RiSH: A Robot-integrated Smart Home for Elderly Care

Ha Manh Do[a], Minh Pham[a], Weihua Sheng[a,*], Dan Yang[b], Meiqin Liu[c]

[a]*School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK, 74078, USA*
[b]*College of Information Science and Engineering, Northeastern University, Liaoning 110004, China*
[c]*College of Electrical Engineering, Zhejiang University, Hangzhou 310027, China*

Abstract

This article presents the development of a robot-integrated smart home (RiSH) which can be used for research in assistive technologies for elderly care. The RiSH integrates a home service robot, a home sensor network, a body sensor network, a mobile device, cloud servers, and remote caregivers. A layered architecture is proposed to guide the design and implementation of the RiSH software. Basic service functions are developed to allow the RiSH to recognize human body activity using an inertial measurement unit (IMU) and the home service robot to perceive the environment through audio signals. Based on these functions, we developed two low-level applications: 1) particle filter-based human localization and tracking using wearable motion sensors and distributed binary sensors; 2) Dynamic Bayesian Network-based human activity recognition using microphones and distributed binary sensors. Both applications extend the robot's perception beyond its onboard sensors. Utilizing the low-level applications, a high-level application is realized that detects and responds to human falls. We conducted experiments in our RiSH testbed to evaluate auditory perception services, human body activity recognition, human position tracking, sound-based human activity monitoring, and fall detection and rescue. Twelve human subjects were asked to conduct daily activities in the testbed and mimic falls. All data of their movement, body activities, and sound events were collected by the robot. Human trajectories were estimated with a root mean square error of less than 0.2 m. The robot was able to recognize 37 human activities through sound events with an average accuracy of 88% and detect falling sounds with an accuracy of 80% at the frame level. The experiments show the operation of the various components in the RiSH and the capabilities of the home service robot in monitoring and assisting the resident.

Keywords:
Elderly Care, Smart Home, Home Service Robot, Assistive Technology.

1. Introduction

The elderly population around the world is steadily increasing. The number of people 60 years old and older increased to almost 900 million in 2015 and is forecasted to reach 2 billion by 2050 [1]. Older adults are an important asset to society and need to be cared for in age-friendly physical and social environments. Although services such as adult day care, long term care, and nursing homes can provide the elderly with healthcare, nutritional, social, and other daily living support, the feeling of independence is lost. Elders would prefer to stay in the comfort of their home where they feel more confident than moving to any expensive adult care or healthcare facilities. Hence, if older adults are able to complete self-care activities on their own, it will encourage them to maintain independence and provide them with a sense of accomplishment and ability to enjoy independence longer [2]. The best way to support them is to provide a physical environment that promotes active aging through the use of innovative technologies, such as smart homes and assistive robots.

A smart home environment is defined as a ubiquitous computing application that is able to provide users with context-aware assistive services and home automation. Moreover, smart homes provide comfort, healthcare, and security services to their inhabitants. On the other hand, mobile robots have come into human environments in recent years. They have been used in many places such as factories, offices, hospitals, and homes. For the elderly who live independently in their own residence, home service robots can act as a tool to serve the human, an avatar to represent the caregiver, and a social companion to collaborate and interact with the elderly. In a robot-integrated smart home (RiSH), the home service robots can utilize the smart home sensor networks just like their own sensors, enabling them to better assist the elderly and collaborate with remote caregivers. Therefore, RiSHs would be the perfect use of technology to achieve the goal of caring for the elderly in their own home.

This paper aims to present the overall ideas of a RiSH and introduce a RiSH testbed to support future research in assistive technologies for elderly care. The rest of this paper is organized as follows. Section 2 reviews related works in home service robots, smart homes, and robot-integrated smart homes. Section 3 presents the overall design of the RiSH for elderly care. Section 4 describes the hardware design of the RiSH testbed. Section 5 explains the software architecture of the RiSH. Section 6 presents the implementation of several key basic services.

*Corresponding author
Email address: weihua.sheng@okstate.edu (Weihua Sheng)

Section 7 details the development of both low-level and high-level applications for the RiSH. Section 8 describes the experiments and gives the results. Section 9 concludes the paper and also discusses the potential future work.

2. Related Works

In recent years there has been much interest in developing robotic technologies for elderly care. Several home service robots or personal robots have been made commercially available, such as Aibo [3], Care-o-Bots [4], and Paro [5]. In academia, researchers have developed many robots for domestic environments, such as Johnny [6], European CompanionAble project's Hector [7], IRT's home-assistant robot [8], and Hobbit robot [9]. Those robots were equipped with various functions such as mapping, navigation, object recognition, speech recognition, speech synthesis, and dialogue management. They can work as autonomous museum tour guides or telepresence robots and remind people about routine activities such as eating, drinking, and taking medicine. However, these robots do not have the capability to connect to a smart home to leverage its sensing capability. In addition, their auditory perception is limited to voice recognition and is not able to recognize various event sounds in a home environment.

Recently, smart homes have been receiving growing interest. Research labs from industry and academia have developed various smart home environments. Smart home platforms that integrate with smart sensor networks, smart appliances, communication modules, as well as ubiquitous computing and displaying devices have been developed by many companies such as Philips, Cisco, GTE, Sun, Ericsson, Samsung, Google, and Microsoft. There have been several research projects on smart homes for elderly care, such as the Aware Home Research Initiative (AHRI) project[10], the Gator Tech Smart House [11], the Adaptive House [12], and the CASAS smart home [13]. Several groups have focused on smart environments that assist individuals with health challenges such as the Gloucester Smart Home [14], the MavHome project [15], the MALITDA smart house for individuals with special needs [16], the smart home for real time activity recognition and wellness determination of elderly [17]. Another solution for intelligent home care is proposed in [18], which combines an intelligent agenda manager and an intelligent monitoring framework to provide the elderly with sensors-based monitoring, activity detection, intelligent assistance in scheduling and decision-making, and fall detection. The AAL4ALL [19], an Ambient Assisted Living (AAL) project, develops a platform that enables caregivers to connect to an ambient assisted living ecosystem. It is clear that such environments are the results of recent advancement in wireless mobile communications and sensor networking technologies, among many others.

Mobile robots have been recently integrated into a few smart homes through wireless communications, but with different roles. In [20], the Sony AIBO robot is used to collect image and audio data of the on-site scene in a tele-medicine system for health and activity monitoring. The Physically Embedded Intelligent Systems (Peis-Ecology) framework [21] treats the devices in the robots and the devices in the environment as interconnected components of the same system. The system functionalities can be allocated to the robots or the device in the environment. The Universal Plug and Play technology was introduced in [22] to integrate different electronic devices and surveillance robots into a smart home. The architecture of Robotics and Cloud-assisted Healthcare System (ROSCHAS) [23] was designed to enable a robot and a smart home to provide pervasive healthcare services and especially mental healthcare for older adults who live alone. The KSERA (Knowledgeable SErvice Robots for Aging) project integrates a social robot into a smart home environment to improve the quality of independent living for elderly people [24][25].

To summarize, smart home environments offer a better quality of life to residents by employing automated appliance control and assistive services. Integration of mobile robots into smart home environments enables many new applications, especially for elderly care. It also helps shift the computation complexity from the mobile robots to the environments. The technologies which are present in smart homes can be used to improve mobile robots in terms of performance and safety while reducing the cost. Although the previous research projects can provide general frameworks to develop the robots and smart home systems, they mainly provided specific solutions in integrating robots into smart homes and developing separate in-home services for elderly care. In addition, these frameworks are hard to be adopted in the RiSH that integrates all components, including a robot, a smart home, a Cloud server, a body sensor network, and a remote caregiver, into a whole system. There lacks a framework that consists of both a complete reference hardware design and a comprehensive software architecture to implement the RiSH. In this paper, we aim to propose a solution to integrating a service robot into smart homes for elderly care. Moreover, an overall software architecture of the RiSH is proposed. This architecture aims to enable modularity, extensibility, customizability, and reusability. The software in each component of the RiSH can be independently developed to provide basic service functions that can be used to develop applications of the RiSH. We implement two examples of low-level applications: human position tracking and human activity monitoring, which allows the development of a high-level application: fall detection and rescue. We develop the human activity monitoring application by combining the wearable sensor-based body activity recognition and the sound event recognition. It can recognize multiple activities in the home environment and detect human falls through sound events. Multiple types of falling sounds are tested with different settings of environmental noise and even when the robot is unable to observe the resident due to occlusion. This application also allows the robot to respond to such situations and connect to a remote caregiver for assistance. These applications show the operation of the various components in the RiSH as a whole. They also demonstrate the capabilities of the home service robot in monitoring and assisting the resident. The RiSH testbed can serve as a reference design for smart homes that involve service robots.

3. Overall Concept of RiSHs for Elderly Care

A smart assistive living environment is expected to provide not only living comfort but also in-home services as human caregivers usually do. Such services include assisting daily activities, providing healthcare, and meeting the need of socialization. Thus, the future RiSHs should provide the elders these services in their own home environments. In this section, we highlight the challenges of elderly care, then propose the overall concept and architecture of RiSHs.

Older adults who live independently in their homes should be able to perform basic activities of daily living (ADLs) such as eating, drinking, bathing, toileting, and sleeping. Moreover, they must be able to carry out instrumental activities of daily living (IADLs) such as managing a medication regimen, doing household chores, and preparing meals of adequate nutrition. Independent living also needs enhanced activities of daily living (EADLs) to adapt to a changing environment, to maintain a positive attitude, and to stay safe and healthy [26]. However, the older they are, the more challenges the elders have to face in independent living. As discussed in [27], these challenges include mobility, communication, emotional health, physical health, and mental health. Mobility is a major problem with the older adults, such as difficulties with stairs or the loss of general physical mobility. Besides, problems of communication may significantly impact older adults' quality of living. For example, the inability to make phone calls, access the Internet, and make appointments online with doctors brings much inconvenience to one's daily life. The problems of emotional health are isolation, the feeling of loneliness, and negative moods. Moreover, when getting older, the older adults face many specific problems of physical health, such as insufficient nutrition and loss of senses (mainly sight, touch, and hearing). Also, the problems with knees, hips, and other joints lead to increase the risk of fall and difficulty in performing ADLs. Furthermore, mental health problems such as Alzheimer's, dementia, or memory loss may cause great difficulties in daily living.

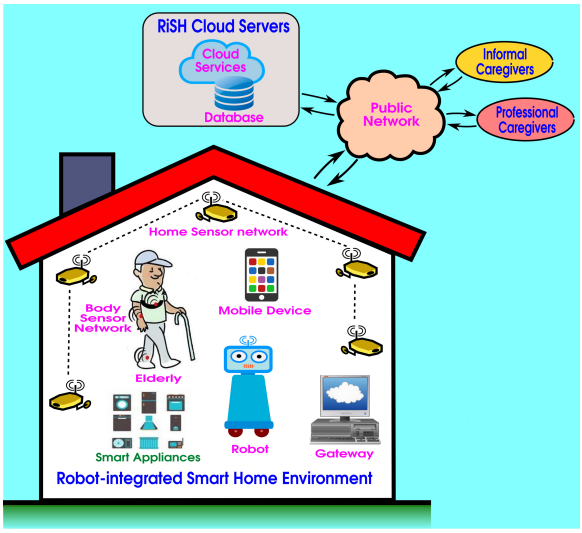


Figure 1: The overall concept of the RiSH for elderly care.

The RiSH, by integrating a smart home, a service robot, and caregivers, would be helpful in the context of elderly care. The RiSH could assist the elderly by providing the following services: *1) comfort services*, *2) emergency services*, and *3) supporting services*. The *comfort services* help the elderly in EADLs, which include home appliance control, home temperature control, communication with caregivers and friends, locating things, and entertainment. The *emergency services* aim to detect, predict, prevent, and recover from critical conditions caused by abnormal behaviors or high risk situations such as falls and serious health problems. The *supporting services* aim to assist the elderly in ADLs and IADLs, compensate for physical decline, and aid recall of past actions. Examples are medication assistance and eating assistance. While the smart home and the robot can handle most scenarios with their local intelligence, they may occasionally have difficulty in perception, learning, and decision making. Therefore, providing the aforementioned services also requires the robot to get remote caregivers to assist in elderly care through teleoperation. It is also necessary for the robot to contact the cloud servers for help.

We propose the conceptual design of the robot-integrated smart home (RiSH) as shown in Fig. 1. It consists of three main parts: *1) a RiSH environment*, which consists of a home with a distributed sensor network, a home service robot with various onboard sensors, a resident who wears a body sensor network and uses a mobile device, a network of smart appliances, and a home gateway. All are connected to a local wireless network; *2) a RiSH cloud server*, which runs cloud services to provide intelligence outsourcing, facilitate teleoperation, and allow for management of the RiSH; *3) remote caregivers*, who can be informal caregivers such as children, relatives, or friends, or professional caregivers such as doctors or specialists. These three parts are connected to a public network.

4. Hardware Design of the RiSH Testbed

In this section, we present the hardware design of the RiSH testbed, which is shown in Fig. 2. This testbed consists of a RiSH prototype and an experimental infrastructure. The RiSH is organized according to the overall concept as discussed in the above section. The smart home is connected to the cloud server through the home gateway. The remote caregiver can provide remote daily care services to the elderly as well as collaborate with the home service robot to take care of them. The experimental infrastructure includes a sound simulation system that generates various sounds corresponding to daily human activities and an indoor localization system that provides the ground truth of human locations. Fig. 3 shows the floor plan design and the prototype of the RiSH testbed developed in our lab. The size of the apartment is about 5 m by 7 m. The major components of the RiSH are explained in the following subsections.

