Supervised Learning Based Multi-modal Perception for Robot Partners using Smart Phones

Dalai Tang, Janos Botzheim, Naoyuki Kubota

Graduate School of System Design, Tokyo Metropolitan University

6-6 Asahigaoka, Hino, Tokyo, 191-0065 Japan

tang@ed.tmu.ac.jp, {botzheim, kubota}@tmu.ac.jp

Abstract: This paper proposes a method for multi-modal perception of human-friendly robot partners based on various types of sensors built in a smart phone. The proposed method can estimate human interaction modes by fuzzy spiking neural network. The learning method of the spiking neural network based on the time series of the measured data is explained as well. Evolution strategy is used for optimizing the parameters of the fuzzy spiking neural network. Several experimental results are presented for confirming the effectiveness of the proposed technique. Finally, the future direction on this research is discussed.

Keywords: Robot Partners; Multi-modal Perception; Computational Intelligence; Human Robot Interaction

1 Introduction

Recently, the rate of elderly people rises in the super-aging society in many countries. For example, the rate is estimated to reach 23.8% in Tokyo in 2015 [1]. In general, the mental and physical care is very important for elderly people living home alone. Such elderly people have little chances to talk to other people and to perform daily physical activity. Human-friendly robots can be used as partners in daily communication to support the care of elderlies. Furthermore, various types of human-friendly robots such as pet robots, amusement robots, and robot partners have been developed to communicate with people [2-4]. However, it is difficult for a robot to converse with a person appropriately even if many contents of the conversation are designed beforehand, because the robot must have personal information and life logs required for daily conversation. Furthermore, in addition to verbal communication, the robot should understand non-verbal communication, e.g., facial expressions, emotional gestures, and pointing gestures. We have used various types of robot partners such as MOBiMac, Hubot, Apri Poco, palro, miuro, and other robots for information support to elderly people, rehabilitation support, and robot edutainment [5-9]. In order to popularize robot partners, the price of a robot partner should be as low as possible. Therefore, we have also been developing on-table small sized robot partners called iPhonoid and iPadrone [10,11]. In this paper, we focus on how to use sensors that a smart phone or a tablet PC equipped with.

Various types of concepts and technologies on ubiquitous computing, sensor networks, ambient intelligence, disappearing computing, intelligent spaces, and other fields have been proposed and developed to realize information gathering, life support, safe and secure society [12-17]. One of the most important issues in the concepts and technologies is the structuralization of information. The structuralization of information means to give qualitative meaning to data and quantitative information in order to improve the accessibility and usability of information. We can obtain huge size of data through sensor networks, however useful, meaningful and valuable information should be extracted from such data.

We have proposed the concept of informationally structured space to realize the quick update and access of valuable and useful information for people and robots [18,19]. The sensing range of both people and robot is limited. If the robot can obtain the exact position of the robot itself, people, and objects in an environment, the robot does not need any sensors for measuring such information. As a result, the weight and size of the robot can be reduced, since many sensors can be removed from the robot. The received environmental information is more precise because the sensors equipped in the environment are designed suitable to the environmental conditions. Furthermore, if the robot can share the environmental information with people, the communication with people might become very smooth and natural. Therefore, we have proposed methods for human localization [20,21], map building [22], and 3D visualization [23]. Various types of estimation method of human state have been proposed. We applied spiking neural network [24,25] for localizing human position, estimating human transportation mode [26], and learning pattern of daily life of elderly people [27].

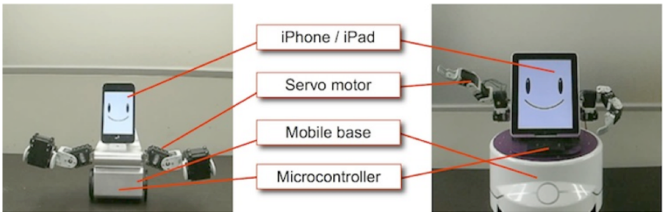
In this paper the sensors of a smart phone is used for estimating human interaction modes. As the computational power of a smart phone is not so high compared to a standard PC, we should reduce the computational cost as much as possible. Computational intelligence techniques are able to find good compromise between computational cost and solution accuracy. In this paper fuzzy spiking neural network is proposed for estimating human interaction modes. Additionally, evolution strategy is used for optimizing the parameters of the fuzzy spiking neural network. The performance of estimation is analyzed by several experimental results.

This paper is organized as follows. Section II explains the hardware specification of robot partners applied in this study, the interaction modes, and the sensory inputs from a smart phone. Section III proposes the method of estimating human interaction modes. Section IV shows several experimental results. Finally, Section V summarizes this paper, and discusses the future direction to realize human-friendly robot partners.

2 Robot Partners using Smart Phones

2.1 Robot Partners

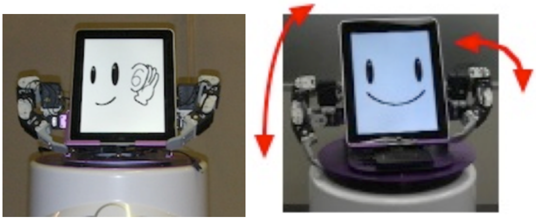
Recently, various types of smart phone and tablet PC have been developed, and their price is decreasing year by year [28]. Furthermore, the embedded system technology enables to miniaturize such a device and to integrate it with many sensors and other equipments. As a result, we can get a mechatronics device including many sensors, wireless communication systems, GPU and CPU composed of multiple cores with low price. Furthermore, elderly people unfamiliar with information home appliances have started using tablet PC [29], because touch panels and touch interfaces have been popularized at ticket machines and information services in public areas. Therefore, we started the development project on on-table small sized human-friendly robot partners called iPhonoid and iPadrone based on smart phone or tablet PC to realize information support for elderly people (Figs.1 (a) and (b)). Since iPhone is equipped with various sensors such as gyro, accelerometer, illumination sensor, touch interface, compass, two cameras, and microphone, the robot itself is enough to be equipped with only cheap range sensors. The mobile base is equipped on the bottom, however the mobile base is not used on the table for safety's sake. In order to control the actuators of a robot partner from the smart phone or tablet PC, wireless LAN and wireless PAN (Bluetooth) can be used in addition to a wired serial communication.



