Particle Filter Based Feedback Control of JAIST Active Robotic Walker

Takanori Ohnuma, Geunho Lee, and Nak Young Chong

***Abstract*— We present a new control scheme of JAIST Active Robotic Walker (JARoW) developed to provide potential users such as the elderly with sufficient ambulatory capability. Toward its practical use, we tackle JARoW's easy and reliable maneuverability by creating a natural user interface between a user and JARoW. Specifically, our focus is placed on how to realize the natural and smooth movement of JARoW despite different gait parameters of users. For this purpose, a particle filtered interface function (PFIF) is proposed to estimate and predict the locations of the user's legs and body. Then, the simple feedback motion control function adjusts the motions of JARoW corresponding to the estimation and prediction. Experimental results show that the proposed control scheme can be quite satisfactory for practical use without requiring any additional user effort.**

I. Introduction

Personal mobility aids are strongly desired to help elderly and/or lower limb disabled people stay independent. Recent advances in robot technology have provided a solid foundation for the development of various walking aids. Notable examples include wheelchairs [1][2], canes [3], and walkers [4]-[7]. Specifically, robotic walkers can be further divided into passive [4][5] and active walkers [6][7]. The features of passive walkers include low cost, simple structure, and compact size. However, users must take overly cautious steps not to push it out too far forward. Also, it is deemed to be unsafe to use on uneven/slope terrain. Active walkers may provide both ambulatory aid and rehabilitation. However, they are still bulky and costly, and their complicated operation often requires considerable skill to use.

Toward more widespread use of active walkers, there are important things that need to be taken into consideration. For instance, the human gait control system is nonlinear, and the gait parameters vary across users. To meet these requirements, voice activation systems have been presented in [8]. Despite their many advantages, there are critical problems such as interference and recognition that remain to be resolved. Instead of using the user's command directly, it would be very convenient if his/her intention can be recognized by robotic walkers. A few examples include the visual recognition using cameras [9] and human gait detection based on pressure sensors [10]. However, under real world conditions, it is difficult to guarantee their reliability.

As shown in Fig. 1, JARoW was developed in 2009 [11] to provide potential users with sufficient ambulatory capability in all directions and easy-to-use features. Toward enabling the practical use of JARoW, there still remains a challenge on how to realize its easier and more reliable maneuverability. As a first step to achieve this, our challenge is placed on how to create a natural user interface between JARoW and users with different gait parameters and to employ a simple feedback control without requiring any special-purpose interface. As the main contribution of this paper, a particle filtered interface function (PFIF) is proposed to estimate and predict the locations of the user's lower limbs and body, resulting in the JARoW's natural and smooth movement. The PID feedback motion control function can then adjust the motions of JARoW corresponding to the user's walking behaviors. We describe the proposed control algorithms in detail, and perform extensive experiments to demonstrate their effectiveness in our laboratory environment.

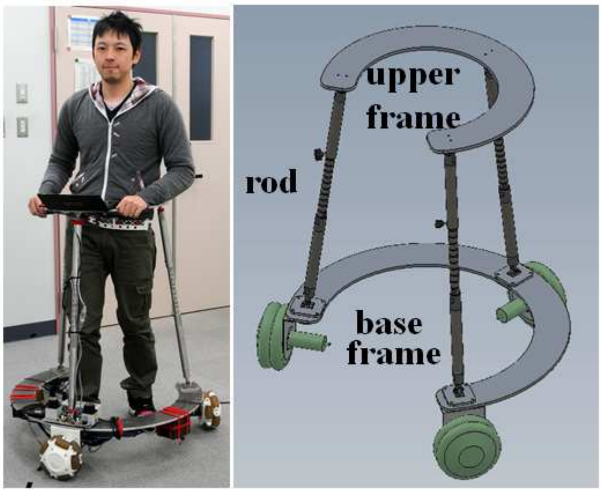


Fig. 1. JAIST active robotic walker (JARoW) prototype

II. JARoW: Prototype Description

A. Control Architecture

JARoW is designed to autonomously adjust its motion according to the user's walking behavior without requiring any additional user effort. Specifically, as illustrated in Fig. 2, JARoW consists of the drive-train, the interface system, and the main controller. We use a laptop PC that runs on Microsoft's Windows XP as the main controller. The input to the main controller includes the measurement data obtained from the interface system. Based on the data, PFIF estimates and predicts the locations of the user's body with respect to JARoW's local coordinates. The feedback motion control function then outputs the desired velocity matrix to the drive-train at each time step. Details on the control scheme will

This work was supported in part by The FY2011 Mitani Foundation for Research and Development.