4.1. Home Service Robot

The ASCC home service robot is built on a Pioneer P3-DX base with an approximately 1.5 m-long aluminum frame holding up a touch screen monitor which is used for video communication and graphic user interface. The robot is equipped with

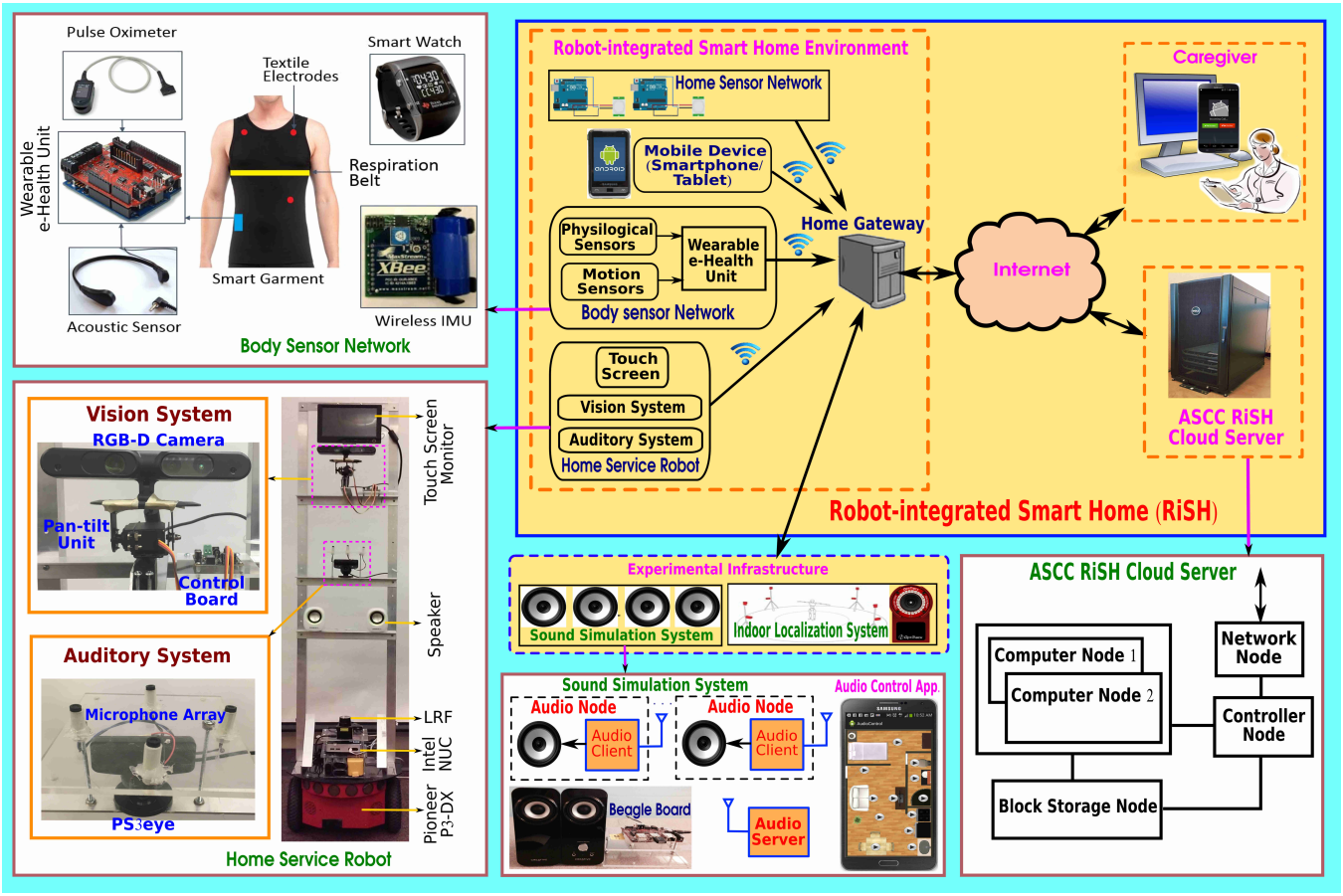


Figure 2: The hardware architecture of the RiSH Testbed.

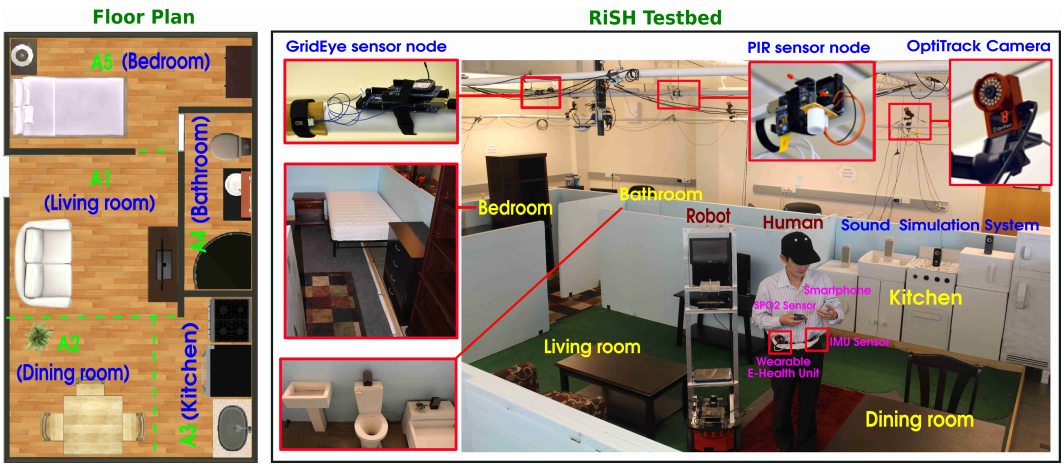


Figure 3: The floor plan (left) and prototype of the RiSH Testbed (right).

various sensors and devices which include a laser rangefinder (LRF), a vision system, an auditory system, an Intel NUC minicomputer, and batteries. The LRF, a Hokuyo URG-04LX-UG01 [28], is a low-power LRF with a distance range up to 5600 *mm*, an angle range up to 240^0, and an accuracy of 30 *mm*. The vision system is built using an Asus Xtion Pro Live RGB and Depth (RGB-D) camera [29], a pan-tilt unit, and a control board. The auditory system is built by extending the built-in microphone array of the PS3eye camera [30]. It features four microphones and employs technologies for echo cancellation and background noise suppression. This allows the auditory system to be used for speech recognition, sound localization, and sound separation in noisy environments. The microphone array operates with each channel processing 16-bit samples at a sampling rate up to 48 kHz per channel and a large dynamic range of signal-to-noise ratio up to 90 dB.

4.2. Body Sensor Network

The human body sensor network is a wearable unit that consists of physiological sensors, motion sensors, a smart watch, and a wearable e-Health Sensor Platform V2.0 from Cooking Hacks [31]. This unit is worn by the human subject and is used to collect physiological signals and activity data. It can also be used for self-health checkup or for remote healthcare. The data collected from this system includes Electrocardiogram (ECG), blood oxygen concentration (SpO_2), respiration rate, acoustic signals around the neck and body activity. The signals are sampled, framed, and timestamped to allow synchronization with ambient sensors in the smart home. We used our custom-built wireless inertial measurement unit (IMU) for collecting body activity information. The IMU consists of a VN-100 orientation sensor [32], an XBee module, and a power management unit. The VN-100 has a three-axis accelerometer, a three-axis gyro, and a three-axis magnetometer providing 3D orientation (roll, pitch, and yaw), acceleration, angular rate, and magnetic field reading.

4.3. Mobile Devices

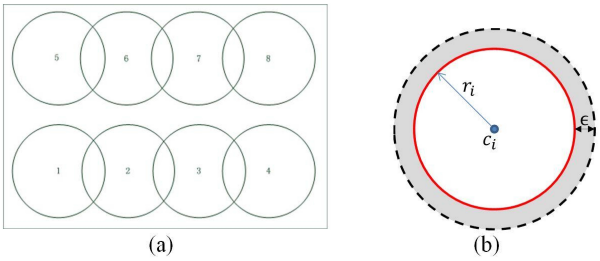
A mobile device such as a smartphone or a tablet is used as a user interface to control the home service robot. It can also be used to collect the data from the body sensor network. Furthermore, the caregiver can use mobile devices to remotely control the robot, connect to the body sensor network, as well as communicate with the older adult at their homes.

4.4. Home Sensor Network

The home sensor network consists of various sensors deployed in the home environment to collect signals regarding the location and activity of the resident. In the testbed, the home sensor network mainly consists of a distributed set of passive infrared (PIR) sensors and GridEye sensors [33] connected through the XBee protocol. The installation of the sensors is shown in Fig. 3. The PIR sensor node can provide binary motion information in its field of view by detecting the IR radiation emitted by the target. The sensor nodes are strategically mounted at different locations on the ceiling. The GridEye sensor node can provide 8 x 8 pixels of temperature data in its field of view. They are placed in larger rooms such as the living room and the bedroom, therefore enabling better tracking performance. As illustrated in Fig. 4-a, eight PIR sensor nodes are placed on the ceiling at a standard height of 8 ft and the coverage of each PIR sensor node is set to be a circle with a radius of 3.6 feet using a cylindrical lens cover. Data from these nodes are transmitted through the XBee protocol to the home gateway.

4.5. Home Gateway

The home gateway is a personal computer which serves as a local hub for data collection and processing in the smart home. It also enables the communication with the cloud server and the remote caregivers. The home gateway receives sensor data from the robot, the body sensor network, and the home sensor network. Data processing that requires less computational power



(a) (b)

Figure 4: (a) The arrangement of the PIR sensor network; (b) The sensing region of a PIR node.

and more realtimeness can be done locally on the home gateway. However, if the data processing requires more powerful computation, such as visual and audio understanding, human health diagnosis, and anomaly detection, it is more desirable to outsource the processing to the cloud server.

4.6. Private Cloud Infrastructure

The Smart Home Cloud is implemented using the private cloud infrastructure available in our lab. This system is set up using the open source Cloud Orchestration Software, OpenStack Juno [34]. The cloud provides an Infrastructure-as-a-Service (IaaS) solution to the RiSH. Three server nodes and a storage server are used in the setup as shown in Fig. 2. Most of the cloud management services including the message queue, authentication, databases, and networking are implemented on the controller node. Two Compute nodes host KVM (Kernel-based Virtual Machine) hypervisors and client services which create a virtualized environment for instances. Block storage is implemented using the storage server to provide the persistent storage for running instances. The OpenStack Sahara project [35] is also deployed to allow rapid configuration, reliable auto-deployment and scaling of Hadoop Clusters [35] in our cloud infrastructure. The Hadoop framework allows distributed processing of large data sets across clusters.

4.7. Experimental Infrastructure

The experimental infrastructure includes the indoor localization system and the sound simulation system. The indoor localization system is used to provide the location ground truth of the robot, the human, and other targets that need to be tracked. We adopted an OptiTrack motion capture system from Natural Point Inc. [36] for this purpose. This system consists of twelve OptiTrack V100:R2 cameras that can capture images within the range of 18 to 433 inches. The cameras are placed around the testbed to cover the whole area with a height of about 6 feet. The location tracking accuracy is about 95% at millimeter level. Moreover, the sound simulation system is developed to simulate the multiple sound events like those heard in a typical house. The sound simulation system includes multiple audio nodes, an audio server, and an audio control application [37]. The audio nodes are developed using the Beagleboard minicomputers [38] and speakers. The sound events in the bathroom, kitchen, living room, bedroom are either recorded in real environments or collected from the Internet. These sound samples form the ASCC

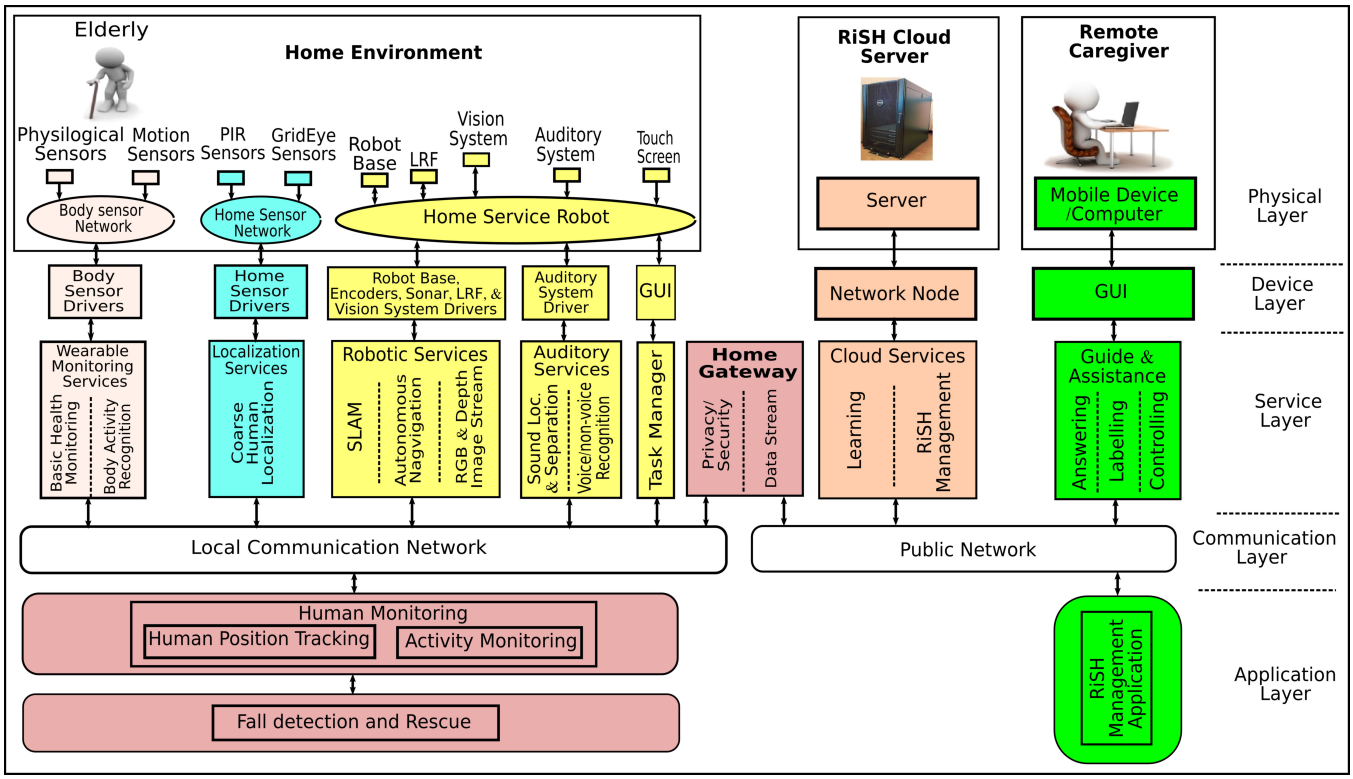


Figure 5: The layered architecture of the RiSH.