(a) iPhonoid (b) iPadrone

Figure 1

Robot partners using a smart phone and a tablet PC



(a) Voice recognition (b) A gesture

Figure 2
Robot behaviors for social communication with people

Basically, human detection, object detection, and voice recognition are performed by smart phone or tablet PC. Furthermore, touch interface is used as a direct communication method. The robot partner starts the multi-modal interaction after a smart phone is attached to the robot base. We use touch interface on the smart phone or tablet PC as the nearest interaction with the robot partner. The facial parts are displayed as icons for the touch interface on the display (Fig.2). Since the aim of this study is to realize information support for elderly people, the robot partner provides elderly people with their required information through the touch interface. The eye icon and mouth icon are used for providing the visual information and text information, respectively.

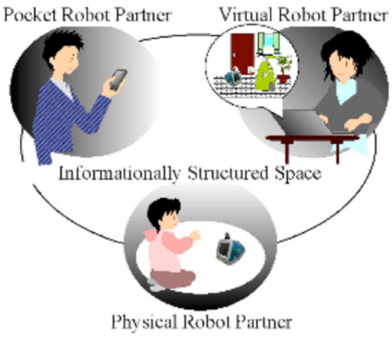


Figure 3
Interaction with robot partners from different view points

The ear icon is used for direct voice recognition because it is difficult to perform high performance of voice recognition in the daily communication with the robot partner. If the person touches the mouth icon, then the ear icon appears, and the voice recognition starts. The voice recognition is done by Nuance Mobile Developer Program (NMDP). NMDP is a self-service program for developers of iOS and Android application [30]. In this way, the total performance of multi-modal communication can be improved by combining several communication modalities of touch interface, voice recognition, and image processing. The conversation system is composed of (A) daily conversation mode, (B) information support mode, and (C) scenario conversation mode [5-9].

2.2 Interaction Modes

We can discuss three different types of robot partners using a smart phone or a tablet PC from the interactive point of view: a physical robot partner, a pocket robot partner, and a virtual robot partner (Fig.3). These modes are not independent, and we interact with the robot partner based on several modes. Interaction modes mean the ways how we interact with the robot partner. We can interact with a physical robot partner by using multi-modal communication like with a human. The interaction is symmetric. The other one is a virtual robot partner. The virtual robot partner exists in the virtual space in the computer and can be considered as a computer agent, but we can interact with it through the virtual person or robot by immersing him or her in the virtual space. Therefore, the interaction is symmetric. The pocket robot partner has no mobile mechanism, but we can easily bring it everywhere and we can interact with the robot partner by touch and physical interface. Furthermore, the pocket robot partner can estimate the human situation by using internal sensors such as illumination sensor, digital compass, gyro, and accelerometer. The advantage of this device is in the compactness of integrated multi-modal communication interfaces in a single device.

Each style of robot partners is different, but the interaction modes depend on each other, and we interact with the robot partner with the same knowledge on personal information, life logs, and interaction rules.

In this paper, since we use the facial expression on the display for human interaction (see Figs.1 and 2), the robot partner should estimate the human interaction mode: (a) the physical robot partner mode (attached on the robot base), (b) the pocket robot partner mode (having removed from the robot base), or (c) other mode (on the table, in the bag, or others). In this paper 7 interaction modes are defined which will be detailed in the experimental section and depicted in Figs. 9 and 10.

2.3 Sensory Inputs from Smart Phones

We can use several sensory data measured by a smart phone. As depicted in Fig. 4, since iOS 4.0 there has been a Core Motion framework to deal with obtaining sensory data. The acceleration of human movement is calculated by using a high-pass filter for the measured data by the accelerometer. The angular velocity is calculated by using a low-pass filter for the measured data by the gyro sensor. The iPhone’s attitude is calculated by the measured data of accelerometer, gyroscope, and magnetometer. The specification of the measured data is presented in Table 1. The data are updated in every 100 ms.

Table 1
Specification of iPhone's sensors

Sensor Name	Accelerometer	Gyro	Attitude
Acquired data	a_x, a_y, a_z	ω_x, ω_y, ω_z	pitch, roll, yaw
Range of data	-/+2.3G	-/+90, -/+180, -/+180,	-/+90, -/+180, -/+180,
Time interval	100ms	100ms	100ms

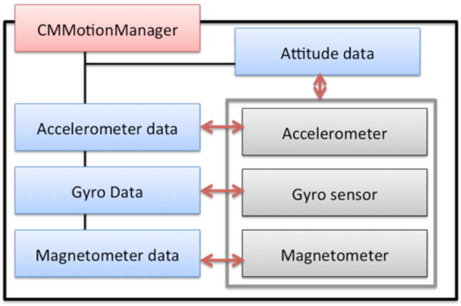


Figure 4
iOS Core Motion Framework for obtaining sensory data

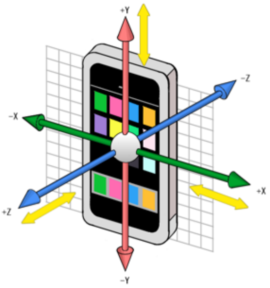


Figure 5
Acceleration data of iPhone

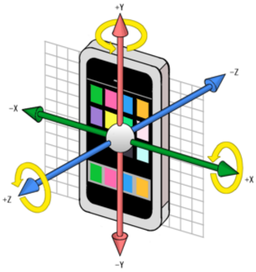


Figure 6
Angular velocity data of iPhone

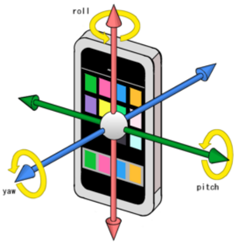


Figure 7
iPhone's attitude data

The acceleration can be calculated as (Fig. 5):

$$a(t) = \sqrt{a_x(t)^2 + a_y(t)^2 + a_z(t)^2}, \tag{1}$$

where $a_x(t)$, $a_y(t)$, and $a_z(t)$ are the components of the acceleration in the unit directions at time t.

The angular velocity is computed as (Fig. 6):

$$\omega(t) = \sqrt{\omega_x(t)^2 + \omega_y(t)^2 + \omega_z(t)^2}, \tag{2}$$

where $\omega_x(t)$, $\omega_y(t)$, and $\omega_z(t)$ are the angular velocities at time t in the X-axis, Y-axis, and Z-axis, respectively.