The authors are with the School of Information Science, Japan Advanced Institute of Science and Technology, Ishikawa, Japan {t-ohnuma, geun-lee, nakyoung}@jaist.ac.jp

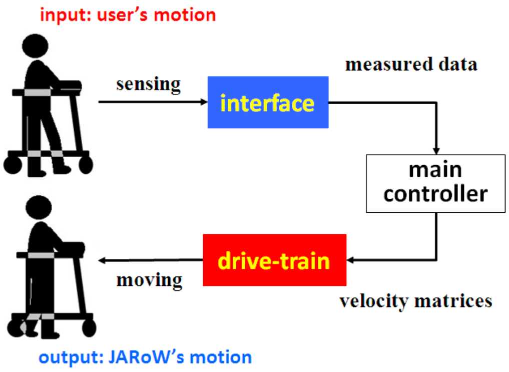


Fig. 2. Schematic for the overall control flow

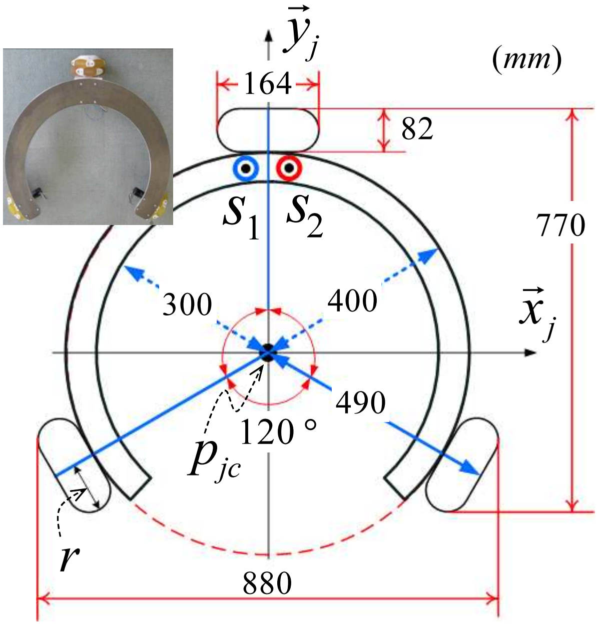


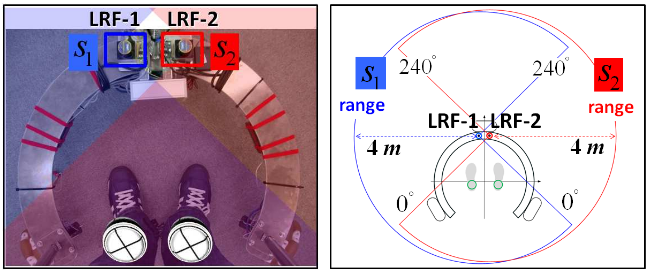
Fig. 3. Specification and notations of the base frame

be explained in the following sections.

B. Mechanical Structure and Hardware Configuration

A compact design, such as 825 $\sim$ 1000 *mm* in height, and 880 *mm* and 770 *mm* in horizontal diameters, saves cost and uses less space. It is therefore considered to be suited for use in various environments including narrow hallways or elevators. As shown in Fig. 1, the outline is a circular shape to reduce possible collisions with obstacles or walls. Its stiff and light design of 20 *kg* is achieved with an aluminum alloy.

JARoW has three main structural parts: base frame, upper frame, and connecting rods. The base frame is to support the superstructure, and is directly connected to three omni-directional wheels and equipped with a pair of Hokuyo URG-04LX laser range finders (LRFs) detecting the user's lower limb locations. The length of the connecting rod can be changed up to 175 *mm* according to the height of users. Users are able to lean their upper body forward and place their forearm onto the upper frame. The main controller is mounted on top of the upper frame.