Sound Library (ASCCsoundLib). The audio control program on the smartphone can trigger the playback of a sequence of sound events associated with human activities or multiple simultaneous sound events using different audio nodes placed at different locations. For example, it can play both the TV sound in the living room and the showering sound in the bathroom or play a sequence of sound events related to the cooking activity in the kitchen. The script or schedule for playing the sound events is written in the JSON (JavaScript Object Notation) format.

5. Software Architecture of the RiSH

In this section, we present the software design of the RiSH. Recently, software architectures of assisted-living home environments have been developed by several groups, such as agent-based architecture [39], logical architecture [40], MAS (Multi-Agent System) architecture [41], and context-aware architecture [42]. However, these architectures are hard to be adopted in the RiSH, mainly because they are designed for systems that are homogeneous and do not provide a comprehensive software architecture to develop the RiSH. On the other hand, the RiSH is a complex system consisting of heterogeneous components. It is also desired to develop the RiSH in a way to achieve modularity, extensibility, customizability, and reusability. Therefore, we propose a layered architecture as shown in Fig. 5. The architecture consists of five layers: physical layer, device layer, service layer, communication layer, and application layer. Based on this architecture, basic services can be developed in each component without disrupting the rest of the system. Each component of the RiSH can provide basic service functions that can be used to develop higher level applications. New services, applications, and even new components can be added into the system while minimizing the impact to the existing system functions. Applications that provide elderly care services can adapt to the needs of the resident.

The physical layer is the hardware of the RiSH, which includes the hardware of the home service robot, human body sensor network, home sensor network, as well mobile devices and computers used by the caregivers. The hardware is managed by the device layer that provides the device drivers for the sensors and actuators on the robot, the graphical user interface (GUI) for the caregiver and the elderly, and the drivers for the sensor networks in the RiSH. The drivers for the robot are developed on ROS (Robot Operating System) [43], which runs in Ubuntu on the Intel NUC minicomputer. For the basic functions in the robot, we utilized existing packages from the ROS repositories to develop the device layer that interfaces with the robot base, the Hokuyo LRF, and the ASUS Xtion PRO LIVE camera. The driver for the auditory system is based on HARK [44], an open-sourced robot audition software consisting of modules for acoustic signal processing, sound source localization, sound source separation, and automatic speech recognition for various microphone array configurations.

The service layer contains the basic services provided by the major components in the RiSH. The home sensor network provides the coarse indoor human localization service. The body sensor network provides wearable monitoring services which

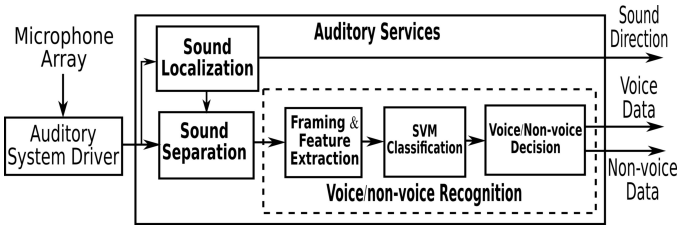


Figure 6: Auditory services for the home service robot.

include body activity recognition and basic health monitoring. The robot provides the robotic and auditory perception services, and a GUI. Developed on existing ROS packages, the robotic services include 2D SLAM (Simultaneous Localization and Mapping) based on the Rao-Blackwellized particle filters [45], autonomous navigation based on the adaptive (or KLD-sampling) Monte Carlo localization approach [46], and RGB-D image streaming. The task manager service processes commands from the users and enables the robot to make decisions and handle task failures. The home gateway offers privacy/security and data streaming between the RiSH and the caregiver, as well as the cloud services through public networks. The RiSH cloud management server stores the data from the RiSH and provides messaging services between the robot and the caregivers. Moreover, cloud learning services allow the robot to outsource the computationally intensive machine learning algorithms to the cloud servers. The caregivers provide the guide and assistance services such as answering robot's queries, labeling new training data, and controlling the robot to assist the elderly. Due to the page limit, only the major services are discussed in detail in the next section.

The application layer contains both low-level applications and high-level applications. The low-level applications are fundamental system functions of the RiSH such as human position tracking and activity monitoring. The high-level applications, developed based on the low-level applications, realize useful functions that humans can directly interact with. Based on the ROS network and the Websocket-based Rosbridge [43], the communication layer provides seamless connection among the modules in the service layer and the application layer through both the local and public networks.

6. Basic Services

This section presents key functions in the service layer, which include auditory perception, coarse human localization, and wearable monitoring services.

6.1. Auditory Perception Services

Although the vision system on the robot provides abundant information regarding the home environment and human activities, it is not always possible to observe the residents due to occlusion or poor lighting condition. Therefore it is desirable to equip the home service robot with auditory perception services. As shown in Fig. 6, audition services perform sound localization, sound separation, and voice/non-voice recognition from the four-channel audio stream obtained by the auditory system driver. These services are based on the HARK audition software.

6.1.1. Sound Localization and Separation

Sound localization is implemented based on the GEVD-MUSIC (Generalized EigenValue Decomposition-Multiple Signal Classification) method [47]. This method localizes sound sources by computing an eigenvalue decomposition vector of the correlation matrix between the inputs signal channels, then calculating MUSIC spectrum of this vector and the impulse responses (transfer functions) of microphones. The DoAs (Direction of Arrival) which have the largest values of the spectrum power are the sound source direction results.

The sound from Ns sources is affected by the transfer function of each microphone $H_i(k)$ in space and perceived by M microphones as expressed by the following equation:

$$X_i(k) = \sum_{j=1}^{Ns} H_i(k) S_j(k) + N_i(k), \; i = 1, 2, ..., M \tag{1}$$

where $S_j(k)$ terms the Fourier transform of the jth sound source at the frequency k; $N_i(k)$ is the additive noise that includes environmental noise and electronic noise in each microphone. Sound source separation extracts the sound in each direction that is estimated by the sound localization from the recorded sound $X(k)$. The Fourier transform of separated sound $Y_j(k)$ is obtained from the following equation:

$$Y_j(k) = W_j(k) X(k) \tag{2}$$

The separation matrix $W_j(k)$ is estimated by Geometric-Constrained High-order Source Separation (GHDSS) [48] which has the highest total performance in various acoustic environments.

6.1.2. Voice/non-voice Recognition

The separated sounds are classified into voice and non-voice by the voice/non-voice recognition (VNR) algorithm. To achieve this, we use the support vector machine (SVM) algorithm. The kernel function widely used in SVM for audio applications is the Gaussian radial basis function (RBF) as follows:

$$K(x_i, x_j) = exp(-\gamma ||x_i - x_j||^2) \tag{3}$$

where γ is a parameter that controls the width of Gaussian, which is estimated from the variance of the distribution function of the training data. RBF-SVM aims to construct the decision function for the data point x based on N support vectors $\{x_k\}_{k=1}^N$ and labels $\{y_k\}_{k=1}^N$ as follows:

$$y(x) = sign[\sum_{k=1}^{N} \alpha_k y_k K(x_k, x) + b] \tag{4}$$

where α_k is the weight assigned to the support vector x_k, b is a constant bias. In order to train the SVM, the audio training data consisting of labeled voice and non-voice segments are decomposed into frames. Statistical features are calculated for

each frame of the audio stream. We use the 31-dimension feature vector x that consists of 12-band MFCCs (Mel-frequency cepstral coefficients), delta of 12-band MFCCs, log energy, zero cross rate, entropy, centroid, spread, skewness, and kurtosis [49]. The trained SVM model can classify frames of separated sounds into voice or non-voice.

6.2. Coarse Human Localization Service

The coarse human localization service estimates the rough human location by using the PIR network. Each PIR node detects the human motion inside its sensing region. Therefore the human location is approximately estimated to be within the sensing region once the sensor gives a high output. To achieve that, a new PIR sensor observation model is developed based on the existing model in [50]. Our new PIR sensor model is expressed as follows:

$$P(z_k^{PIR,i}|s_k) = \begin{cases} p^{z_k^{PIR,i}}(1-p)^{1-z_k^{PIR,i}} & if \quad |s_k - C_i| \le r_i \\ q^{z_k^{PIR,i}}(1-q)^{1-z_k^{PIR,i}} & if \quad r_i \le |s_k - C_i| \le r_i + \epsilon \\ 1 - z_k^{PIR,i} & if \quad |s_k - C_i| \ge r_i + \epsilon \end{cases} \tag{5}$$

where p is the probability of detection; q is the probability of false alarm; $z_k^{PIR,i}$ is the binary output {0,1} from PIR sensor i at time k; s_k is the human state which is the 2D location; C_i and r_i are the center and the radius of the sensing region of PIR sensor i, respectively. We discovered that false alarm may occur when the human is not in the sensing range, but not too far away from the sensor, which is depicted by the gray area ϵ as shown in Fig. 4-b. Inside the gray area, those probabilities estimated from our measurements are ($p = 0.9$ and $q = 0.05$). If the human is out of the dashed circle, the false alarm rate q becomes 0.

6.3. Wearable Monitoring Services

Wearable monitoring services aim to understand the human state using the data collected from the body sensor network. In this work, we mainly focus on basic health monitoring and body activity recognition.

Basic health monitoring service is developed to record the ECG signals, heart rate, SpO_2, and respiration rate. Moreover, a front-end web application is built on the cloud server to serve as a portal for caregivers and the robot. In addition, a mobile client application is developed to browse the health data on mobile devices such as smartphones. This allows the caregivers to access the health data of the older adult in both offline and online modes.

Human body activity has a direct impact on the accuracy of health assessment. It is well known that body movement can introduce artifacts into physiological signals. For example, a fast heart rate inferred by ECG signals is not necessarily considered tachycardia, if the human activity is recognized as vigorous running. To recognize body activities, motion data from wearable IMU sensors can be used. In this work, we use a single IMU attached to human's right thigh. The motion data consist of 3D linear acceleration and 3D angular rate captured at a rate of $20Hz$. Those 6 raw signals are filtered to remove some noise, and then features are extracted using the sliding window method with a 2-second window size, and a 1-second step size. Features of acceleration data and angular rate data are calculated in both the time and frequency domains. The selected features in the time domain are the mean, standard deviation, median, maximum, minimum, squared sum of magnitude data. In the frequency domain, the selected features are the peak frequency in the spectrum of magnitude data below $5Hz$, number of peaks in the spectrum of magnitude data below $5Hz$, and integral of the spectrum of magnitude data from 0 to $5Hz$. The body activities of interest include static postures (lie, sit, and stand) and dynamic activities (walk, turn left, turn right, stand-to-sit, sit-to-stand, sit-to-lie, and lie-to-sit). These body activities are recognized based on the Gradient Boosting Decision Tree algorithm [51].

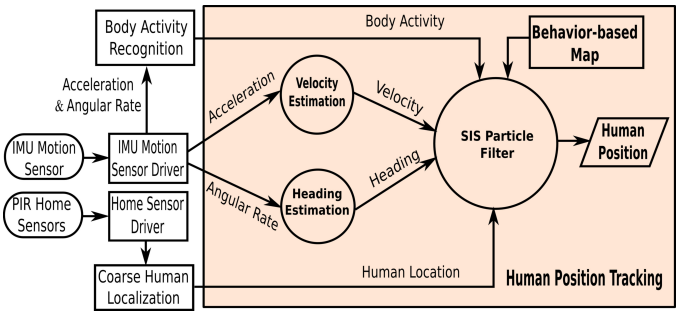


Figure 7: Human position tracking through multi-sensor fusion.

7. Applications for the RiSH

The applications are developed based on the services from the service layer. We first present two low-level applications of monitoring, which are human position tracking and activity monitoring. Then, a high-level application is implemented which focuses on fall detection and rescue.

7.1. Human Position Tracking

Indoor human position tracking plays an important role in living assistance and emergency response. This application estimates the human position based on multi-sensor fusion using data from PIRs and an IMU. The PIR network is used to provide approximate location information without capturing any image, which minimizes the violation of the resident's privacy. An IMU sensor attached to the human's thigh is used to estimate the resident's walking velocity and heading angle in order to provide body movement estimation in a short period. In addition, this approach takes advantage of the correlation between human locations and their activities in home environments to improve the localization accuracy.