The iPhone's attitude is (Fig. 7):

$$\theta_x(t) = \left|\frac{\theta_x(t)}{90^\circ}\right|, \theta_y(t) = \left|\frac{\theta_y(t)}{180^\circ}\right|, \theta_z(t) = \left|\frac{\theta_z(t)}{180^\circ}\right|, \tag{3}$$

where $\theta_x(t)$, $\theta_y(t)$, and $\theta_z(t)$ are the pitch, roll, and yaw Euler angles at time t, respectively.

Since the measured data includes noise, we have to use some smoothing functions. Although there are many methods for decreasing the noise, in our application the computation complexity is an important issue. We realize our system in a smart phone which has limited computational power compared to a PC. Therefore, two simple weighted moving averages are applied as presented in Eqs. (4) and (5), where d is the window length. In Eq. (4) the weights increase from the smallest weight at time (t-d+1) to the current data point at time t. In Eq. (5) the weights increase first, from the smallest weight at time (t-$d/2$) to the current point at time t, after that they decrease till time (t+$d/2$-1). In Eqs. (4) and (5) j indicates the input.

$$\tilde{x}_j(t) = \frac{1}{\sum_{k=0}^{d-1} \exp\left(-\frac{k}{d}\right)} \sum_{k=0}^{d-1} \exp\left(-\frac{k}{d}\right) x_j(t-k) \tag{4}$$

$$\tilde{x}_j(t) = \frac{1}{\sum_{k=0}^{d-1} \exp\left(-\frac{\left|k-\frac{d}{2}\right|}{d}\right)} \sum_{k=0}^{d-1} \exp\left(-\frac{\left|k-\frac{d}{2}\right|}{d}\right) \cdot x_j\left(t+k-\frac{d}{2}\right) \tag{5}$$

3 Fuzzy Spiking Neural Network for Estimation of Human Interaction Modes

3.1 Fuzzy Spiking Neural Network

We estimate the human interaction modes by fuzzy spiking neurons. One important feature of spiking neurons is the capability of temporal coding. In fact, various types of spiking neural networks (SNNs) have been applied for memorizing spatial and temporal context [31-33]. A simple spike response model is used in order to reduce the computational cost. In our model the SSN has fuzzy inputs, it is a fuzzy spiking neural network (FSNN) [25-27]. Figure 8 illustrates the FSNN model. We use evolution strategy to adapt the parameters of the fuzzy membership functions applied as inputs to the spiking neural network. Figure 9 depicts the detailed structure of the FSNN model. The inputs of the FSNN are the sensory data, the outputs are the interaction modes. We use 7 interaction modes as presented in Fig. 9.

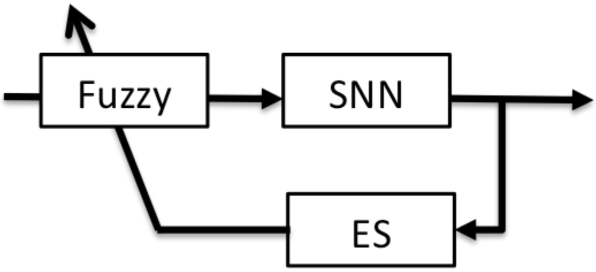


Figure 8
Fuzzy spiking neural network

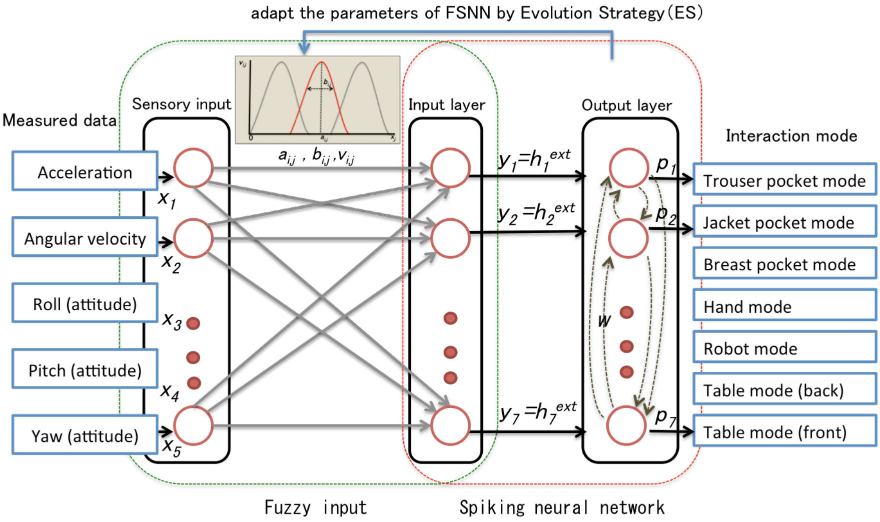


Figure 9
Detailed structure of fuzzy spiking neural network

On the sensory input fuzzy inference is performed by:

$$\mu_{A_{i,j}}(x_j) = \exp\left(-\frac{(x_j - a_{i,j})^2}{b_{i,j}}\right) \quad (6)$$

$$y_i = \prod_{j=1}^{m} v_{i,j} \cdot \mu_{A_{i,j}}(x_j) \quad (7)$$

where $a_{i,j}$ and $b_{i,j}$ are the central value and the width of the membership function $A_{i,j}$ and $v_{i,j}$ is the contribution of the j-th input to the estimation of the i-th human interaction mode. The result of fuzzy inference, y_i, will be the input of the spiking neurons.

The membrane potential, or internal state $h_i(t)$ of the i-th spiking neuron at the discrete time t is given by:

$$h_i(t) = \tanh(h_i^{syn}(t) + h_i^{ext}(t) + h_i^{ref}(t)), \tag{8}$$

where $h_i^{syn}(t)$ includes the pulse outputs from the other neurons, $h_i^{ref}(t)$ is used for representing the refractoriness of the neuron, $h_i^{ext}(t)$ is the input to the i-th neuron from the external environment. The hyperbolic tangent function is used to avoid the bursting of neuronal fires.

