d/, /v/, etc.), in coordination with stimulus provision through the array of tactile interface devices. For example, the method can include providing haptic outputs (e.g., at the array of tactile interface devices) representing phoneme components related to phonemes that users with high-frequency hearing loss (e.g., presbycusis) may have difficulty perceiving. In a first specific example, the method includes providing haptic outputs representing phoneme components (e.g., high frequency phoneme components) including /f/, /z/, /b/, /th/, /dh/, /t/, /d/, /s/, and/or /v/. In a second specific example, the method includes providing haptic outputs representing phoneme components (e.g., high frequency phoneme components) including /th/, /f/, /s/, /h/, and/or /k/. However, variations of the training protocol can alternatively be implemented for users with other impairments, users with no impairments, and/or for any other suitable phonemes or speech components.

**[0086]** While some variations of machine learning techniques are described above, in relation to steps of the method 100 above, the method 100 can additionally or alternatively utilize any other suitable machine learning algorithms. In variations, the machine learning algorithm(s) can be characterized by a learning style including any one or more of: supervised learning (e.g., using logistic regression, using back propagation neural networks), unsupervised learning (e.g., using an Apriori algorithm, using K-means clustering), semi-supervised learning, reinforcement learning (e.g., using a Q-learning algorithm, using temporal difference learning), and any other suitable learning style. Furthermore, the machine learning algorithm can implement any one or more of: a regression algorithm (e.g., ordinary least squares, logistic regression, stepwise regression, multivariate adaptive regression splines, locally estimated scatterplot smoothing, etc.), an instance-based method (e.g., k-nearest neighbor, learning vector quantization, self-organizing map, etc.), a regularization method (e.g., ridge regression, least absolute shrinkage and selection operator, elastic net, etc.), a decision tree learning method (e.g., classification and regression tree, iterative dichotomiser 3, C4.5, chi-squared automatic interaction detection, decision stump, random forest, multivariate adaptive regression splines, gradient boosting machines, etc.), a Bayesian method (e.g., naïve Bayes, averaged one-dependence estimators, Bayesian belief network, etc.), a kernel method (e.g., a support vector machine, a radial basis function, a linear discriminate analysis, etc.), a clustering method (e.g., k-means clustering, expectation maximization, etc.), an associated rule learning algorithm (e.g., an Apriori algorithm, an Eclat algorithm, etc.), an artificial neural network model (e.g., a Perceptron method, a back-propagation method, a Hopfield network method, a self-organizing map method, a learning vector quantization method, etc.), a deep learning algorithm (e.g., a restricted Boltzmann machine, a deep belief network method, a convolution

network method, a stacked auto-encoder method, etc.), a dimensionality reduction method (e.g., principal component analysis, partial least squares regression, Sammon mapping, multidimensional scaling, projection pursuit, etc.), an ensemble method (e.g., boosting, bootstrapped aggregation, AdaBoost, stacked generalization, gradient boosting machine method, random forest method, etc.), and any suitable form of machine learning algorithm.

#### 4.7 Alternative Feature-Output Transformations.

**[0087]** The method **100** can additionally or alternatively include other feature-output transformation techniques (e.g., not using neural networks, supplementing the neural network techniques described above regarding Block **S140**, etc.). In one such variation, the method includes transforming frequency-related features of the input signal into output signals.

**[0088]** A first example of this variation includes: segmenting the input signal into windows (e.g., as described above regarding Block **S120**) and, for each window: performing a transformation (e.g., DCT and/or other functional decomposition, such as described above regarding Block **S120**); optionally discarding extraneous portions of the transformation output (e.g., for a user with high-frequency hearing loss, discarding DCT coefficients corresponding to lower-frequency input signal features, such as frequencies easily perceived by the user); and mapping the remaining (e.g., not discarded) transformation outputs (e.g., DCT coefficient magnitudes) to actuator outputs (e.g., as described above regarding Block **S150**). This example can optionally include performing a dimensionality reduction procedure. For example, if the number of remaining DCT coefficients is greater than the number of haptic output devices, the coefficients can be binned (e.g., into larger frequency bands), and the average magnitude of the coefficients in each bin can be mapped to actuator outputs. The binning can be uniform or non-uniform (e.g., pre-determined and/or dynamically-determined binning; determined based on individual user needs, typical preferences, etc.).