The overview of the approach is demonstrated in Fig. 7. The IMU sensor provides motion data including 3D acceleration and 3D angular rate which are used to estimate the human velocity and heading direction. The coarse location data are obtained from the PIR sensor network. Both motion data and coarse location data are fused through a Particle Filter module

to estimate the accurate human position. A behavior-based map is used to represent the correlation between human location and activity. In this map, the position of walls and furniture (such as tables, chairs and beds), as well as other facilities are identified. This map basically encodes the location probability of the resident when he/she is conducting certain activities such as walking, sitting, and lying. This is important information that can be used to improve the localization accuracy in a Bayesian filtering framework as presented in our previous work [52][53].

7.1.1. IMU-Based Motion Model

The IMU sensor attached to the human body can provide important information regarding the activity and motion of the wearer. The motion model of the wearer includes the velocity and heading. The moving velocity is estimated based on the recognized human body activities. The heading is computed from the gyro output.

Once the walking activity is detected, the velocity v_k is estimated by integrating the acceleration during the swing phase of each step, which includes a stance phase and a swing phase [54] as follows:

$$v_k = \begin{cases} a_{xy}T_s + v_{k-1} & if \quad \text{swing phase} \\ 0 & if \quad \text{stance phase;} \\ & \text{or standing, sitting, lying} \end{cases} \tag{6}$$

where $a_{xy} = a_x + a_y$ is the horizontal acceleration; a_x and a_y are accelerations along the x and y coordinate axis, respectively; T_s is the sampling period.

Heading provides walking direction and this information can be directly read from the IMU outputs (yaw, pitch, and roll). However, in indoor living environments, due to the magnetic disturbance caused by many devices such as computers, microwaves, and other electrical appliances, the accuracy of heading read from the IMU is neither accurate nor reliable. Therefore, we developed a different approach to estimating the walking direction. In our approach, the angular rate from the IMU is used to estimate the heading changes when the human is walking. Similar to velocity estimation, the heading angle change can be estimated by integrating the angular rate over time. It is clear that the drifts of the gyroscope may lead to poor results since the errors are also integrated. In order to overcome this problem, we utilize the results from activity recognition, in which turning right and turning left are detected. The time t_0 when the human starts turning, and the time t_1 when the human finishes that turning are recorded. Then the estimated heading θ_k is calculated by adding the amount of angle change to the previous heading angle θ_{k-1} and the measurement noise $N(0, \sigma_\theta)$ which has zero mean and a standard deviation value of σ_θ. The standard deviation is set to $\pi/6$ based on the measurements obtained in our experimental testing.

The propagated position can be expressed by the following equation:

$$\begin{bmatrix} x_k \\ y_k \end{bmatrix} = \begin{bmatrix} x_{k-1} + T_s v_k \cos(\theta_k) + n_k^x \\ y_{k-1} + T_s v_k \sin(\theta_k) + n_k^y \end{bmatrix} \tag{7}$$

where T_s is the sampling period; v_k and θ_k are the velocity and heading at time k sampled from normal distribution $N(v_k, \sigma_v)$ and $N(\theta_k, \sigma_\theta)$ with velocity mean v_k and heading mean θ_k which are estimated from the acceleration and angular rate information as mentioned earlier; n_k^x and n_k^y are the process noise along the x and y coordinate axis, respectively. The standard deviation σ_v and σ_θ are set to 20% of the mean velocity and $\pi/6$ rad respectively.

7.1.2. Behavior-Based Map

To facilitate the behavioral landmark-assisted localization, we introduce the concept of the behavior-based map and conduct the location inference in a Bayesian framework through particle filtering. Basically, the behavior-based map can be represented by an accessibility probability function (APF) which can be defined as follows:

$$P(s_k|a_k) = \varphi \tag{8}$$

where φ is the probability of being at location s_k when the human is conducting activity a_k which can be either sit, lie or walk. We assume the furniture location in the home is fixed and the human can walk anywhere except places occupied by furniture and walls.

7.1.3. Human Localization through Sensor Fusion

By fusing the two channels of information: PIR data and IMU data, we can derive more accurate location estimate. Although Kalman filtering is a popular method used in many navigation systems, the requirement of linear model and Gaussian noise is not satisfied in our case. Therefore, we choose particle filtering for human localization with multiple data sources.

The particle filter, or Sequential Monte Carlo, is one of the numerical methods to estimate posterior density function $P(s_k|z_{1:k})$ of system state s_k given the observation data up to time k, $z_{1:k}$. It is an iterative process consisting of two main steps (prediction and update), which employs two models, system model (or motion model) and measurement model. Each particle represents a possible state, and all particles can approximate the posterior probability distribution of the state. The prediction step is based on the state motion model $P(s_k|s_{k-1})$, in which the propagation of particles utilizes velocity and heading angle estimates provided by the IMU. The propagation of particles is expressed by Equation (7). The update step is based on the observation model $P(z_k|s_k)$, in which the PIR data and activity information derived from the IMU data are used to compute the weight of each particle. At the end of the loop is the resampling step which removes low-weight particles and generates new particles with normalized weight.

Here we explain how to update the weight of particles. Basically, we need calculate the likelihood of the observations

$z_k^i = [z_k^{PIR,i}, z_k^{IMU,i}]$ given a human location s_k

$$\begin{aligned} P(z_k^i|s_k) &= P(z_k^{PIR,i}, z_k^{IMU,i}|s_k) \\ &= P(z_k^{PIR,i}|s_k) \cdot P(z_k^{IMU,i}|s_k) \end{aligned}$$

Here

$$\begin{aligned} P(z_k^{IMU,i}|s_k) &= \sum_{a_k} P(z_k^{IMU,i}, a_k|s_k) \\ &\propto \sum_{a_k} P(z_k^{IMU,i}, s_k|a_k) \cdot P(a_k) \\ &= \sum_{a_k} P(z_k^{IMU,i}|a_k) \cdot P(s_k|a_k) \cdot P(a_k) \\ &= \sum_{a_k} P(z_k^{IMU,i}|a_k) \cdot P(a_k|s_k) \end{aligned} \quad (9)$$

Therefore we have

$$P(z_k^i|s_k) = P(z_k^{PIR,i}|s_k) \sum_{a_k} P(z_k^{IMU,i}|a_k) \cdot P(a_k|s_k) \quad (10)$$

The above equation calculates the likelihood of observing the PIR data $z_k^{PIR,i}$ and IMU data $z_k^{IMU,i}$ given that the location s_k of the human is known. With the correlation between activity and location $P(a_k|s_k)$, the behavior-based map is integrated into the update step. The weight of each particle is updated to generate the posterior distribution $P(s_k|z_{1:k})$. Let $w_k^{PIR,i} \sim P(z_k^{PIR,i}|s_k)$ be the weight of ith particle at time k based on the PIR sensors, and $w_k^{IMU,i} \sim P(z_k^{IMU,i}|s_k)$ be the weight based on the IMU sensor. $P(z_k^{IMU,i}|s_k)$ can be calculated according to Equation (9). The updated weight should be the product of these two weights and the previous weight

$$w_k^i = w_k^{PIR,i} \cdot w_k^{IMU,i} \cdot w_{k-1}^i \quad (11)$$

If the number of particles with low weight reaches a certain threshold, resampling should be conducted. Otherwise, the particle filter will degenerate when there are only a few heavy weighted particles.

7.2. Human Activity Monitoring

Many activities such as talking, eating, cooking, having shower, and brushing teeth are hard to recognize based on one IMU motion sensor. However, many of these activities generate sound events. Therefore, human activity monitoring can be realized by combining the wearable sensor-based body activity recognition and the sound event recognition. Generally, it is very hard to recognize sound events due to the diversity of the sounds associated with the same event. One reason that allows humans to distinguish different events is their knowledge of the context, which helps them form predictions and adapt their perception to the environment [55]. Context-aware sound event recognition (CoSER) allows us to associate contextual information with sound events, which helps recognize the sound events. Therefore we propose a novel context-based method for sound event recognition using a Dynamic Bayesian Network (DBN) that models intra-temporal and inter-temporal

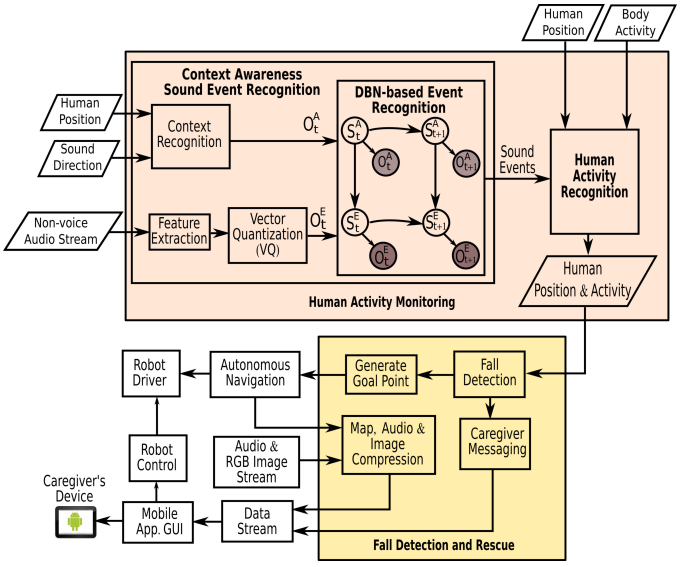


Figure 8: Human activity monitoring and fall detection and rescue.

constraints among the context and sound events. As shown in Fig. 8, context-based sound event recognition consists of four main parts: context recognition, feature extraction, vector quantization (VQ), and DBN-based event recognition.

7.2.1. Context Recognition

The context of sound events in home environments typically consists of human's location, sound source's direction, robot's location, time, and environmental condition. In this work, we consider location as the context of sound events, which can be estimated from human's location H and sound source's direction θ using the naive Bayes classifier. In this paper, our main goal is to evaluate sound event recognition based on the context. Therefore the context is recognized only from human's location. The context observation is obtained as follows:

$$O^A = \arg\max_{A \in \{A_1, A_2, \ldots A_K\}} \{P(A|H)\} \quad (12)$$

where $P(A|H)$ is the probability of sound events that occur at location A given human's location H.

7.2.2. Feature Extraction

Statistical features are calculated for each frame of the audio stream. The length of the frame ranges from 5 ms to 150 ms and the overlap between two adjacent frames is set to a half of the frame size. We also use the 31-dimension feature vectors as presented in Subsection 6.1.2.

7.2.3. Vector Quantization

Vector quantization (VQ) is to compress a data set into a small set of representatives, which reduces the space to store data, but still maintains sufficient information. The LBG (Linde-Buzo-Gray) algorithm [56] is applied to design codebooks in the VQ and transform audio feature vectors to the labels of codewords. These labels are used as the observation O^E of event sounds

7.2.4. DBN-based Event Recognition

The location contexts that are to be recognized are the rooms in a house, including the living room, dining room, kitchen, bathroom, and bedroom. In indoor environments, sound events and human's locations are highly correlated. They have both intra-temporal causal relationship and inter-temporal constraints, which are modeled by the two-level dynamic Bayesian network model shown in Fig. 8. The human location observation O^A given by the PIR network is mapped into N_A semantic areas. The observation O^E of event sounds is computed from the audio stream. The high-level of the model represents the sound context, or the location of the sound S^A and the low-level represents sound events S^E. The dependencies between the nodes in the DBN have both spatial and temporal components. The observation O_t^A and O_t^E are dependent on the corresponding intra-temporal hidden states S_t^A and S_t^E, respectively. The context S_{t+1}^A at time $t+1$ depends on the previous context at time t. The sound event S_{t+1}^E at time $t+1$ depends on the previous sound event at time t and the context at time $t+1$. The state transition probability distribution in each level reflects the intra-temporal dependency and is trained by using the EM (Expectation Maximization) algorithm as proposed in [57]. Based on the DBN model, we have the probability of the sequence as follows:

$$\begin{aligned} &P(S_{1:T}^A, S_{1:T}^E, O_{1:T}^A, O_{1:T}^E) \\ &= P(S_1^A)\prod_{t=2}^{T} P(S_t^A|S_{t-1}^A)\prod_{t=1}^{T} P(O_t^A|S_t^A) \\ &P(S_1^E|S_1^A)\prod_{t=2}^{T} P(S_t^E|S_{t-1}^E, S_t^A)\prod_{t=1}^{T} P(O_t^E|S_t^E) \end{aligned} \tag{13}$$

where T is the length of the observation sequence.

The above general formula cannot be used for realtime recognition because of its computational complexity. Therefore, the short-time Viterbi algorithm [58] is applied to estimate the probability recursively. The algorithm retrieves the state sequence, which has the maximum likelihood given the observation sequence from time 1 to T [59].

7.3. Fall Detection and Rescue

Older adults often face the risk of fall while living alone at their home. It is highly desirable to equip the robot with the capability of fall detection and rescue. Therefore, we conduct a case study of fall detection and rescue based on the two low-level applications: position tracking and activity monitoring. The context-aware sound event recognition allows the robot to not only recognize the human activity based on sound events but also detect the body falling sound. As shown in Fig. 8, the fall detection module observes the human position and activity detected by these low-level applications. If a fall event is detected, a goal point of the robot is generated in four steps. The first step is to create an initial goal point at the human position with the same orientation as the robot. Then an initial path plan is generated by the autonomous navigation service. Next, the intersection point between the initial path and the safety circle around the human is calculated. Finally, a new goal point is created at the intersection point with the orientation directed to the human position and a new path plan is also updated. The navigation module autonomously controls the robot to move to this goal point to check what is happening with the human and a message is sent to the caregiver requesting for assistance. The map, audio, and image are also compressed and then sent to an application on the caregiver's mobile device through the Websocket protocol. The caregiver can control the robot and have video conferencing with the older adult to respond to such an urgent situation.