The external input, $h_i^{ext}(t)$ is calculated based on the fuzzy inference in Eqs. (6) and (7), and it is equal to y_i as illustrated in Fig. 9, thus:

$$h_i^{ext}(t) = \prod_{j=1}^{M} v_{i,j} \cdot \exp\left(-\frac{(x_j - a_{i,j})^2}{b_{i,j}} \right) \tag{9}$$

Furthermore, $h_i^{syn}(t)$ indicates the output pulses from other neurons presented by dashed arrows in Fig. 9 in the output layer:

$$h_i^{syn}(t) = \sum_{j=1, j\neq i}^{N} w_{j,i} \cdot h_j^{PSP}(t-1), \tag{10}$$

where $w_{j,i}$ is a weight coefficient from the j-th to the i-th neuron; $h_j^{PSP}(t)$ is the presynaptic action potential (PSP) approximately transmitted from the j-th neuron at the discrete time t; N is the number of neurons. When the internal state of the i-th neuron is larger than the predefined threshold, a pulse is outputted as follows:

$$p_i(t) = \begin{cases} 1 & if\ h_i(t) \geq q^{pul} \\ 0 & otherwise \end{cases} \tag{11}$$

where q^{pul} is a threshold for firing. The outputs of FSNN are the p_i values as presented in Fig. 9. Thus, the output is the interaction mode i which for p_i=1. If there are more than one neuron with output pulse 1 (i.e., $\exists i, j\ i \neq j\ p_i = p_j = 1$), then the output will be that one which fired in the previous step, t–1. If none fired, then the output neuron will be selected randomly.

Furthermore, R is subtracted from the refractoriness value as follows:

$$h_i^{ref}(t) = \begin{cases} \gamma^{ref} \cdot h_i^{ref}(t-1) - R & if\ p_i(t-1) = 1 \\ \gamma^{ref} \cdot h_i^{ref}(t-1) & otherwise \end{cases} \tag{12}$$

where γ^{ref} is a discount rate and R>0.

The spiking neurons are interconnected, and the presynaptic spike output is transmitted to the connected neuron according to the PSP with the weight connection. The PSP is calculated as follows:

$$h_i^{PSP}(t) = \begin{cases} 1 & \mathit{if}\ p_i(t) = 1 \\ \gamma^{PSP} \cdot h_i^{PSP}(t-1) & \mathit{otherwise} \end{cases} \tag{13}$$

where γ^{PSP} is the discount rate (0< γ^{PSP} <1.0). Therefore, the postsynaptic action potential is excitatory if the weight parameter, $w_{j,i}$ is positive, and inhibitory if $w_{j,i}$ is negative. In our case we set $w_{j,i}$=-0.2 in order to suppress the firing chance of other neurons when a given neuron fires.

In the equations describing the three components of the internal state simple functions are used instead of the differential equations proposed in the original model of spiking neural network [33]. By the proposed simple spike response model we can keep the computational complexity at low level.

3.2 Evolution Strategy for Optimizing the Parameters of FSNN

We apply (μ+ λ)-Evolution Strategy (ES) for the improvement of the parameters of fuzzy spiking neural network in the fuzzy rules. In (μ+ λ)-ES μ and λ indicate the number of parents and the number of offspring produced in a single generation, respectively [34]. We use (μ+1)-ES to enhance the local hill-climbing search as a continuous model of generations, which eliminates and generates one individual in a generation. The (μ+1)-ES can be considered as a steady-state genetic algorithm (SSGA) [35]. As it can be seen in Equations (6), (7), (9) and Figure 9, a candidate solution will contain the parameters of the fuzzy membership functions which play role in the input layer of the spiking neural network. These parameters are the central value ($a_{i,j}$), the width ($b_{i,j}$), and the contribution value ($v_{i,j}$):

$$\begin{aligned} \mathbf{g}_k &= \left[g_{k,1}\ g_{k,2}\ g_{k,3} \cdots g_{k,l} \right] \\ &= \left[a_{k,1,1}\ b_{k,1,1}\ v_{k,1,1} \ldots v_{k,n,m} \right] \end{aligned} \tag{14}$$

where n is the number of human interaction modes; m is the number of inputs; $l = n \cdot m$ is the chromosome length of the k-th candidate solution. The fitness value of the k-th candidate solution is calculated by the following equation:

$$f_k = \sum_{i=1}^{n} f_{k,i} \tag{15}$$

where $f_{k,i}$ is the number of correct estimation rates of the i-th human interaction mode. We compare the FSNN's each output in the time sequence with the corresponding desired output. If the FSNN's output is the same as the desired output, then we count this as a matching. The number of matchings for the i-th interaction mode is $f_{k,i}$. Thus, the evaluation of the individual and consequently the learning process is performed in supervised manner.

In (μ+1)-ES, only one existing solution is replaced with the candidate solution generated by crossover and mutation. We use elitist crossover and adaptive mutation. Elitist crossover randomly selects one individual, and generates one individual by combining genetic information between the selected individual and the best individual in order to obtain feasible solutions from the previous estimation result rapidly. The newly generated individual replaces the worst individual in the population after applying adaptive mutation on the newly generated individual. In the genetic operators we use the local evaluation values of the human interaction mode estimation. The inheritance probability of the genes corresponding to the *i*-th rule of the best individual is calculated by:

$$p_i = \frac{1}{2} \cdot (1 + f_{best,i} - f_{k,i}) \tag{16}$$

where $f_{\mathrm{best,i}}$ and $f_{k,i}$ are the part of the fitness value related to the *i*-th rule (*i*-th genes) of the best and the randomly selected *k*-th individuals, respectively. By Eq. (16) we can bias the selection probability of the *i*-th genes from 0.5 to the direction of the better individual's *i*-th genes among the best individual and the *k*-th individual. Thus, the newly generated individual can inherit the *i*-th genes (*i*-th rule) from that individual which the better *i*-th genes has. After the crossover operation, an adaptive mutation is performed on the generated individual:

$$g_{k,h} \leftarrow g_{k,h} + \alpha_h \cdot (1 - t/T) \cdot N(0,1) \tag{17}$$

where $N(0,1)$ indicates a normal random value; α_h is a parameter of the mutation operator (*h* stands for identifying the three subgroups in the individual related to *a*, *b*, and *v*); *t* is the current generation; and *T* is the maximum number of generations.