(a) cylinder-like model (b) coverage area of LRFs

Fig. 4. Lower limbs modeled as a cylinder with a diameter d

The drive-train is composed of three Mecanum wheels, three motors equipped with encoder and 43:1 gear reduction unit, three motor drivers, and one motor controller. Three Mecanum wheels are mounted underneath the base frame 120 degrees apart from each other (see Fig. 3), allowing JARoW to move forward/backward, slide sideways, and turn right/left. Such omni-directionality provides an efficient means of direction control in highly cluttered indoor environments. The maximum stall torque is determined in such a way that JARoW can accommodate up to 90 *kg*. It is reported that the average maximum walking speed for elderly pedestrians is 4.8 km/h on flat terrain [12]. The maximum continuous torque is determined to meet the maximum velocity requirement 6.58 km/h of the drive-train.

C. Interface System

As shown in Fig. 4, a pair of LRFs detects the locations of the user's lower limbs, as well as obstacles or area borders. The lower limbs are modeled as a cylinder with a diameter d, representing each shin as illustrated in Fig. 4-(a). Further details on this model can be found in [11]. We define a valid region for the location measurement of shins as a rectangle with 900 $\times$ 700 *mm* (length and width) inside the base frame. It is assumed that the locations of shins always remain within its region.

Each LRF outputs a 240 degree scan and measures up to 4000 *mm* with 100 *ms* sampling time. Accordingly, a pair of LRFs can cover a full 360 degrees. After each scan, the range data from the individual LRF rays are sorted into surface information of shins and obstacles, respectively, according to pre-determined regions. The interface system outputs the LRF-to-surface distance that is fed to the main controller.

D. Kinematics

As shown in Fig. 3, JARoW has its local coordinates $\vec{x}_j$ (vertical axis) and $\vec{y}_j$ (horizontal axis). Its center position is denoted as p_{jc}. S_1 and S_2 denote the positions of LRFs. For a mobile robot with Mecanum wheels, the *Instantaneous Center of Rotation* (ICR) corresponds to its centroid during rotation. In contrast, we set the ICR to the center of the user body defined as $p_{bc} = (x_b, y_b)$ in Fig. 5, the midpoint of the line segment connecting the center points of two circles projecting both shins onto the ground. By doing this, JARoW effectively avoids being bumped into the user while it rotates.

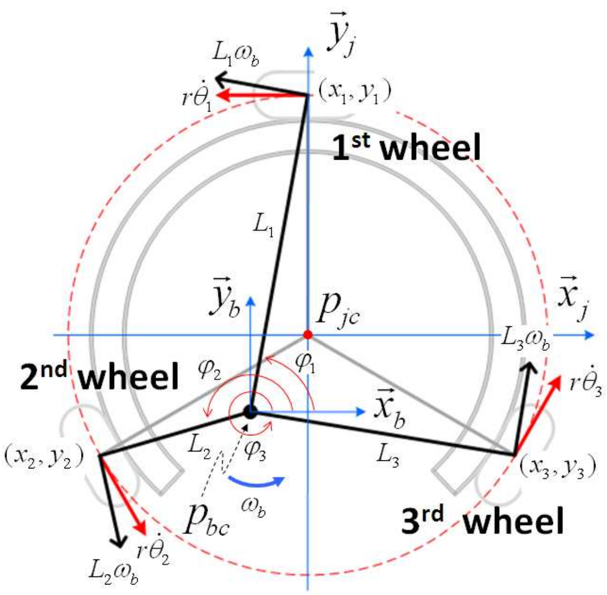


Fig. 5. Illustration of JARoW kinematics

Now, the first-order kinematics of JARoW can be derived. $\dot\theta_i$ and φ_i denote the driving angular velocity of the *i*-th wheel and the angular displacement of the *i*-th wheel relative to $\vec{x}_b$-axis in the user body reference frame $\vec{x}_b$ and $\vec{y}_b$, respectively. Using the tangent formula, φ_i is computed:

$$\varphi_i = \tan^{-1}(\frac{y_i - y_b}{x_i - x_b}). \qquad (1)$$

Next, L_i which denotes the distance between p_{bc} and each wheel position (x_i, y_i) is given by