8. Experiments and Results

We conducted physical experiments to test and evaluate the theoretical framework using the RiSH testbed. The test mainly focuses on the following parts: auditory perception services, human body activity recognition, human position tracking, human activity monitoring, and fall detection and rescue. The experiments show the operation of the components in the RiSH and the capabilities of the home service robot in monitoring and assisting the resident.

To evaluate the performance of the proposed classification and recognition models, evaluation measures are computed from confusion matrices. These measures include the true positive TP, true negative TN, false positive FP, false negative FN, recall R, precision Pr, F1 score F_1, accuracy acc, and overall accuracy ACC - the average accuracy of the total number N_tests of tested instances for every classes. The R_i, P_i, F_{1i} and acc_i of the class i and ACC are computed as follows:

$$R_i = \frac{TP_i}{TP_i + FN_i} \tag{14}$$

$$Pr_i = \frac{TP_i}{TP_i + FP_i} \tag{15}$$

$$F_{1i} = 2 \cdot \frac{Pr_i \cdot R_i}{Pr_i + R_i} = \frac{2TP_i}{2TP_i + FP_i + FN_i} \tag{16}$$

$$acc_i = \frac{TP_i}{TP_i + FP_i + FN_i} \tag{17}$$

$$ACC = \frac{\sum_{i=1}^{N} TP_i}{N\_tests} \tag{18}$$

8.1. Auditory Perception Services

In this part, we present the testing and evaluation of the sound localization and voice/nonvoice recognition. The sound localization was tested using the sound simulation system and the OptiTrack system. To fully evaluate the accuracy of sound localization, the speaker was placed at different directions (0°, ±45°, ±90°, and ±135°) and distances (0.5 m, 1 m, 2 m and 4 m) with respect to the robot. The OptiTrack system obtained the relative locations between the speaker and the robot, which were treated as the ground truth. At each location, we ran the sound localization algorithm 10 times and calculated the mean and the standard deviation which are shown in Table 1. This

Table 1: Results of sound localization

Distance	Errors	Direction				
		0°	±45°	±90°	±135°	Sum
0.5 m	Mean [°]	-0.3	-0.1	-0.2	0.2	-0.2
	Std [°]	1.5	2.0	1.9	1.6	1.7
1 m	Mean [°]	0.6	-0.8	-0.2	0.5	-0.1
	Std [°]	2.2	2.1	2.3	2.0	2.2
2m	Mean [°]	0.1	0.2	-0.1	0.1	-0.3
	Std [°]	3.1	2.9	3.0	2.7	2.9
4m	Mean [°]	1.9	0.4	-1.0	-0.9	0.6
	Std [°]	4.3	3.7	4.0	3.8	3.9

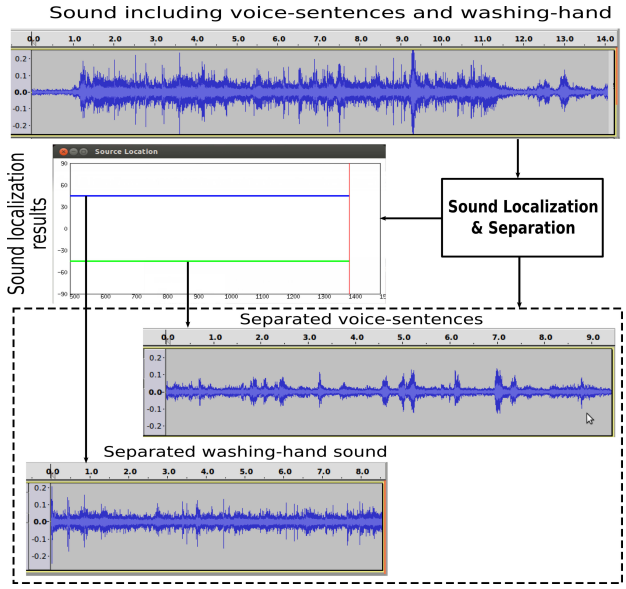


Figure 9: An example of sound localization and separation.

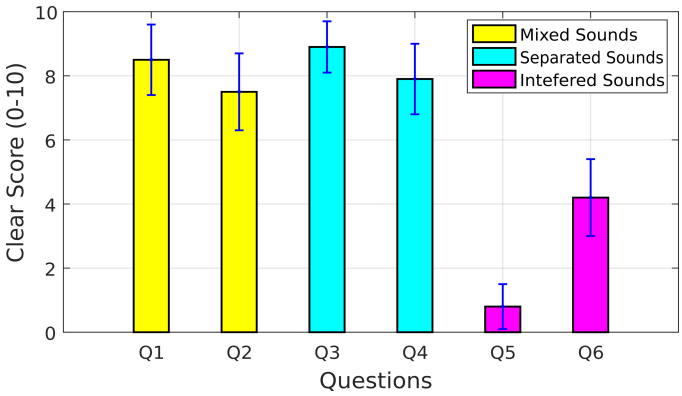


Figure 10: The average scores and the standard deviations of the user study of sound separation.

table shows that the sound sources can be localized with a reasonable accuracy. From Table 1, the detection errors are small for the same distance and not very sensitive to the direction of the sound sources. However, the errors increase with the distance. The standard deviation of errors is less than 2° at 0.5 m and less than 4° at 4 m away from the robot.

To evaluate the sound separation, event sounds and voice sounds in our ASCCsoundLib were randomly divided into voice/non-voice pairs that were simultaneously played by two speakers with an SNR (Signal-to-Noise Ratio) of 3 dB. The robot successfully separated each mixed sound into event sound and voice sound. An example of the sound localization and separation result is shown in Fig. 9, from which we can see that the two voice and non-voice sources can be separated. In this experiment, speaker 1 and speaker 2 placed at -45° and 45°, respectively, played the five speech sentences and the washing-hands sound, respectively. The robot was able to localize and track both sources correctly and then separate them into two sounds that have the similar waveforms as their original sounds. Although each sound is still interfered by the other, the separated sounds can be used in voice/non-voice recognition.

Moreover, we conducted a user study to evaluate the sound separation using 6 questions as follows:

Q1: How clear are the event sounds you heard from the mixed sounds?

Q2: How clear are the voice sentences you heard from the mixed sounds?

Q3: How clear are the separated event sounds?

Q4: How clear are the separated voice sentences?

Q5: How clear are the interfered voice sentences you heard from the separated event sounds?

Q6: How clear are the interfered event sounds you heard from the separated voice sentences?

A total of 10 human subjects who are graduate students from our department participated in the study. The score for each question was rated from 0 to 10 (0-impossible, 1-super hard, 2-very hard, 3-hard, 4-not too hard, 5-possible, 6-quite easy, 7-pretty easy, 8-easy, 9-very easy, and 10-super easy to recognize). The average score and the standard deviation for each question are shown in Fig. 10. The separation makes the sounds easier to recognize. Most users rated the question Q5 around 1 that means it was super hard to recognize the interfered voice sentences from the separated event sounds. However, most of them felt not too hard to recognize the interfered event sounds from the separated voice sentences. The main reason is that it is impossible to totally remove event sounds that have a wide frequency spectrum overlapping with that of voice signals.

The SVM-based voice/non-voice recognition (VNR) is not new, but it is one of the best methods for the VNR. We do not provide comparative results of our work with other works because we used different audio datasets to train and test the SVM. Therefore it is not fair to compare with other works. Most of the state-of-the-art SVM-based VNR methods have not been tested in audio data obtained through the sound source separation algorithms. Both voice and non-voice signals were generated at the same time. We used sound separation to improve the SVM-based VNR performance in such cases. Therefore, in this paper, we focus on evaluating the SVM algorithm with audio data which consists of contaminated voice and non-voice signals due to the fact that the sound separation algorithm cannot completely separate the two sound sources. In order to evaluate the VNR algorithm, event sounds and voice sounds in our ASCCsoundLib were divided randomly into voice/non-voice pairs that were simultaneously played by two speakers with an SNR

Table 2: Cross-validation accuracy of the VNR with the hyperparameters at $(C, \gamma) = (2^{10}, 2^{-20})$

	Fold 1	Fold 2	Fold 3	Average
ACC	0.9445	0.8936	0.9231	0.9204

Table 3: Evaluation of the VNR performance

	TP	*FN*	*FP*	*TN*	*R*	*Pr*	F_1	*acc*
Voice	50226	3819	4978	51512	0.929	0.910	0.919	0.851
Non-voice	51512	4978	3819	50226	0.912	0.931	0.921	0.854

Table 4: Cross-validation accuracy of the body activity recognition using IMU

	Fold 1	Fold 2	Fold 3	Average
ACC	85.21%	86.39%	86.81%	86.14%

of 3 dB. Each pair was played 5 times with different locations of two speakers. The robot separated each pair into two different sounds. These separated sounds and the clear sounds in the ASCCsoundLib were used to evaluate our SVM-based VNR algorithm using 3-fold cross-validation. The collected audio data set which includes 30.1 minutes of event sounds and 28.8 minutes of voice sounds was partitioned into 3 subsets with equal sizes. We labeled non-voice for all frames in the event sounds and voice for all frames in the voice sounds. Sequentially one subset was tested using the SVM trained on the remaining 2 subsets. The cross-validation accuracy is the average of the rate of audio frames that are correctly classified in 3 testing subsets. The soft-margin SVM with the soft-margin constant C and the RBF kernel was implemented for the VNR. The pair of SVM hyperparameters (C, γ) that has the best cross-validation accuracy was found via grid-search on $C = \{2^5, 2^6, ..., 2^{15}\}$ and $\gamma = \{2^{-15}, 2^{-11}, ..., 2^{-25}\}$. The best hyperparameters were found at $(C, \gamma) = (2^{10}, 2^{-20})$ with the best cross-validation accuracy of 92.04% as shown in Table 2. The evaluation result is shown in Table 3. Both recalls and precisions are more than 91% and the accuracy of two classes is more that 85% at the frame level. The experiments show that more than 98% of these separated sounds have more than 70% of frames that are correctly recognized into voice or non-voice frames. Therefore, when the thresholds of voice/non-voice decisions are set to be around 70%, the voice/non-voice recognition results of the robot can reach an accuracy of 98% for the whole separated sounds.

8.2. Wearable Monitoring Services

This part presents the results of health monitoring and body activity recognition. All health data were successfully uploaded to the cloud server and made available to the caregivers. This allows caregivers to collect these health data from the elderly, which consist of ECG signals, heart rate, SpO_2, as well as the location and activity. Overall, this application will serve numerous purposes including vital sign monitoring for older adults, follow-up of discharged patients for rehabilitation and recovery. This also allows for researchers to study the relationship between daily activity and health.

For body activity recognition, experiments were carried out by 12 subjects (males and females, age: 25-45). Each subject wore an IMU on the right thigh and performed different activities for about 15 minutes. The data were labeled manually during the experiments by using a wireless keypad with each key assigned to a corresponding activity (*0:sit-to-lie, 1:lying, 2:lie-to-sit, 3:sitting, 4:turn left, 5:stand-to-sit, 6:turn right, 8:sit-to-stand, 7:standing, 9:walking*). The data have 12046 observations, which were split into 3 subsets with equal sizes for 3-fold cross-validation. The target variables have 10 values corresponding to 10 different activities. As shown in Table 4, the cross-validation accuracy is more than 86%. The evaluation result of each class is shown in Table 5. Besides, Table 6 shows the comparison between our method and recent projects of body activity recognition, in terms of the placement of IMUs, classified activities, and overall accuracy.

Table 5: Evaluation of the performance of body activity recognition

Activity	*TP*	*FN*	*FP*	*TN*	*R*	*Pr*	F_1	*acc*
Sit-to-lie	496	138	176	11236	0.782	0.738	0.760	0.612
Lying	1209	125	144	10568	0.906	0.894	0.900	0.818
Lie-to-sit	550	120	162	11214	0.821	0.772	0.796	0.661
Sitting	1441	63	78	10464	0.958	0.949	0.953	0.911
Turn-left	1043	173	161	10669	0.858	0.866	0.862	0.757
Stand-to-sit	962	277	299	10508	0.776	0.763	0.770	0.625
Turn-right	865	245	273	10663	0.779	0.760	0.770	0.625
standing	1240	298	217	10291	0.806	0.851	0.828	0.707
Sit-to-stand	1130	113	72	10731	0.909	0.940	0.924	0.859
Walking	1440	118	88	10400	0.924	0.942	0.933	0.875

Table 6: Recent projects of body activity recognition using IMUs

Project	Placement of IMU	Classified Activities	Accuracy
[60]	2 IMUs worn on thigh and waist	Lying down; Sitting; Standing; Walking	98%
[61]	2 IMUs worn on thigh and shank	Agility cut; Walking Jumping on box; Jogging; Sprinting; Kicking	98.3%
[62]	1 IMU and 8 pressure sensors placed in a shoe	Sitting; Standing; Level walking; Upstairs; Downstairs; Uphill; Downhill; Elevator up; Elevator down	97.41%
[63]	1 IMU worn on thigh	Standing; Sitting; Walking	85.56%
Our project	1 IMU worn on thigh	Static postures (lying, sitting, and standing); Dynamic activities (walk, turn left, turn right, stand-to-sit, sit-to-stand, sit-to-lie, and lie-to-sit)	86.14%

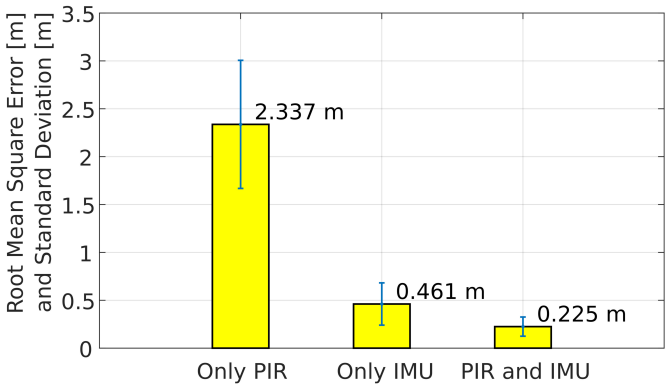


Figure 11: The root mean square error and standard deviation of human position tracking

.