4 Experimental Results

This section shows comparison results and analyzes the performance of the proposed method. In the spiking neural network there are 5 inputs in the input layer: acceleration, angular velocity, and attitude of pitch, roll yaw. In the output layer there are 7 outputs related to the following 7 robot interaction modes (Fig. 10): (1) TableMode(front), (2) TableMode(back), (3) RobotMode, (4) HandMode, (5) BreastPocketMode, (6) JacketPocketMode, (7) TrouserPocketMode. The parameters of the neural network are as follows: the temporal discount rate for refractoriness (γ^{ref}) is 0.88, the temporal discount rate for PSP (γ^{PSP}) is 0.9, the threshold for firing (q^{pul}) is 0.9, and R is 1. Fourteen training datasets and 4 test datasets are used in the experiments. When obtaining the training set, in the case of BreastPocketMode, JacketPocketMode, and TrouserPocketMode the person was walking for about 2 minutes, then standing for about 2 minutes. In the TableModes, RobotMode, and HandMode there was no motion.

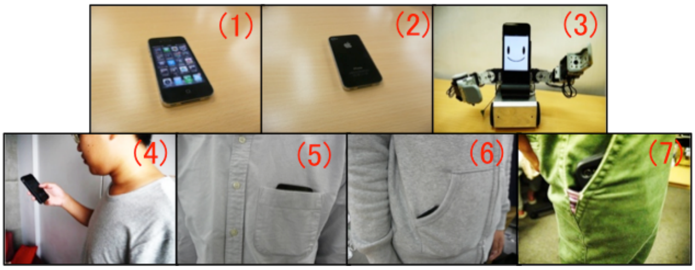


Figure 10
Robot interaction modes

Figure 11 illustrates the experimental example of the measured smart phone mode. The cyan line is the high-pass filtered data measured by the accelerometer. The green line depicts the angular velocity calculated by the low-pass filtered data measured by the gyro sensor. The red line is the attitude of pitch data. The blue line is the attitude of roll data. The pink line is the attitude of yaw data. The second part of Fig. 11 shows the target output (blue line) and the estimated output by FSNN (red line). The number of spiking neurons is 5. These neurons are used for measuring the 7 robot interaction modes.

In the first experiment the sensor's raw data are used as input to the FSNN. Figure 11 shows experimental results by using the raw data of the second training dataset. In this case the phone is put on the table by the front side (a), and the person takes it in hand (b). Thereafter he/she puts the phone in jacket pocket (c) and takes out the phone putting it on the robot base (b,d). After that the person puts the phone in trouser pocket (b,e). Then he/she takes out the phone putting it on the table by the back side (b,f), and finally he/she takes the phone putting it in breast pocket (b,g). The number of fitting data is 16911 from 19213 training data in the case of second dataset. There are 14 training datasets, the total number of fitting data is 64936 from 72638 training data, and the running time is 4410 ms. The fitting rate is 89.4%. Figure 12 shows experimental results by using the raw data of second test dataset. The number of fitting data is 5966 from 7974 test data. There are 4 test datasets, the total number of fitting data is 31377 from 40096 test data, and the running time is 1688 ms. The fitting rate is 78.3%.

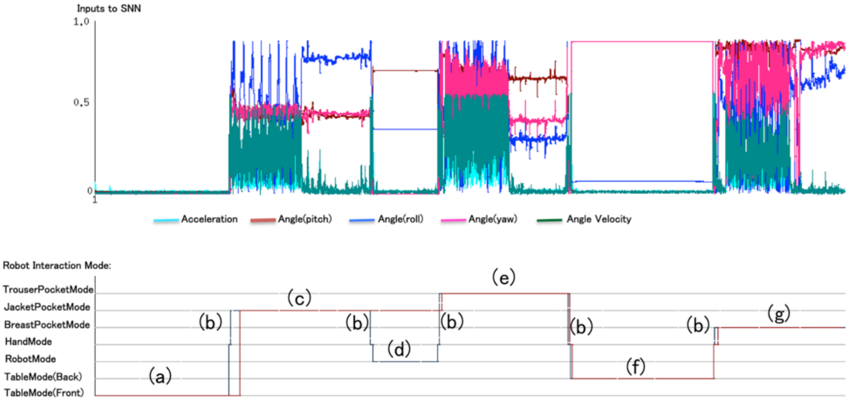


Figure 11

Experimental results by using the raw data for training dataset 2

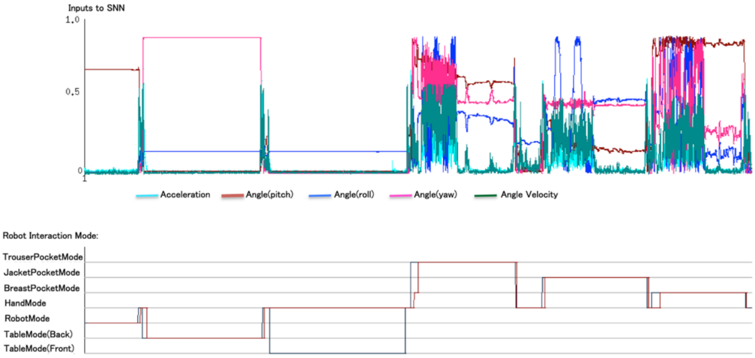


Figure 12

Experimental results by using the raw data for test dataset 2

In order to reduce the noise we have to use some smoothing functions as mentioned in Section 2. We apply two different kinds of weighted moving averages. In the second experiment we present results by using smoothing function described in Eq. (4). Figure 13 depicts the result for second training dataset. The number of fitting data is 18234 from 19213 training data. The total number of fitting data using all training datasets is 63659 from 72638 training data, the running time is 4445 ms, and the fitting rate is 87.6%. Figure 14 shows experimental results by using smoothing function in Eq. (4) for test dataset 2. The number of fitting data is 6911 from 7974 training data. The total number of fitting data using all test datasets is 34327 from 40096 test data, the running time is 1609 ms, the fitting rate is 85.6%.