$$L_i = \sqrt{(y_i - y_b)^2 + (x_i - x_b)^2}. \qquad (2)$$

For the desired JARoW velocity vector $[\dot x_b \ \dot y_b \ \omega_b]^t$, the angular velocities $\dot\theta_1$, $\dot\theta_2$, and $\dot\theta_3$ of individual wheels through the inverse Jacobian are derived:

$$\begin{bmatrix}\dot\theta_1\\ \dot\theta_2\\ \dot\theta_3\end{bmatrix} = \frac{1}{r}\begin{bmatrix} -1 & 0 & L_1\cos(\frac{\pi}{2}-\varphi_1)\\ \cos\frac{\pi}{3} & -\sin\frac{\pi}{3} & L_2\cos(\frac{7\pi}{6}-\varphi_2)\\ \cos\frac{\pi}{3} & \sin\frac{\pi}{3} & L_3\cos(\frac{\pi}{6}+\varphi_3)\end{bmatrix}\begin{bmatrix}\dot x_b\\ \dot y_b\\ \omega_b\end{bmatrix}, \qquad (3)$$

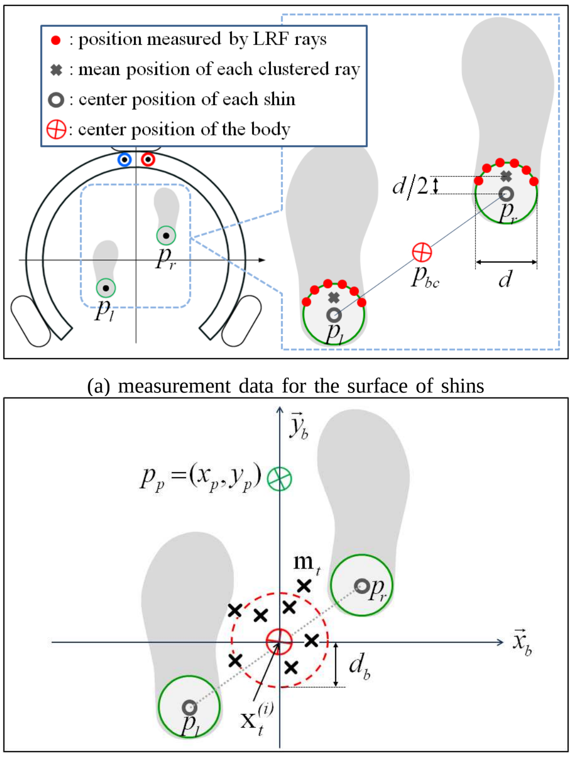
where r denotes the wheel radius.

III. Particle Filtered Interface Function (PFIF)

Our PFIF is to measure the locations of both shins, to estimate the body position of a user using the measured shins' locations, and to predict the next position of the body. According to the phase, the interface function is implemented as follows.

A. Formulation

Measurement data for the surface of shins are represented as red circles in Fig. 6-(a). After each sampling time, they are divided into left and right clusters, and then the mean positions of these clusters are calculated. Through preliminary tests, we empirically learned that the mean positions were in the immediate vicinity of the cylinder's surface and the distances from the mean position to p_r or p_l were approximately one-half of its diameter d. Therefore, $p_l = (x_l, y_l)$ and $p_r = (x_r, y_r)$ can be obtained by adding the mean position to $d/2$. Based on p_r and p_l, the p_{bc} candidate is defined as the midpoint on the line segment connecting these estimates as illustrated in Fig. 6. Specifically, $p_{bc} = (x_b, y_b)$ is considered to be the body position. The motion of p_{bc} is also assumed to be at constant velocity. (Then, the particle filtering formulations based on p_r and p_l can be utilized for the body matching.)