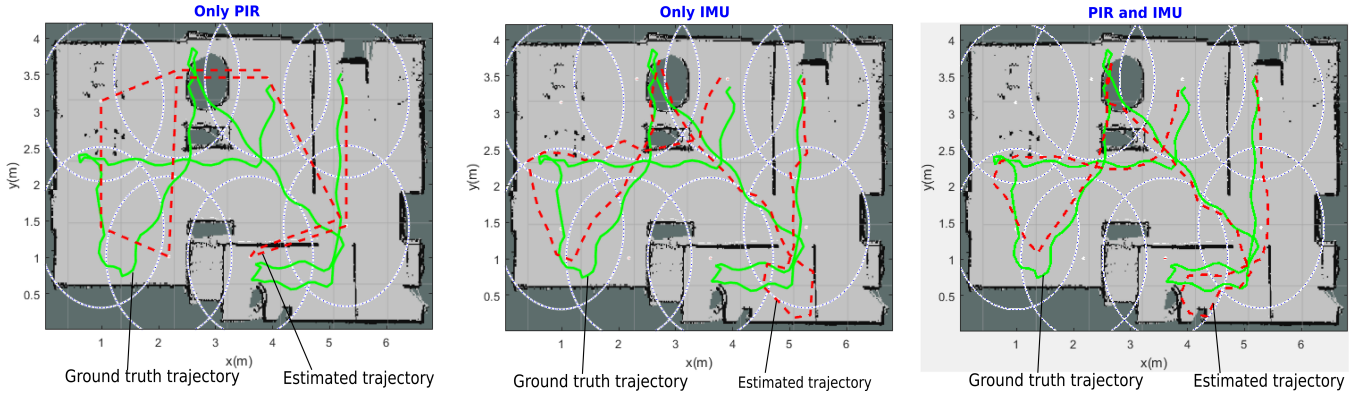


Figure 12: An example of ground truth trajectories and estimated trajectories.

8.3. Human Position Tracking

This part presents the results of human position tracking. Since our placement of PIR and IMU sensors is different from other previous projects, it is not fair to compare our method with other methods. However, we provide the comparison of results between different sensing modalities to evaluate the human position tracking method. We designed human motion trajectories corresponding to common human activities in a home environment. Each trajectory includes a series of actions such as entering the house, walking to the dining table, sitting on the chair, walking to the refrigerator, walking to the sofa, going to the bathroom, and going to the bedroom. In our experiments, 12 human subjects walked following 9 different trajectories for 3 times. The results of localization and tracking were compared based on the sensor data used. The localization errors were calculated by comparing the ground truth from the OptiTrack motion capture system and the estimated trajectories using only PIRs, using only IMU, and using both PIRs and IMU, correspondingly. The root mean square errors and standard deviations of the nine trajectories are shown in Fig. 11. It can be seen that using both sensors results in the smallest errors compared to using only one type of sensors.

One example of the experiment is shown in Fig. 12. The experiment consists of the following human activities: getting up from the bed, walking into the bathroom, going to living room to watch TV, going to the kitchen to prepare breakfast, getting to the dining table, and getting out of the house. The solid green line trajectory shows the ground truth and the dashed red line is the estimated trajectory. The distance error between the predicted path and the ground truth is reduced after each time the human sits down on the chair or sofa. The length of the total route is 35.6 m and the root mean square errors of the trajectories estimated by using only PIRs, only IMU, and both PIRs and IMU are 2.25 m, 0.411 m, 0.140 m, respectively.

8.4. Human Activity Monitoring

Experiments were conducted to evaluate human activity monitoring based on sound event recognition. The sound events in the bathroom, kitchen, living room, bedroom, and dinning room were recorded or collected from the Internet. Currently, the ASCCsoundLib has 38 sound events including 6 dining

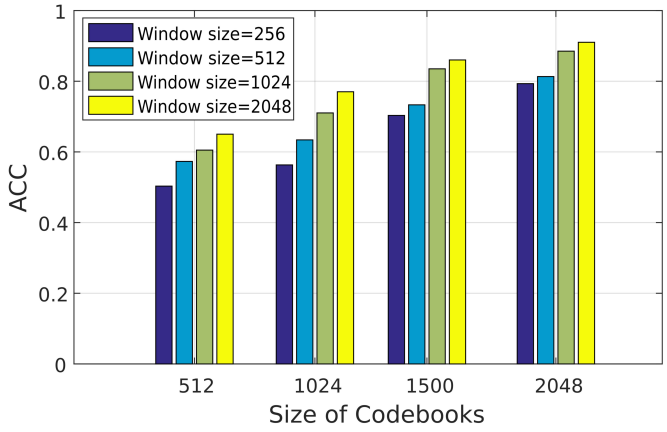


Figure 13: Overall accuracy of the CoSER with respect to the size of codebooks and the window sizes.

room sounds, 3 living room sounds, 8 bathroom sounds, 14 kitchen sounds, 2 bedroom sounds, and 5 other sounds that occur in every room. Sound events and their labels are as follows:

Dining room: *2:eating-cereal, 3:eating-shredded-wheat, 4:drinking-water, 5:eating-pizza, 6:pouring-water-in-glass, 7:eating-snack.*

Living room: *8:door-opening, 9:pouring-iced-tea, 10:glass-dropping.*

Bathroom: *15:brushing-teeth, 16:filling-water-into-sink, 17:having-shower, 18:soaping-hands, 19:flushing-toilet, 20:urinating, 21:washing-hands, 22:washing-machine.*

Kitchen: *23:sifting-flour, 24:filling-water, 25:blender, 26:boiling-water, 27:making-coffee, 28:glass-dropping, 29:teapot-whistle, 30:cooker-hood, 31:washing-dishes, 32:dripping-faucet, 33:emptying-water, 34:food-processor, 35:frying-pan, 36:oil-boiling.*

Bedroom: *37:heavy-breathing-sleeping, 38:snoring.*

Other sounds: *1: background-sound, 11:eructation, 12:speaking, 13:yawning, 14:cough, 39: falling-sounds.*

The audio data set was collected in the testbed by playing all 38 sound events corresponding to the human activities at each location multiple times using the sound simulation system. These audio data were recorded by the robot at the sampling rate of 16 kHz. The total time of recording is approxi-

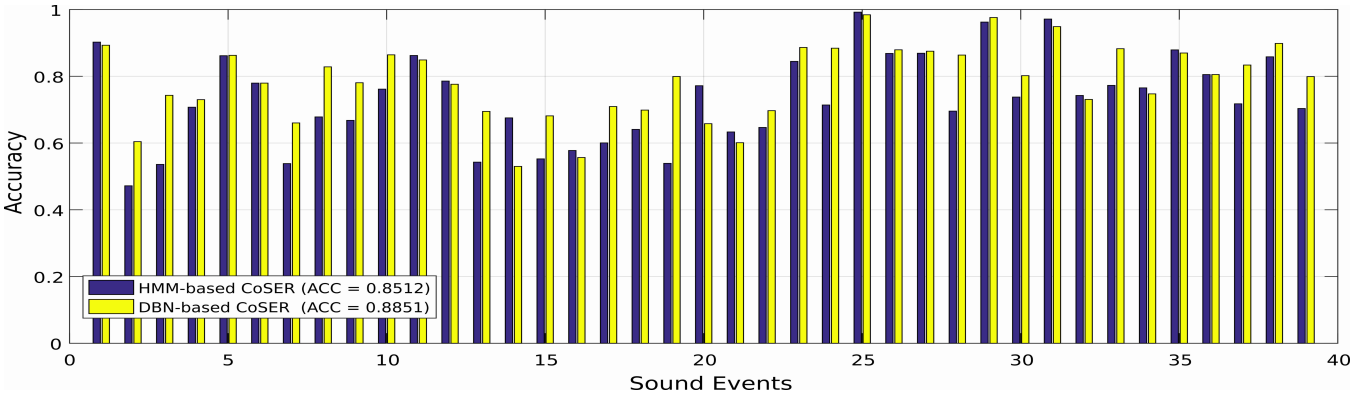


Figure 14: The recognition accuracy of sound events at the frame level of using HMM-based CoSER and DBN-based CoSER.

mately 48 minutes. Those audio files and recorded files were used to train the codebooks with different sizes and window sizes. The window steps were set to a half of the window sizes. The mean square distortion errors of codebooks decrease almost linearly with the numbers of codewords in the codebooks, but increase with the window sizes. However, using the large number of codewords and the small number of window size incurs more computation. Those parameters also affect the accuracy of sound event recognition. Therefore, they were also considered in the experiments to evaluate the CoSER.

We also used the above recorded audio files and context observations provided by the PIR network to train the DBN using 3-fold cross-validation. The overall accuracy of the CoSER with respect to the codebook sizes and the window sizes is shown in Fig. 13. In order to reduce the effect of the codebook's distortion error on the CoSER performance, the codebook size was set to 2048. Therefore, the total numbers of the states S^E, O^E, S^A, and O^A are 39, 2048, 5, and 5 respectively. The overlap between two adjacent sequences is 1. For each sound event, ten different recorded testing data were used. Although larger window size settings have less time-domain resolution, they give better frequency resolution, which is more efficient in classifying sound events that have a wide range of frequencies. The recognition accuracy of sound events at the frame level with a window size of 1024 and a Viterbi sequence size of 10 is shown in Fig. 14. Besides, the evaluation result of each class is shown in Table 7. The CoSER produces the results with accuracy rates more than 80% for more than a half of sound events. Only two sound events (14:cough and 16:filling-sink) have an accuracy rate of below 60%. In addition, the recognition accuracy of DBN-based CoSER performance was compared with that of the HMM-based CoSER that was proposed in [64]. The location context is also provided by the PIR network. HMMs were implemented and trained for each context by 3-fold cross-validation as the DBN model. As shown in Fig. 14, the DBN-based method has better performance in more than two thirds of the sound events and has a higher average accuracy rate than the HMM-based method.

Table 7: Evaluation of the CoSER performance (Total number of tested audio frames: 103707)

Activity	TP	FN	FP	R	Pr	F_1	acc
01-background-sound	25439	142	2904	0.994	0.898	0.944	0.893
02-eating-cereal	1076	256	449	0.808	0.706	0.753	0.604
03-eating-shredded-wheat	1561	396	144	0.798	0.916	0.853	0.743
04-drinking-water	2090	154	619	0.931	0.772	0.844	0.730
05-eating-pizza	4623	328	407	0.934	0.919	0.926	0.863
06-pouring-water-in-glass	626	135	42	0.823	0.937	0.876	0.780
07-eating-snack	2549	870	442	0.746	0.852	0.795	0.660
08-door-opening	1032	214	0	0.828	1.000	0.906	0.828
09-pouring-iced-tea	943	217	48	0.813	0.952	0.877	0.781
10-glass-dropping	484	60	16	0.890	0.968	0.927	0.864
11-eructation	2203	306	86	0.878	0.962	0.918	0.849
12-speaking	8049	627	1694	0.928	0.826	0.874	0.776
13-yawning	1115	186	304	0.857	0.786	0.820	0.695
14-cough	1230	898	193	0.578	0.864	0.693	0.530
15-brushing-teeth	3240	759	756	0.810	0.811	0.811	0.681
16-filling-water-into-sink	1018	576	235	0.639	0.812	0.715	0.557
17-having-shower	6633	1843	875	0.783	0.883	0.830	0.709
18-soaping-hands	1454	300	327	0.829	0.816	0.823	0.699
19-flushing-toilet	4553	583	560	0.886	0.890	0.888	0.799
20-urinating	1181	210	404	0.849	0.745	0.794	0.658
21-washing-hands	1250	605	225	0.674	0.847	0.751	0.601
22-washing-machine	1058	116	344	0.901	0.755	0.821	0.697
23-sifting-flour	2559	312	16	0.891	0.994	0.940	0.886
24-filling-water	1260	155	10	0.890	0.992	0.939	0.884
25-blender	250	4	0	0.984	1.000	0.992	0.984
26-boiling-water	860	20	98	0.977	0.898	0.936	0.879
27-making-coffee	1339	181	10	0.881	0.993	0.933	0.875
28-glass-dropping	2811	368	76	0.884	0.974	0.927	0.864
29-teapot-whistle	733	16	2	0.979	0.997	0.988	0.976
30-cooker-hood	866	214	0	0.802	1.000	0.890	0.802
31-washing-dishes	779	22	20	0.973	0.975	0.974	0.949
32-dripping-faucet	331	10	112	0.971	0.747	0.844	0.731
33-emptying-water	767	6	96	0.992	0.889	0.938	0.883
34-food-processor	414	68	72	0.859	0.852	0.855	0.747
35-frying-pan	414	60	2	0.873	0.995	0.930	0.870
36-oil-boiling	868	202	8	0.811	0.991	0.892	0.805
37-heavy-breathing-sleeping	1143	206	22	0.847	0.981	0.909	0.834
38-snoring	1177	131	2	0.900	0.998	0.947	0.898
39-falling-sounds	1813	160	296	0.919	0.860	0.888	0.799