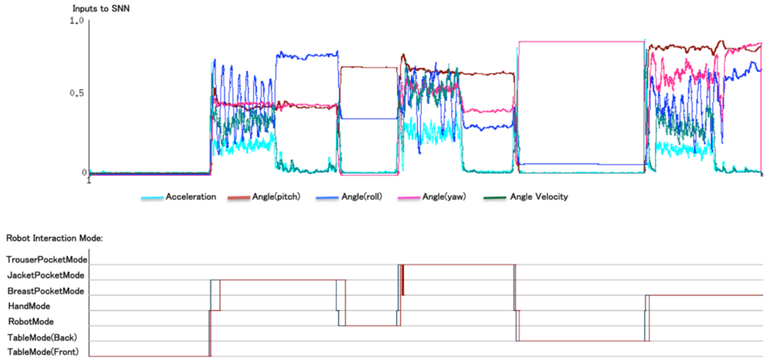


Figure 13

Experimental results by using the smoothing function in Eq. (4) for training dataset 2

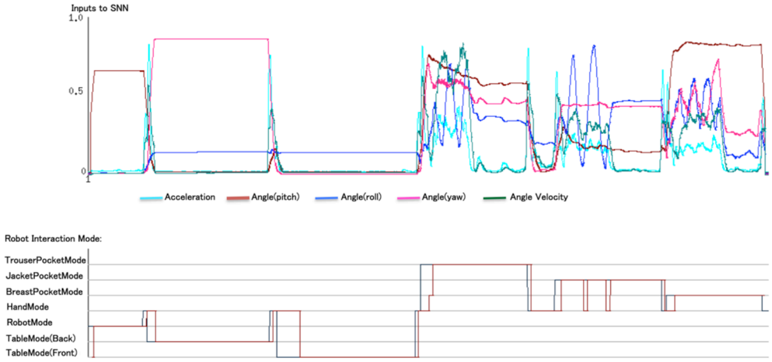


Figure 14

Experimental results by using the smoothing function in Eq. (4) for test dataset 2

In the third experiment, we present experimental result by the other smoothing function defined by Eq. (5). Figure 15 illustrates the results for second training dataset. In the case of second training dataset the number of fitting data is 18507 from 19213, and for all training datasets the total number of fitting data is 69356 from 72638, the running time is 4438 ms, the fitting rate is 95.5%. Figure 16 shows experimental results by using smoothing function in Eq. (5) for test dataset 2. The number of fitting data is 7499 from 7974 test data. The total number of fitting data for all test datasets is 36842 from 40096 test data, the running time is 1610 ms, the fitting rate is 91.9%.

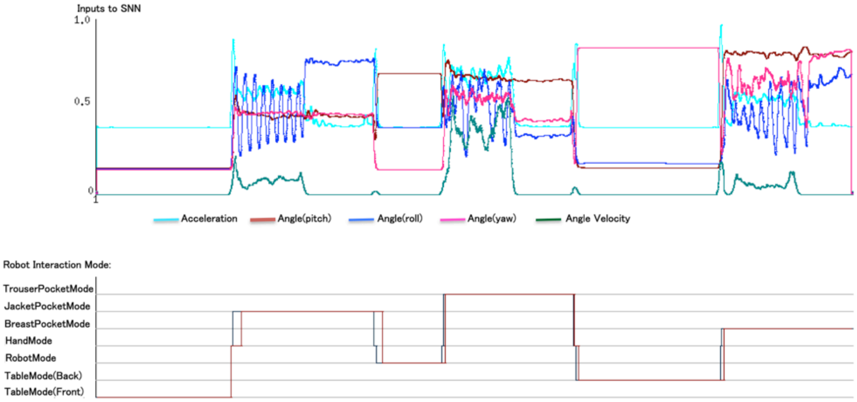


Figure 15

Experimental results by using the smoothing function in Eq. (5) for training dataset 2

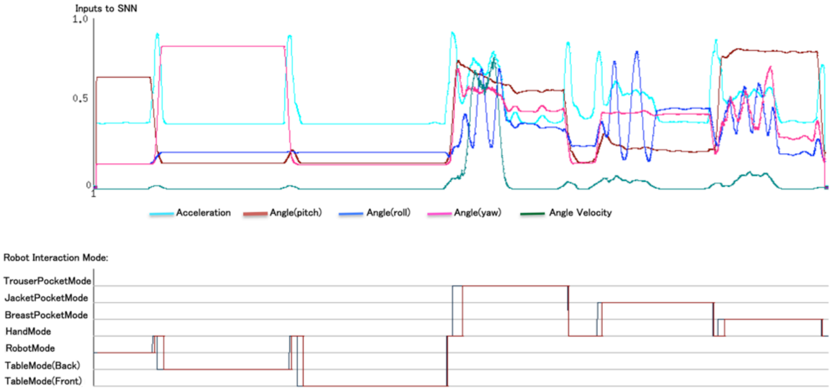


Figure 16

Experimental results by using the smoothing function in Eq. (5) for test dataset 2

In the fourth experiment we use evolution strategy for optimizing the parameters of FSNN. The population size is 100, the number of generations is 6000, and the evaluation time step is 1000, α_a=0.01, α_b=0.005, α_v=0.05. Figure 17 illustrates the best results by ES for the raw data of second training dataset. The total number of fitting data using all training datasets is 68933 from 72638. The application of ES has an additional computational cost. The running time is 684259 ms, the fitting rate is 94.9%. Figure 18 shows the result for second test dataset after using evolution strategy for optimizing the parameters of FSNN based on training datasets. In the case of test datasets the total number of fitting data is 34842 from 40096 test data. The running time is 1810 ms, the fitting rate is 86.9%.

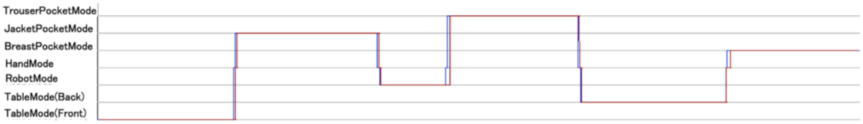


Figure 17

Experimental results by using ES for parameter optimization for training dataset 2 (for raw data)

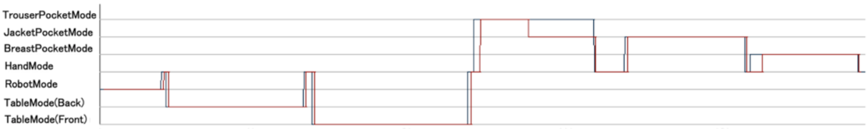


Figure 18

Experimental results by using FSNN for test dataset 2 after parameter optimization (for raw data)

Figure 19 depicts the best results by ES for training dataset 2 when using smoothing function in Eq.(4). The total number of fitting data using all training datasets is 69441 from 72638. The running time is 683293 ms, the fitting rate is 95.6%. Figure 20 shows the result for second test dataset when using smoothing function in Eq.(4) after using evolution strategy for optimizing the parameters of FSNN based on training datasets. In the case of test datasets the number of fitting data is 35805 from 40096 test data. The running time is 1743 ms, the fitting rate is 89.3%.