● : position measured by LRF rays
✖ : mean position of each clustered ray
○ : center position of each shin
⊕ : center position of the body
p_r
p_l
$d/2$
p_{bc}
d
$\vec{y}_b$
$p_p = (x_p, y_p)$
$\mathbf{m}_t$
p_r
$\vec{x}_b$
d_b
p_l
$\mathbf{x}_t^{(i)}$

(a) measurement data for the surface of shins

(b) particle $\mathbf{x}_t^{(i)}$ for location vector of the body

Fig. 6. Definitions and notations used in the particle filtered interface function (PFIF)

As shown in Fig. 6-(b), the variable of interest at time t is the location of the body with respect to the walker's local coordinates, represented as a set of n samples (*i.e.*, particles $s_t^{(i)} = \{\mathbf{x}_t^{(i)}, w_t^{(i)}\} : i = 1, 2, \cdots, n$), where i denotes the i-th particle. Each particle consists of its 2-D location vector $\mathbf{x}_t^{(i)}$ and its associated weight $w_t^{(i)}$. In particular, $\mathbf{x}_t^{(i)}$ with respect to the walker's local coordinates is defined:

$$\mathbf{x}_t^{(i)} = [x_{b,t}^{(i)} \ y_{b,t}^{(i)}]^T. \qquad (4)$$

The state vector for a given $\mathbf{x}_t$ is defined as X_t. p_p in Fig. 6-(b) denotes the predicted position of the body at $t+1$. Moreover, the sensor observation vector for measurement data $\mathbf{m}_t$ is defined as M_t.

B. Location Estimation

To estimate the posterior probability $p(X_t|M_t)$ of X_t given M_t, we need to have a desired model. Our scheme in the estimation phase is realized as follows.

- Step-1: initialization

As a set of particles estimating the location of the body at $t = 0$, the number of n initial particles are generated and denoted by $\{s^{(i)}_{0|0} | 1 \leq i \leq n\}$, where $s^{(i)}_{0|0}$ indicates $\{\mathbf{x}^{(i)}_{0|0}, w^{(i)}_{0}\}$. By the use of the Metropolis-Hastings (for simplicity, M-H) algorithm [13], $\mathbf{x}^{(i)}_{0}$ is obtained through sampling drawn from a Gaussian distribution with the variance vector $\sigma^{(2)}_{0}$ and the mean vector μ_0. It is assumed that w_0 is a constant.

- Step-2: system model at time t

The system model representing the forward and backward movements of both lower limbs is formalized as follows:

$$\mathbf{x}^{(i)}_{t|t-1} = \begin{bmatrix} x^{(i)}_{b,t|t-1} \\ \\ y^{(i)}_{b,t|t-1} \end{bmatrix} = \begin{bmatrix} x^{(i)}_{b,t-1|t-1} \\ \\ y^{(i)}_{b,t-1|t-1} \end{bmatrix} + \begin{bmatrix} n^{(i)}_{bx,t-1} \\ \\ n^{(i)}_{by,t-1} \end{bmatrix}, \quad (5)$$

where $n^{(i)}_{bx,t-1}$ and $n^{(i)}_{by,t-1}$ denote the random system noises, which are sampled from a Gaussian distribution with σ^2_y and μ_y using the M-H algorithm. By using (5), $\mathbf{x}^{(i)}_{t|t-1}$ and $s^{(i)}_{t|t-1}$ are computed from $\mathbf{x}^{(i)}_{t-1|t-1}$ in $s^{(i)}_{t-1|t-1}$.

- Step-3: computation of the *i*-th posterior probability

Given the state vector X_t, the posterior probability $p^{(i)}(X_t|M_t)$ of the observation vector M_t is computed:

$$p^{(i)}(X_t|M_t) = \frac{1}{\sqrt{2\pi}\sigma_s} \exp(-\frac{D^{(i)^2}}{2\sigma^2_s}), \quad (6)$$

where σ_s denotes the standard deviation for the permissible location error. In addition, the Euclidean distance $D^{(i)}$ between $\mathbf{m}_t$ and $\mathbf{x}^{(i)}_t$ is defined as:

$$D^{(i)}_k = \|\mathbf{x}^{(i)}_{t|t-1} - \mathbf{m}_t\|. \quad (7)$$

To reduce computational loads practically, $\mathbf{x}^{(i)}_{t|t-1}$ is not regarded as the representation for the location of the body in the following three cases: 1) $\mathbf{x}^{(i)}_{t|t-1}$ located on the base frame, 2) $\|\mathbf{x}^{(i)}_{t|t-1} - \mathbf{m}_t\| < d_b$ (see Fig. 6-(b)), and 3) $\mathbf{x}^{(i)}_{t|t-1} < \mathbf{m}_t$ with respect to the walker's local coordinates. In our implementation, $p^{(i)}(X_t|M_t)$ corresponding to any aforementioned cases is set to *0*.