8.5. Fall Detection and Rescue

Recently, fall detection systems based on ambient sensors have received much interest. Ambient sensors often include infrared sensors, vibration, vision sensors, or acoustic sensors. Table 8 shows the comparison between our project and the recent projects of fall detection and rescue using acoustic sensors, in terms of the deployed sensors, features, algo-

Table 8: The comparison between our project and recent projects of fall detection and rescue

Project	Deployed Sensors	Features	Algorithm	Evaluation	Performance	Rescue
[65]	Accelerometer; and a microphone	MFCCs (sound)	Bayes decision rule classifier	Mimicking doll Rescue Randy, 40 drops. Other objects: 80 drops	Sensibility: 97.5% Specificity: 98.6% (*at segment level*)	Not provided
[66]	A single far-field microphone	Gaussian mixture model supervector	SVM	Audio dataset (7 hours, consisting of 32 sessions that involve 13 different actors as subjects that may fall or perform other activities	F-score: 67% Precision: 71% Recall: 64% *(at the segment level)*	Not provided
[67]	A circular microphone array	MFCCs	k-NN classifier	Using a dataset consisting of 120 falls and 120 non-falls performed by three stunt actors	With an SNR of -2 dB Sensitivity: 81% Recall: 62% Accuracy: 72% (*at the segment level*)	Not provided
[68]	Smartphone microphones	Spectrogram features	Artificial neural network	Nine young volunteers perform fall exercises (fall from sleep, fall from sitting, and fall from standing) and 23 non-fall sound events	Sensitivity: 99% Recall: 98% Accuracy: 98% (*at the segment level*)	Not provided
Our project	A microphone array on the robot	MFCCs and other statistical features	DBN	Multiple types of falling sounds and 37 other sound events	With an SNR of 3 dB Sensitivity:86% Recall:92% Accuracy:80% (*at the frame level*)	Using the robot

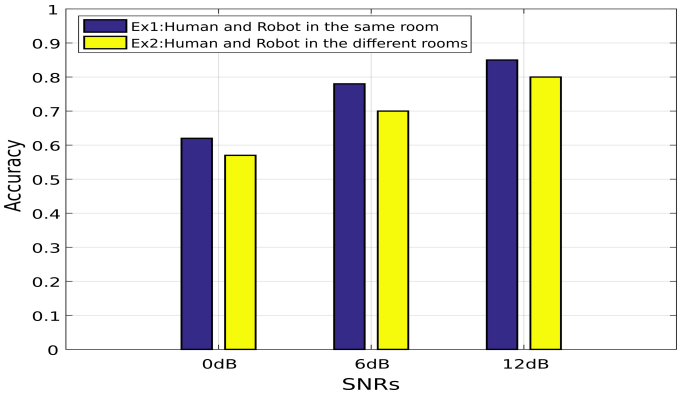


Figure 15: Fall detection performance at the audio frame level.

rithm, evaluation, performance, and rescue methods. The performance of our method is competitive with the methods proposed in [65][66][67][68], although it was evaluated at the audio frame level. The performance at frame level provides better evaluation for the method in terms of its realtimeness. In addition, our method can recognize more types of falling sounds with different settings of environmental noise and even when the robot is unable to observe the resident due to occlusion. Our method can also recognize 37 other sound events that happen in home environments, which other projects did not consider. While these projects did not provide a rescue service, we used the robot to provide the rescue service by connecting to a remote caregiver for assistance.

In addition, we performed experiments in the testbed to evaluate the fall detection and rescue capability of the robot. We tested the fall detection in scenarios when the robot is unable to observe the resident by the camera due to occlusion. The experiments were conducted to evaluate how the falling sounds were detected in such scenarios. Due to safety concerns, the falls were simulated. The falling sounds were collected from the Internet, which were recorded in real home environments. In our experiments, the human subjects mimicked to fall by playing these sounds via a smartphone. The robot recorded these falling sounds at different noise conditions and distances between the human and the robot. In our testbed, the bedroom and the bathroom are closed areas, while the living room, dining room, and kitchen share an open area. We set up two experiments. In the first experiment (Ex1), the human and the robot were in the same room or area. In the second experiment (Ex2), the human and the robot were in different rooms. In both experiments, different fall situations were simulated, such as falling onto carpet, falling onto wooden floor, falling into table, falling into wall, and falling into chair. The falling sounds were recorded by the robot with SNRs at around 0dB, 6dB, and 12dB. The recognition accuracy calculated by Equation (17) is shown in Fig. 15. In both experiments, the robot could detect falling sounds at an accuracy of more than 55% of frames in the noisy environment with an SNR of 0dB even when the audio signal was blocked and more than 80% in the environment with an SNR of 12dB.

Besides, a case study of human activity monitoring, fall detection and rescue was conducted in the testbed. As shown in Fig. 16, a human subject was asked to move around the apartment and conduct daily activities while the sounds were played from the sound simulation system. One IMU motion sensor was used, which was attached to the right thigh of the human subject. The robot was positioned at location (a). The human got up and walked to the bathroom at location *(b)*. He brushed his teeth, soaped his hands, and flushed the toilet. The sounds were captured as shown in the graph *(f1)* and correctly classified as shown in the graph (f2). Then he walked to the kitchen at location *(c)* to fill the water into the teapot and boiled it. After that, he walked to the dining table at location *(d)* and had breakfast with cereal. These activity sounds were also captured and classified. He finished the breakfast and stood up, but fell onto the floor at location *(e)*. The robot captured and detected this falling

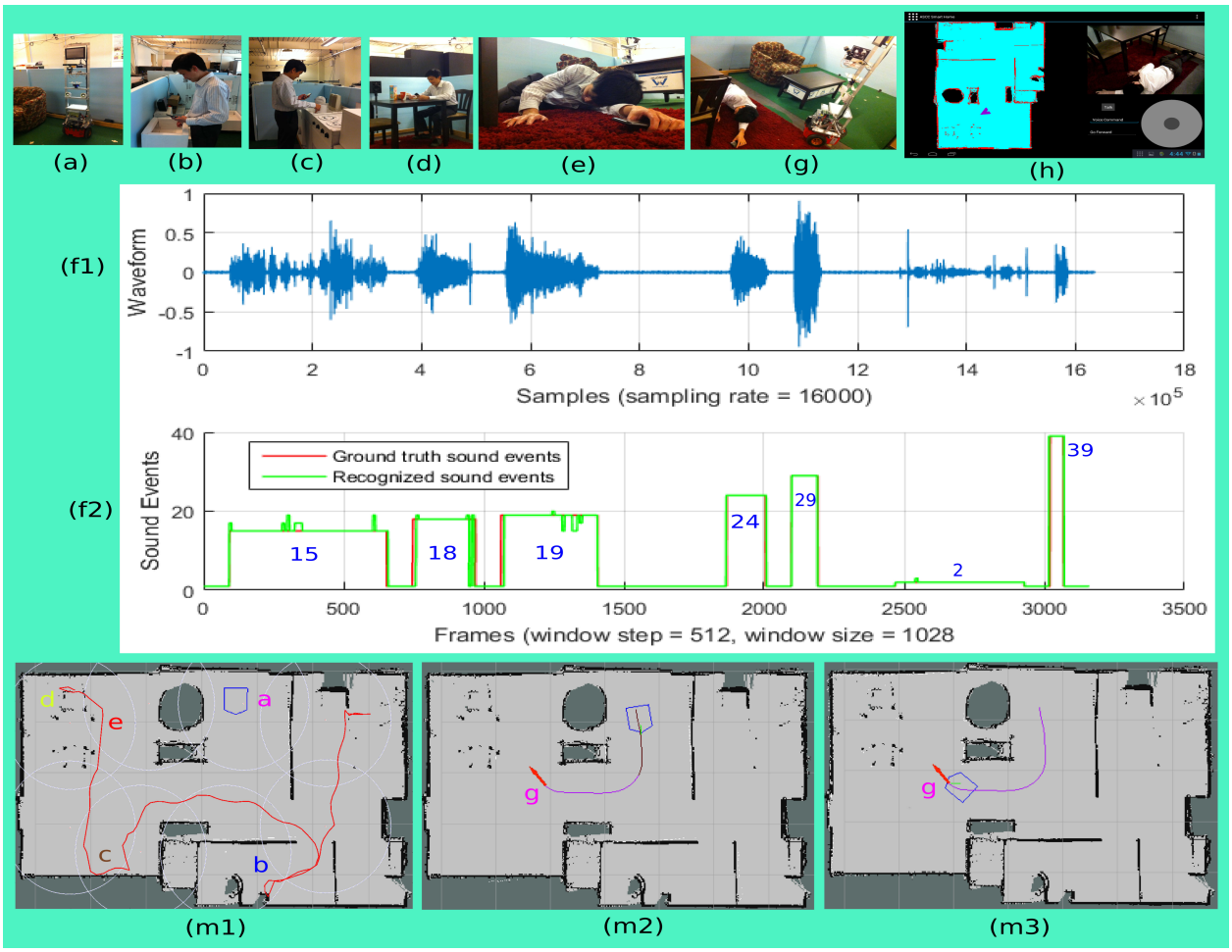


Figure 16: A case study of activity monitoring, fall detection and rescue.

sound. Based on the human position and sound direction, it estimated the goal point (g) and autonomously navigated to *(g)*. An alert message was sent to the RiSH management service on the cloud and then pushed to the caregiver's Android device. The caregiver then used the graphical user interface as shown in Fig. 16-(h) to control the robot to help the older adult.

9. Conclusions and Future Works

In this work, we proposed a robot-integrated smart home testbed for elderly care and developed a home service robot platform. A layered architecture was proposed for the development of the software platform. Several key services and three applications of the RiSH were implemented to show the operation of the various components in the RiSH and the system as a whole. The home service robot is equipped with the following auditory perception capabilities: sound localization with an error of less than 2^0 and voice/non-voice recognition with an accuracy of 98% for the whole sounds separated by sound separation. Through sensor fusion, the RiSH is capable of human body activity recognition with an accuracy of more than 86%, human location tracking with a root mean square error of less than 0.2 m, and sound event recognition with an average accuracy of 88%. The RiSH can detect falling sounds with an accuracy of 80% at the frame level even when the robot is unable to observe the resident by the camera due to occlusion. However, the fall detection needs to be improved by fusing multiple sensors in the RiSH to detect falls that do not generate sound. Overall, we proposed a comprehensive framework to implement the RiSH that has the potential to be used to conduct research in assistive technologies for elderly care. The RiSH and its applications of human activity monitoring, fall detection and rescue still need to be tested with elderly subjects in real environments. The future work will focus on multi-sensor fall detection, friendly human-robot interface, social intelligence, and other elderly care services for the home service robot.

Acknowledgements

This work was supported by the National Science Foundation (NSF) grant CISE/IIS 1231671/IIS 1427345, the Open Research Project of the State Key Laboratory of Industrial Control Technology ICT170314, Zhejiang University, China, the Basic Public Research Program of Zhejiang Province (No. LGF18F030001) andthe Shenzhen Overseas High Level Talent (Peacock Plan) Program (No. KQTD20140630154026047).

References

[1] WHO, World Health Organization: 10 facts on ageing and the life course. URL Website:http://www.who.int/features/factfiles/ageing/en/

[2] J. Secker, R. Hill, L. Villeneau, S. Parkman, Promoting independence: but promoting what and how?, Ageing and Society 23 (03) (2003) 375–391.

[3] Sony aibo. URL http://www.sony-aibo.com/

[4] Care-o-bot-4. URL http://www.care-o-bot-4.de/

[5] Paro therapeutic robot. URL http://www.parorobots.com/

[6] T. Breuer, G. R. Giorgana Macedo, R. Hartanto, N. Hochgeschwender, D. Holz, F. Hegger, Z. Jin, C. Müller, J. Paulus, M. Reckhaus, J. A. Álvarez Ruiz, P. G. Plöger, G. K. Kraetzschmar, Johnny: An Autonomous Service Robot for Domestic Environments, Journal of Intelligent & Robotic Systems 66 (1-2) (2011) 245–272. doi:10.1007/s10846-011-9608-y.

[7] H. Gross, C. Schroeter, S. Mueller, M. Volkhardt, E. Einhorn, A. Bley, C. Martin, T. Langner, M. Merten, Progress in Developing a Socially Assistive Mobile Home Robot Companion for the Elderly with Mild Cognitive Impairment, in: Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on, 2011, pp. 2430–2437.

[8] K. Yamazaki, R. Ueda, S. Nozawa, M. Kojima, K. Okada, K. Matsumoto, M. Ishikawa, I. Shimoyama, M. Inaba, Home-assistant robot for an aging society, Proceedings of the IEEE 100 (8) (2012) 2429–2441.

[9] D. Fischinger, P. Einramhof, K. Papoutsakis, W. Wohlkinger, P. Mayer, P. Panek, S. Hofmann, T. Koertner, A. Weiss, A. Argyros, et al., Hobbit, a care robot supporting independent living at home: First prototype and lessons learned, Robotics and Autonomous Systems 75 (2014) 60–78.

[10] G. D. Abowd, A. F. Bobick, I. A. Essa, E. D. Mynatt, W. A. Rogers, The aware home: A living laboratory for technologies for successful aging, in: Proceedings of the AAAI-02 Workshop "Automation as Caregiver", 2002, pp. 1–7.

[11] Gato-tech smart house. URL http://www.icta.ufl.edu/gatortech/

[12] M. Mozer, R. Dodier, D. Miller, M. Anderson, J. Anderson, D. Bertini, M. Bronder, M. Colagrosso, R. Cruickshank, B. Daugherty, et al., The adaptive house, in: IEE Seminar Digests, Vol. 11059, IET, 2005, pp. v1–39.