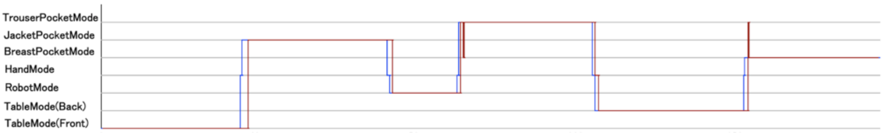


Figure 19

Experimental results by using ES for parameter optimization for training data set 2 (using smoothing function in Eq. (4))

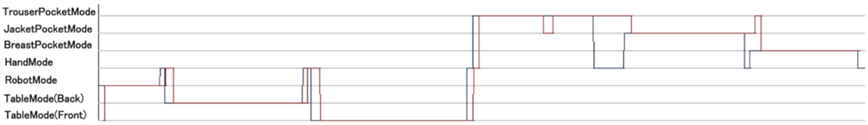


Figure 20

Experimental results by using FSNN for test dataset 2 after parameter optimization (using smoothing function in Eq. (4))

Figure 21 presents the best results by ES for training dataset 2 when using smoothing function in Eq.(5). The total number of fitting data using all training datasets is 70604 from 72638. The running time is 654505 ms, the fitting rate is 97.2%. Figure 22 shows the result for second test dataset when using smoothing function in Eq.(5) after using evolution strategy for optimizing the parameters of FSNN based on training datasets. In the case of test datasets the number of fitting data is 37730 from 40096 test data. The running time is 1586 ms, the fitting rate is 94.1%.

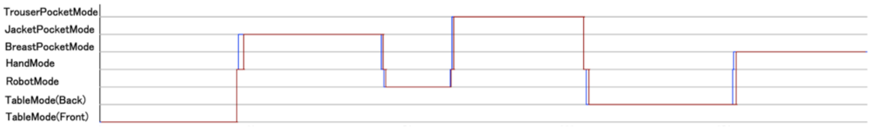


Figure 21

Experimental results by using ES for parameter optimization for training data set 2 (using smoothing function in Eq. (5))

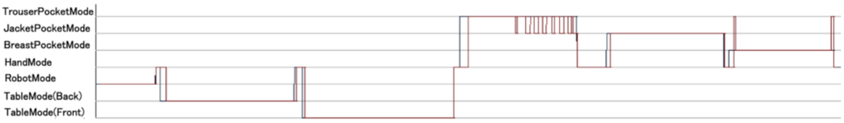


Figure 22

Experimental results by using FSNN for test dataset 2 after parameter optimization (using smoothing function in Eq. (5))

Table 2 summarizes the experimental results. The best results were obtained by evolution strategy.

Table 2

Summary of experimental results

Sensor Name	Experiment	Number of data	Number of fitting data	Fitting rate	Running time(ms)
Raw data	traning	72638	64936	89.4%	4410
	test	40096	31377	78.3%	1688
Smoothing function Eq.(4)	traning	72638	63659	87.6%	4445
	test	40096	34327	85.6%	1609
Smoothing function Eq.(5)	traning	72638	69356	95.5%	4438
	test	40096	36842	91.9%	1610
ES for raw data	traning	72638	68933	94.9%	684259
	test	40096	34842	86.9%	1810
ES for Smoothing function Eq.(4)	traning	72638	69441	95.6%	683293
	test	40096	35805	89.3%	1743
ES for Smoothing function Eq.(5)	traning	72638	70604	97.2%	654505
	test	40096	37730	94.1%	1586

5 Summary

In this paper, we proposed a method for estimating human interaction mode using accelerometer, gyro, and magnetometer. First, we introduced the robot partners applied in this paper. Next, we proposed an estimation method of human interaction modes using evolution strategy and fuzzy spiking neural network based on a simple spike response model. In the experimental results we showed, that the proposed method is able to estimate human interaction modes based on iPhone's sensors.

As a future work, we intend to improve the learning performance according to human life logs, and extend the method by combining it with the estimation of human transport modes which has been presented in [26] .

Acknowledgement

This work is partially supported by MEXT Regional Innovation Strategy Support Program.

References

[1] Statistics Bureau, Ministry of Internal Affairs and Communications, "Population estimates", September, 2012.

[2] M. Pollack, "Intelligent Technology for an Aging Population: The Use of AI to Assist Elders with Cognitive Impairment," AI Magazine, 2005, vol. Summer.

[3] T. Hashimoto, N. Kato, and H. Kobayashi, "Study on Educational Application of Android Robot SAYA: Field Trial and Evaluation at Elementary School," ICIRA 2010, Part II, LNAI 6425, pp. 505–516, 2010.

[4] H. Ishiguro, M. Shiomi, T. Kanda, D. Eaton, and N. Hagita, "Field Experiment in a Science Museum with communication robots and a ubiquitous sensor network," Proc. of Workshop on Network Robot System at ICRA2005, 2005.

[5] N. Kubota and T. Mori, "Conversation System Based on Boltzmann Selection and Bayesian Networks for A Partner Robot," Robot and Human Interactive Communication (RO-MAN), Toyama, Japan, 2009.

[6] H. Kimura, N. Kubota, and J. Cao, "Natural Communication for Robot Partners Based on Computational Intelligence for Edutainment," MECATRONICS2010, pp.610-615, Yokohama, Japan, 2010.

[7] A. Yorita and N. Kubota, "Cognitive Development in Partner Robots for Information Support to Elderly People," IEEE Transactions on Autonomous Mental Development, Vol. 3, Issue 1, pp. 64-73, 2011.

[8] N. Kubota, T. Mori, and A. Yorita, “Conversation System for Robot Partners based on Informationally Structured Space,” IEEE Symposium Series on Computational Intelligence 2011 (SSCI2011), Paris, France, April 11-15, 2011.