**[0089]** A second example of this variation includes: optionally performing a time-domain filtering of the input signal to discard extraneous frequencies (e.g., for a user with high-frequency hearing loss, using a high-pass filter to discard lower-frequencies of the input signal, such as frequencies easily perceived by the user); extracting a plurality of frequency bands (e.g., one for each actuator or controlled subset) from the filtered input signal (e.g., performing parallel bandpass filtering processes on the signal); determining a metric associated with each band (e.g., the envelope of each band); and mapping this metric to actuator outputs. The frequency bands extracted for the actuators are preferably contiguous frequency ranges, but can additionally or alternatively include any suitable frequency bands.

**[0090]** However, the method **100** can additionally or alternatively include any other suitable feature-output transformation techniques.

#### 4.8 Repetition.

**[0091]** The method **100** can optionally include repeating any or all of the method blocks (e.g., Blocks **S110-S160**). In one example, the method **100** include receiving a continuous stream of input information (e.g., real-time audio signals, such as sampled at the microphone systems of the device) in

Block **S110**, and continuously performing Blocks **S120-S160** based on the input information (e.g., based on a recently-received subset of the information). In a second example, the method **100** includes receiving bulk input information in Block **S110**, segmenting the input information into consecutive time windows, and performing Blocks **S120-S160** for each time window. However, the method **100** and/or any of its elements can be repeated in any other suitable manner.

**[0092]** Although omitted for conciseness, the preferred embodiments include every combination and permutation of the various system components and the various method processes. Furthermore, various processes of the preferred method can be embodied and/or implemented at least in part as a machine configured to receive a computer-readable medium storing computer-readable instructions. The instructions are preferably executed by computer-executable components preferably integrated with the system. The computer-readable medium can be stored on any suitable computer readable media such as RAMs, ROMs, flash memory, EEPROMs, optical devices (CD or DVD), hard drives, floppy drives, or any suitable device. The computer-executable component is preferably a general or application specific processing subsystem, but any suitable dedicated hardware device or hardware/firmware combination device can additionally or alternatively execute the instructions.

**[0093]** The FIGURES illustrate the architecture, functionality and operation of possible implementations of systems, methods and computer program products according to preferred embodiments, example configurations, and variations thereof. In this regard, each block in the flowchart or block diagrams may represent a module, segment, step, or portion of code, which comprises one or more executable instructions for implementing the specified logical function(s). It should also be noted that, in some alternative implementations, the functions noted in the block can occur out of the order noted in the FIGURES. For example, two blocks shown in succession may, in fact, be executed substantially concurrently, or the blocks may sometimes be executed in the reverse order, depending upon the functionality involved. It will also be noted that each block of the block diagrams and/or flowchart illustration, and combinations of blocks in the block diagrams and/or flowchart illustration, can be implemented by special purpose hardware-based systems that perform the specified functions or acts, or combinations of special purpose hardware and computer instructions.

**[0094]** As a person skilled in the art will recognize from the previous detailed description and from the figures and claims, modifications and changes can be made to the preferred embodiments of the invention without departing from the scope of this invention defined in the following claims.

We claim:

1. A method for providing information to a user, the method comprising:
  - receiving an input signal from an audio sensing device, the input signal associated with speech;
  - generating a preprocessed signal upon preprocessing the input signal with a set of preprocessing operations for a set of frames of the input signal;
  - extracting a set of features from the preprocessed signal;
  - processing the set of features with a neural network system;

mapping outputs of the neural network system to a device domain associated with a limb-worn device including a distribution of haptic actuators in proximity to the user; and

in near real time and at the distribution of haptic actuators, cooperatively producing a haptic output representative of speech components of the input signal, thereby providing speech information to the user at the limb-worn device.

2. The method of claim 1, wherein receiving the input signal is performed contemporaneously with producing the haptic output and delivering stimulation to the user with the haptic output, thereby allowing the user to perceive speech content captured through the input signal, through touch receptors, in near real time.