[13] P. Dawadi, D. J. Cook, M. Schmitter-Edgecombe, Smart home-based longitudinal functional assessment, in: Proceedings of the 2014 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct Publication, ACM, 2014, pp. 1217–1224.

[14] R. Orpwood, C. Gibbs, T. Adlam, R. Faulkner, D. Meegahawatte, The gloucester smart house for people with dementiauser-interface aspects, in: Designing a More Inclusive World, Springer, 2004, pp. 237–245.

[15] D. J. Cook, G. M. Youngblood, E. O. Heierman III, K. Gopalratnam, S. Rao, A. Litvin, F. Khawaja, Mavhome: An agent-based smart home., in: PerCom, Vol. 3, 2003, pp. 521–524.

[16] P. N. Dawadi, D. J. Cook, M. Schmitter-Edgecombe, Automated cognitive health assessment using smart home monitoring of complex tasks, Systems, Man, and Cybernetics: Systems, IEEE Transactions on 43 (6) (2013) 1302–1313.

[17] N. K. Suryadevara, S. C. Mukhopadhyay, Wireless sensor network based home monitoring system for wellness determination of elderly, IEEE Sensors Journal 12 (6) (2012) 1965–1972.

[18] Â. Costa, J. C. Castillo, P. Novais, A. Fernández-Caballero, R. Simoes, Sensor-driven agenda for intelligent home care of the elderly, Expert Systems with Applications 39 (15) (2012) 12192–12204.

[19] A. Costa, P. Novais, R. Simoes, A caregiver support platform within the scope of an ambient assisted living ecosystem, Sensors 14 (3) (2014) 5654–5676.

[20] P. R. Liu, M. Q.-H. Meng, P. X. Liu, F. F. Tong, X. Chen, A telemedicine system for remote health and activity monitoring for the elderly, Telemedicine Journal & e-Health 12 (6) (2006) 622–631.

[21] M. Broxvall, M. Gritti, A. Saffiotti, B.-S. Seo, Y.-J. Cho, Peis ecology: Integrating robots into smart environments, in: Robotics and Automation (ICRA), 2006 IEEE International Conference on, IEEE, 2006, pp. 212–218.

[22] R. Borja, J. De La Pinta, A. Álvarez, J. M. Maestre, Integration of service robots in the smart home by means of upnp: A surveillance robot case study, Robotics and Autonomous Systems 61 (2) (2013) 153–160.

[23] M. Chen, Y. Ma, S. Ullah, W. Cai, E. Song, ROCHAS: robotics and cloud-assisted healthcare system for empty nester, in: Proceedings of the 8th International Conference on Body Area Networks, 2013.

[24] D. O. Johnson, R. H. Cuijpers, J. F. Juola, E. Torta, M. Simonov, A. Frisiello, M. Bazzani, W. Yan, C. Weber, S. Wermter, et al., Socially assistive robots: a comprehensive approach to extending independent living, International journal of social robotics 6 (2) (2014) 195–211.

[25] E. Torta, F. Werner, D. O. Johnson, J. F. Juola, R. H. Cuijpers, M. Bazzani, J. Oberzaucher, J. Lemberger, H. Lewy, J. Bregman, Evaluation of a small socially-assistive humanoid robot in intelligent homes for the care of the elderly, Journal of Intelligent & Robotic Systems 76 (1) (2014) 57.

[26] E. D. Mynatt, I. Essa, W. Rogers, Increasing the opportunities for aging in place, in: Proceedings on the 2000 conference on Universal Usability, ACM, 2000, pp. 65–71.

[27] H. Lehmann, D. Syrdal, K. Dautenhahn, G. J. Gelderblom, S. Bedaf, What Should a Robot do for you ? - Evaluating the Needs of the Elderly in the UK, Interactions (c) (2013) 83–88.

[28] Hokuyo laser. URL http://www.hokuyo-aut.jp/

[29] ASUS, Xtion pro live. URL https://www.asus.com/us/3D-Sensor/

[30] Sony, Ps3 eye camera. URL http://www.sony.co.in/product/playstation+eye

[31] C. Hacks, e-health sensor platform for arduino and raspberry pi. URL http://www.cooking-hacks.com

[32] The vn-100 rugged imu. URL http://www.vectornav.com/products/vn100-rugged

[33] Infrared array sensor grid-eye. URL http://industrial.panasonic.com/ww/products/sensors/built-in-sensors/grid-eye

[34] Open source software for creating private and public clouds. URL https://www.openstack.org/

[35] O. Khedher, OpenStack Sahara Essentials, Packt Publishing Ltd, 2016.

[36] Motion capture systems - optitrack. URL https://www.naturalpoint.com/optitrack/

[37] H. M. Do, W. Sheng, M. Liu, Human-assisted sound event recognition for home service robots, Robotics and Biomimetics 3 (1) (2016) 7.

[38] Beagleboard-xm. URL http://beagleboard.org/beagleboard-xm

[39] T. Reichherzer, S. Satterfield, J. Belitsos, J. Chudzynski, L. Watson, An agent-based architecture for sensor data collection and reasoning in smart home environments for independent living, in: Canadian Conference on Artificial Intelligence, Springer, 2016, pp. 15–20.

[40] J. Nehmer, M. Becker, A. Karshmer, R. Lamm, Living assistance systems: an ambient intelligence approach, in: Proceedings of the 28th international conference on Software engineering, ACM, 2006, pp. 43–50.

[41] C. Roda, A. Rodríguez, V. López-Jaquero, P. González, E. Navarro, A multi-agent system in ambient intelligence for the physical rehabilitation of older people, in: Trends in Practical Applications of Agents, Multi-Agent Systems and Sustainability, Springer, 2015, pp. 113–123.

[42] J. M. Fernández, R. Fuentes-Fernández, J. Pavón, A dynamic context-aware architecture for ambient intelligence, in: International Work-Conference on Artificial Neural Networks, Springer, 2011, pp. 637–644.

[43] Ros wiki. URL http://www.ros.org/wiki/

[44] K. Nakadai, T. Takahashi, H. G. Okuno, H. Nakajima, Y. Hasegawa, H. Tsujino, Design and implementation of robot audition system'hark'open source software for listening to three simultaneous speakers, Advanced Robotics 24 (5-6) (2010) 739–761.

[45] W. B. G. Grisetti, C. Stachniss, Improved Techniques for Grid Mapping With Rao-Blackwellized Particle Filters, Robotics, IEEE Transactions on 23 (1) (2007) 34–46.

[46] D. Fox, Adapting the Sample Size in Particle Filters Through KLD- Sampling, International Journal of Robotics Research 22.

[47] K. Nakamura, K. Nakadai, F. Asano, Y. Hasegawa, H. Tsujino, Intelligent sound source localization for dynamic environments, in: Intelligent

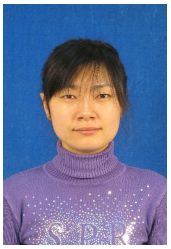
Robots and Systems (IROS), 2009 IEEE/RSJ International Conference on, IEEE, 2009, pp. 664–669.
[48] K. Nakadai, G. Ince, K. Nakamura, H. Nakajima, Robot audition for dynamic environments, in: Signal Processing, Communication and Computing (ICSPCC), 2012 IEEE International Conference on, IEEE, 2012, pp. 125–130.
[49] K.-M. Kim, S.-Y. Kim, J.-K. Jeon, K.-S. Park, Quick audio retrieval using multiple feature vectors, Consumer Electronics, IEEE Transactions on 52 (1) (2006) 200–205.
[50] S. Oh, L. Schenato, P. Chen, S. Sastry, Tracking and coordination of multiple agents using sensor networks: system design, algorithms and experiments, Proceedings of the IEEE 95 (1) (2007) 234–254.
[51] J. H. Friedman, Stochastic gradient boosting, Computational Statistics & Data Analysis 38 (4) (2002) 367–378.
[52] M. Pham, D. Yang, W. Sheng, M. Liu, Human localization and tracking using distributed motion sensors and an inertial measurement unit, in: 2015 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2015, pp. 2127–2132.
[53] M. Pham, Y. Mengistu, H. Do, W. Sheng, Delivering home healthcare through a cloud-based smart home environment (coshe), Future Generation Computer Systems.
[54] E. Foxlin, Pedestrian tracking with shoe-mounted inertial sensors, IEEE Computer graphics and applications 25 (6) (2005) 38–46.
[55] M. Bar, The proactive brain: using analogies and associations to generate predictions, Trends in cognitive sciences 11 (7) (2007) 280–289.
[56] Y. Linde, A. Buzo, R. M. Gray, An algorithm for vector quantizer design, Communications, IEEE Transactions on 28 (1) (1980) 84–95.
[57] K. P. Murphy, Dynamic bayesian networks: representation, inference and learning, in: Ph.D thesis at the University of California, Berkeley, 2002.
[58] J. Bloit, X. Rodet, Short-time viterbi for online hmm decoding: Evaluation on a real-time phone recognition task, in: Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on, 2008, pp. 2121–2124.
[59] H. M. Do, W. Sheng, M. Liu, S. Zhang, Context-aware sound event recognition for home service robots, in: Automation Science and Engineering (CASE), 2016 IEEE International Conference on, IEEE, 2016, pp. 739–744.
[60] G. Fortino, R. Giannantonio, R. Gravina, P. Kuryloski, R. Jafari, Enabling effective programming and flexible management of efficient body sensor network applications, IEEE Transactions on Human-Machine Systems 43 (1) (2013) 115–133.
[61] A. Ahmadi, E. Mitchell, F. Destelle, M. Gowing, N. E. O'Connor, C. Richter, K. Moran, Automatic activity classification and movement assessment during a sports training session using wearable inertial sensors, in: Wearable and Implantable Body Sensor Networks (BSN), 2014 11th International Conference on, IEEE, 2014, pp. 98–103.
[62] C. M. el Achkar, C. Lenoble-Hoskovec, A. Paraschiv-Ionescu, K. Major, C. Büla, K. Aminian, Instrumented shoes for activity classification in the elderly, Gait & posture 44 (2016) 12–17.
[63] N. Abhayasinghe, I. Murray, Human activity recognition using thigh angle derived from single thigh mounted imu data, in: Indoor Positioning and Indoor Navigation (IPIN), 2014 International Conference on, IEEE, 2014, pp. 111–115.
[64] T. Heittola, A. Mesaros, A. Eronen, T. Virtanen, Context-dependent sound event detection, EURASIP Journal on Audio, Speech, and Music Processing 2013 (1) (2013) 1–13.
[65] Y. Zigel, D. Litvak, I. Gannot, A method for automatic fall detection of elderly people using floor vibrations and soundproof of concept on human mimicking doll falls, IEEE Transactions on Biomedical Engineering 56 (12) (2009) 2858–2867.
[66] X. Zhuang, J. Huang, G. Potamianos, M. Hasegawa-Johnson, Acoustic fall detection using gaussian mixture models and gmm supervectors, in: Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on, IEEE, 2009, pp. 69–72.
[67] Y. Li, K. Ho, M. Popescu, A microphone array system for automatic fall detection, IEEE Transactions on Biomedical Engineering 59 (5) (2012) 1291–1301.
[68] M. Cheffena, Fall detection using smartphone audio features, IEEE journal of biomedical and health informatics 20 (4) (2016) 1073–1080.

Authors

Ha Manh Do received his B.Sc. degree in Electronics and Telecommunications from Hanoi University of Technology and Science, Vietnam in 1999. He earned his M.S. degree in 2015 and is currently a Ph.D candidate in Electrical and Computer Engineering at Oklahoma State University. His research interests include home service robots and smart homes for elderly care, auditory perception, natural language understanding, and deep learning.

Minh Pham received the B.Sc. degree in Computer Science from Hanoi University of Science and Technology, Vietnam in 2007. He is currently pursuing his Ph.D. degree in Electrical and Computer Engineering at Oklahoma State University. His research interests include smart homes and wearable computing.

Weihua Sheng is the Director of the Laboratory for Advanced Sensing, Computation and Control at Oklahoma State University. He received his Ph.D degree in Electrical and Computer Engineering from Michigan State University in May 2002. He obtained his M.S and B.S. degrees in Electrical Engineering from Zhejiang University, China in 1997 and 1994, respectively. He is the author of more than 170 peer-reviewed papers in major journals and international conferences. His research interests include mobile robotics, wearable computing, human robot interaction and intelligent transportation systems. He serves as an associate editor for IEEE Transactions on Automation Science and Engineering.



Dan Yang received the B.S. and M.S. degrees in biomedical engineering from Northeastern University(NEU), Shenyang, China, in 2002 and 2005, respectively, and the Ph. D. degree in Detection and automatic control engineering in NEU,2009. From 2009 to 2012, she was a Postdoctoral researcher with the Computer and Science Engineering in NEU. From 2013 July to 2014 July, she was a visiting scholar sponsored by China Scholarship Council, with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater. She is a lecturer with the Department of Intelligent Perception and Electronics Engineering in NEU. She has

been actively involved in research projects with the Fundamental Research Funds for the Central Universities of China, and the National Natural Science Foundation of China. Her current research includes wearable computing, physiological signal detection, and biological electromagnetic systems.

Meiqin Liu received the B.E. and Ph.D. degrees in control theory and control engineering from Central South University, Changsha, China, in 1994 and 1999, respectively. She was a post-doctoral research fellow with the Huazhong University of Science and Technology, Wuhan, China, from 1999 to 2001. She was a visiting scholar with the University of New Orleans, New Orleans, LA, USA, from 2008 to 2009. She is currently a professor with the College of Electrical Engineering, Zhejiang University, Hangzhou, China. She has authored more than 60 peer reviewed papers, including 33 journal papers. Her current research interests include neural network, robust control, multi-sensor network, and information fusion.