[9] D. Tang and N. Kubota, “Information Support System Based on Sensor Networks,” Proc. of World Automation Congress (WAC) 2010, Kobe, Japan, September 19-23, 2010.

[10] D. Tang, B. Yusuf, J. Botzheim, N. Kubota, and I. A. Sulistijono, “Robot Partner Development Using Emotional Model Based on Sensor Network,” Proc. of IEEE Conference on Control, Systems and Industrial Informatics (ICCSII 2012), pp. 196-201, 2012.

[11] N. Kubota and Y. Toda, “Multi-modal Communication for Human-friendly Robot Partners in Informationally Structured Space,” IEEE Transaction on Systems, Man, and Cybernetics-Part C, Vol. 42, No. 6, pp. 1142-1151, 2012.

[12] I. Khemapech, I. Duncan, and A. Miller, “A Survey of Wireless Sensor Networks Technology,” Proc. of The 6th Annual PostGraduate Symposium on The Convergence of Telecommunications, Networking and Broadcasting, 2005.

[13] I. Satoh, “Location-based Services in Ubiquitous Computing Environments,” International Journal of Digital Libraries, Springer, 2006.

[14] P. Remagnino, H. Hagras, N. Monekosso, and S. Velastin, “Ambient Intelligence: A Gentle Introduction,” In the book entitled “Ambient Intelligence A Novel Paradigm” (Eds: P. Remagnino, G. Foresti, T. Ellis), Springer Verlag, pp. 1-15, 2005.

[15] V. Callaghan, M. Colley, H. Hagras, J. Chin, F. Doctor, and G. Clarke, “Programming iSpaces: A Tale of Two Paradigms,” In the book entitled “Intelligent Spaces: The Application of Pervasive ICT” (Eds: A. Steventon, S. Wright), Springer-Verlag, Chapter 24, pp. 389- 421, 2005.

[16] K. Morioka, and H. Hashimoto, “Appearance Based Object Identification for Distributed Vision Sensors in Intelligent Space,” Proc. of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'04), Vol.1, pp. 199-204, 2004.

[17] K. Kemmotsu, T. Tomonaka, S. Shiotani, Y. Koketsu, and M. Iehara, “Recognizing Human Behaviors with Vision Sensors in Network Robot Systems,” Proc. of The 1st Japan-Korea Joint Symposium on Network Robot Systems (JK-NRS2005), 2005.

[18] N. Kubota and A. Yorita, “Topological environment reconstruction in informationally structured space for pocket robot partners,” in Proc. of the 2009 IEEE International Symposium on Computational Intelligence in Robotics and Automation, CIRA, pp. 165-170, 2009.

[19] N. Kubota, H. Sotobayashi, and T. Obo, “Human interaction and behavior understanding based on sensor network with iPhone for rehabilitation,” in Proc. of the International Workshop on Advanced Computational Intelligence and Intelligent Informatics, 2009.

[20] N. Kubota, D. Tang, T. Obo, and S. Wakisaka, “Localization of Human Based on Fuzzy Spiking Neural Network in Informationally Structured Space,” in Proc. of 2010 IEEE World Congress on Computational Intelligence (WCCI 2010), Barcelona, Spain, pp. 2209-2214, 2010.

[21] T. Obo, N. Kubota, and B. H. Lee, “Localization of Human in Informationally Structured Space Based on Sensor Networks,” in Proc. of 2010 IEEE World Congress on Computational Intelligence (WCCI 2010), Barcelona, Spain, pp. 2215-2221, 2010.

[22] N. Kubota, K. Yuki, and N. Baba, “Integration of Intelligent Technologies for Simultaneous Localization and Mapping,” in Proc. of ICROS-SICE International Joint Conference 2009 (ICCAS-SICE 2009), Fukuoka, Japan, 2009.

[23] M. Satomi, H. Masuta, and N. Kubota, “Hierarchical Growing Neural Gas for Information Structured Space,” in Proc. of the IEEE Symposium Series on Computational Intelligence 2009.

[24] N. Kubota, T. Obo, and T.Fukuda, “An Intelligent Monitoring System based on Emotional Model in Sensor Networks,” in Proc. of the 18th IEEE International Symposium on Robot and Human Interactive Communication, pp. 346-351, 2009.

[25] D. Tang and N. Kubota, “Human Localization by Fuzzy Spiking Neural Network Based on Informationally Structured Space,” in Proc. of the 17th International Conference on Neural Information Processing (ICONIP 2010), Sydney, Australia, pp. 25-32, 2010.

[26] D. Tang, J. Botzheim, N. Kubota, and T. Yamaguchi, “Estimation of Human Transport Modes by Fuzzy Spiking Neural Network and Evolution Strategy in Informationally Structured Space,” IEEE International Workshop on Genetic and Evolutionary Fuzzy Systems (GEFS), Singapore, April 16-19, pp. 36-43, 2013.

[27] J. Botzheim, D. Tang, B. Yusuf, T. Obo, N. Kubota, and T. Yamaguchi, “Extraction of Daily Life Log Measured by Smart Phone Sensors using Neural Computing,” 17th International Conference in Knowledge Based and Intelligent Information and Engineering system - KES2013, Procedia Computer Science 22, pp. 883-892, 2013.

[28] http://www.letsgodigital.org/en/23646/smartphone-price/

[29] http://green.tmcnet.com/news/2013/02/18/6929794.htm

[30] http://www.nuance.com/for-partners/by-solution/mobile-developer-program/index.htm

[31] J. A. Anderson and E.Rosenfeld, Neurocomputing. The MIT Press, Cambridge, Massachusetts, US, 1988.

[32] W. Gerstner, "Spiking Neurons," In the book entitled "Pulsed Neural Networks" (Eds: W. Maass and C. M. Bishop), Chapter 1, MIT Press, pp. 3-53, 1999.

[33] W. Gerstner and W. M. Kistler, Spiking Neuron Models. Cambridge University Press, 2002.

[34] H.-P. Schwefel, Numerical Optimization of Computer Models. John Wiley & Sons, New York, 1981.

[35] G. Syswerda, "A Study of Reproduction in Generational and Steady-State Genetic Algorithms," In Foundations of Genetic Algorithms, Morgan Kaufmann Publishers, Inc., pp. 94-101, 1991.