- Step-4: computation of the *i*-th weight

The associated weight $w^{(i)}_t$ is computed:

$$w^{(i)}_t = \frac{p^{(i)}(X_t|M_t)}{\sum^{n}_{i=1} p^{(i)}(X_t|M_t)}. \quad (8)$$

- Step-5: re-sampling [14]

For $\mathbf{x}^{(i)}_{t|t-1}$, our re-sampling is to eliminate $s^{(i)}_{t|t-1}$ with small weights, and then is to concentrate and replicate on $s^{(i)}_{t|t-1}$ with large wights in order to best explain $\mathbf{m}_t$ according to their likehoods. Through such a re-sampling, $s^{(i)}_{t|t} = \{\mathbf{x}^{(i)}_{t|t}, w^{(i)}_t\}$ is obtained.

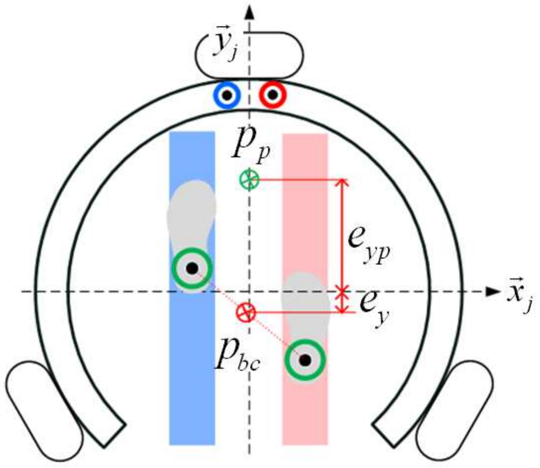


Fig. 7. Definitions and notations of the distance e_{yp} between p_{jc} and p_p in $\vec{x}_j$ and $\vec{y}_j$ directions during moving forward/backward

- Step-6: location estimate of the body

Using $\mathbf{x}^{(i)}_{t|t}$ obtained in Step-5, the location estimates of the body $\hat{\mathbf{x}}^{(i)}_{t|t}$ are calculated:

$$\hat{\mathbf{x}}_t = \frac{1}{n} \sum^{n}_{i=1} \mathbf{x}^{(i)}_{t|t}. \quad (9)$$

Then, the processes from Step-2 to Step-6 are reiterated.

C. Location Prediction

Employing the re-sampled location state in the manner of Step-5, the predicted locations of the body $\tilde{\mathbf{x}}^{(i)}_{t|t}$ are given:

$$\tilde{\mathbf{x}}_{t|t} = \begin{bmatrix} \hat{x}_{b,\ t} \\ \hat{y}_{b,\ t} + \dot{y}_b \delta t \end{bmatrix} \quad (10)$$

where $\dot{y}_b$ is obtained through (3) and δt denotes the sampling period.

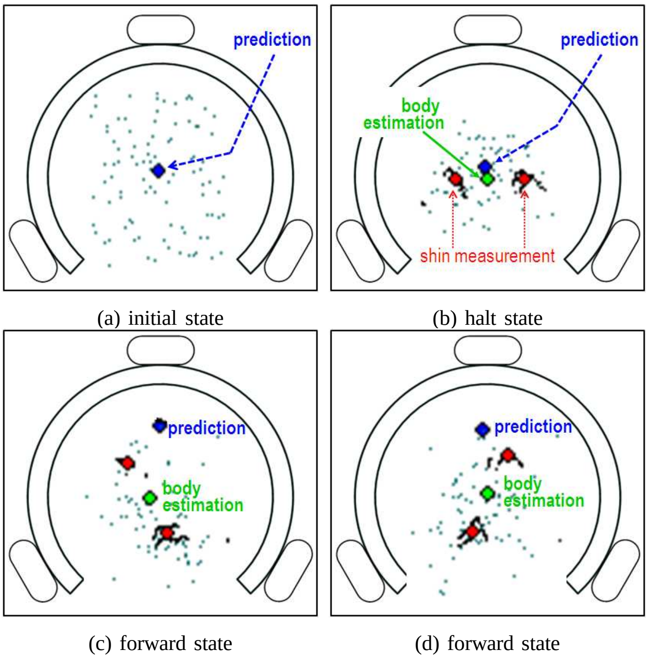
IV. Feedback Motion Control Function

The basic idea behind the proposed control is that p_{jc} and p_{bc} must remain coincident with each other. For this, a *Proportional-plus-Integral-plus-Derivative* (PID) controller is implemented. Based on the PID control technique, preliminary tests for walking behaviors of 10 people were performed and analyzed. From these results, the typical behavior pattern of moving forward/backward is modeled as shown in Fig. 7.