3. The method of claim 2, wherein producing a haptic output representative of speech components comprises mapping outputs of the neural network system, associated with high frequency speech components, to the device domain and delivering tactile stimuli representative of the high frequency speech components to the user.

4. The method of claim 1, wherein speech components represented in the haptic output comprise at least one of: phoneme-associated components and word-associated components.

5. The method of claim 1, wherein mapping between the distribution of haptic actuators and speech components comprises a 1:1 mapping.

6. The method of claim 1, wherein receiving the input signal comprises receiving input signals from a set of microphones of the audio sensing device, and implementing a noise removal operation that applies a beam forming operation to the input signals from the set of microphones.

7. The method of claim 1, further comprising refining at least one of: generating the preprocessed signal, extracting the set of features from the preprocessed signal, processing the set of features with the neural network system, mapping outputs of the neural network system to the device domain, and producing the haptic output with a training dataset, wherein refining comprises improving power consumption of the distribution of haptic actuators with a user tactile perception constraint.

8. A method for providing information to a user, the method comprising:

receiving an input signal from a sensing device associated with a sensory modality of the user;

generating a preprocessed signal upon preprocessing the input signal with a set of preprocessing operations;

extracting a set of features from the preprocessed signal; processing the set of features with a neural network system;

mapping outputs of the neural network system to a device domain associated with a device including a distribution of haptic actuators in proximity to the user; and at the distribution of haptic actuators, cooperatively producing a haptic output representative of at least a portion of the input signal, thereby providing information to the user.

9. The method of claim 8, wherein the sensing device comprises at least two of: a microphone system, a motion sensing unit, and a location identifying unit, and wherein the sensing device is coupled to the device including the distribution of haptic actuators.

10. The method of claim 9, wherein the device including the distribution of haptic actuators comprises at least one of a wrist-borne form factor and a torso-coupled form factor in retaining the distribution of haptic actuators proximal a body region of the user during use.

11. The method of claim 8, wherein generating the preprocessed signal comprises segmenting the input signal into a set of frames, and wherein the method further comprises a) generating an accuracy metric for a predicted speech component captured in at least a subset of the set of frames, and b) modulating a stimulation parameter of the haptic output based upon the accuracy metric.

12. The method of claim 8, wherein the neural network system comprises a convolutional layer and a recurrent architecture, and implements a statistical model for recognition of speech components from the input signal.

13. The method of claim 8, wherein processing the set of features with the neural network system comprises compressing the neural network system with a network compression technique comprising a deep compression process including pruning, trained quantization, and Huffman coding, and executing the compressed neural network system on at least one of embedded hardware of the device and a mobile device of the user.

14. The method of claim 8, wherein the set of features comprises spectral peak frequencies, spectral peak magnitudes, spectral rolloff-derived features, spectral centroid-derived features, frequency band energy ratios, gammatone transforms, measures of spectral deformation, zero crossing rates, lacunarity, and moments of crossing intervals.

15. The method of claim 8, wherein the set of preprocessing operations comprises a filtering operation to remove environmental artifacts, a windowing operation, and a transformation operation implemented across a set of frames of the input signal.

16. The method of claim 8, wherein the haptic output represents a high frequency phoneme components comprising at least one of: a /t/ phoneme, a /z/ phoneme, a /b/phoneme, a /th/ phoneme, a /dh/ phoneme, a /t/ phoneme, a /d/ phoneme, a /s/phoneme, and a /v/ phoneme.

17. The method of claim 8, wherein receiving the input signal comprises receiving input signals from a set of microphones of the sensing device a motion sensing unit of the sensing device; and wherein the method further includes implementing a noise removal operation that applies a beam forming operation to the input signals based on data from the set of microphones and the sensing device.

18. The method of claim 8, wherein producing the haptic output comprises delivering a unique waveform using a subset of the distribution of haptic actuators, for each of a set of phonemes captured in the input signal.

19. The method of claim 8, wherein receiving the input signal is performed contemporaneously with producing the haptic output and delivering stimulation to the user with the haptic output, thereby allowing the user to perceive content captured through the input signal, through touch receptors, in near real time.

20. The method of claim 8, wherein the distribution of haptic actuators is arranged circumferentially about a band support structure operable to be positioned at a wrist region of the user during operation of the method.