During moving forward/backward, separating the center position errors e_x, e_y, and e_{yp} of p_{bc} according to $\vec{x}_b$ and $\vec{y}_b$ directions with respect to $\vec{x}_j$ and $\vec{y}_j$, e_x and e_y are defined as $e_x = x_j - x_b$, $e_y = y_j - y_b$, and $e_{yp} = y_j - y_p$, respectively, that need to be minimized:

$$\begin{cases} \dot{x}_b & = & K_{p,x}\ e_x + K_{i,x} \int e_x dt + K_{d,x}\ \dot{e}_x \\ \dot{y}_b & = & K_{p,y}\ e_y + K_{i,y} \int e_y dt + K_{d,y}\ \dot{e}_y +\ e_{yp} \end{cases}, \quad (11)$$

where $\dot{x}_b$ and $\dot{y}_b$ are input velocities of JARoW in (3), and K_p, K_i, and K_d denote the proportional, integral, and derivative gains, respectively.



(a) initial state

(b) halt state

(c) forward state

(d) forward state

Fig. 8. Experimental results for mobility state identification

V. Experimental Results and Discussion

This section presents the test results for the enhanced control capability of JARoW. Our proposed control scheme is verified through extensive experiments. Parameters defined in PFIF are set as follows: $\mu_0 = (0, 0.34)$ *m*, $\sigma_0^2 = (2.5e^{-5}, 0.1e^{-5})\ m^2$, $\mu_y = 0$ *m*, and $\sigma_y^2 = (0.1e^{-5})\ m^2$. Moreover, JARoW moves with the maximum linear velocity of 4.8 *km/h*. When it makes rotational motion, the magnitude of the angular velocity is 0.5 rad/s. The gains of the PID controller in (11) are set to $K_{p,y} = K_{p,x}$ =2, $K_{i,y} = K_{d,y}$ =0.5, and $K_{i,x} = K_{d,x}$ =0.2, respectively.

To begin, we examined the effectiveness of PFIF. Fig. 8 presents the snapshots of walking process. The outputs of the interface function at each corresponding process step were taken from the developer interface in the main controller. Here the tiny points in each snapshot indicate the generated particles. Fig. 8-(a) presents the initial state where the particles are scattered in the valid region defined in Fig. 6. The test scenario is as follows: First, in Fig. 8-(b) a subject was standing still. By measuring the locations of both shins, PFIF could estimate the body location as the midpoint on the line segment connecting the locations. Then, the interface function could compute the next predicted location drawn by the blue circle. Secondly, when the subject takes a step forward as shown in Fig. 8-(c), the interface function offered the reliable estimate of the body location and the next prediction according to the outputs of shin locations. Finally, while driving JARoW forwards according to the outputs measured as shown in Fig. 8-(d), the interface function detects the changed body location.

Next, to test the validity of PFIF and the feedback motion control function, we performed a long distance test in an outdoor environment. Fig. 9 presents snapshots for the forward movements of a subject. In this test, he walked as in normal times without operation of any manual controls. Corresponding to his moving forward behaviors, JARoW autonomously controlled its feedback motions based on the estimated and predicted data. This result verifies that the control functions of the JARoW prototype equipped with LRFs work satisfactorily under our laboratory conditions.

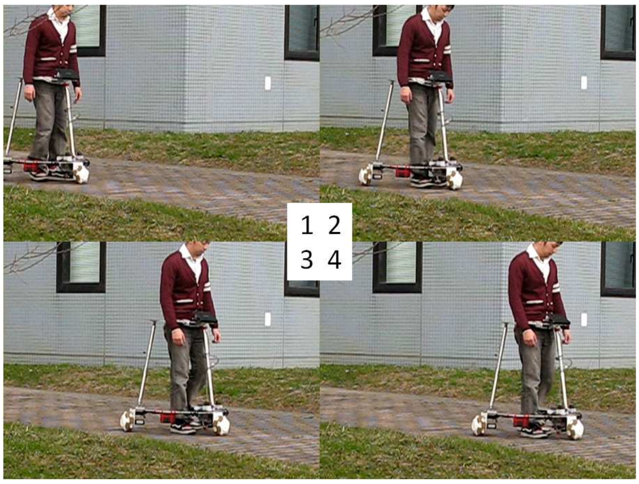
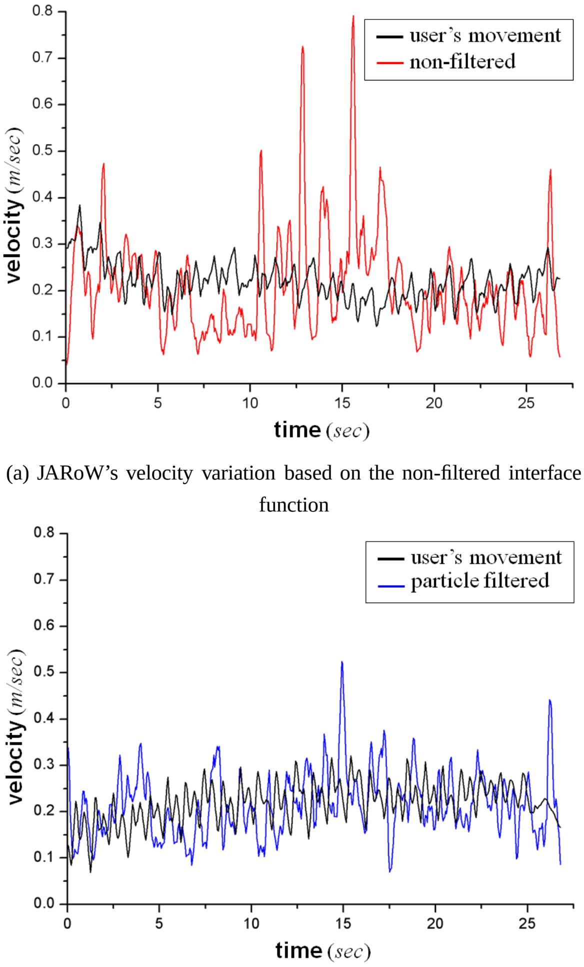


Fig. 9. Experimental results for the forward movement in an outdoor environment

Finally, to investigate the performances of JARoW's PFIF, the moving forward motions employing the PID controller are performed. As a subject took steps forward for 30 seconds while keeping up a steady pace, JARoW followed his walking behaviors. Figs. 10-(a) and (b) show the results of JARoW's velocity variations based on the no-filtered function and PFIF, respectively. Here, the black bold solid lines indicate user's walking velocities, and the red and blue lines present the variation of JARoW's velocity by non-filtered function and PFIF, respectively. Compared with Fig. 10-(a), JARoW's velocity variation in Fig. 10-(b) became flattened and JARoW attempted to follow the subject's walking velocity as closely as possible. Specifically, the time period between 5 second and 13 second in Fig. 10-(a) indicates unstable velocity responses against the subject's walking. Moreover, Fig. 11 presents statistic data to assist the understanding of the results in Figs. 10-(a) and (b). Although overshoots were observed under the proposed control scheme, the lower peak value and frequency were investigated. Clearly, compared with the non-filtered result, the particle filtered result shows JARoW could generate the more reliable moving forward motion according to the actual user's behavior. More notably, the proposed feedback control allows users to easily control JARoW without requiring any mental or physical efforts. In contrast to existing active walkers, our walker features simple structure and compact size that can be fit into our everyday environment.

VI. Conclusions

This paper presented the enhanced control scheme for JARoW that is easy for potential users to handle and trans-



(a) JARoW's velocity variation based on the non-filtered interface function

(b) JARoW's velocity variation based on PFIF

Fig. 10. Comparison results between non-filtered interface function and PFIF

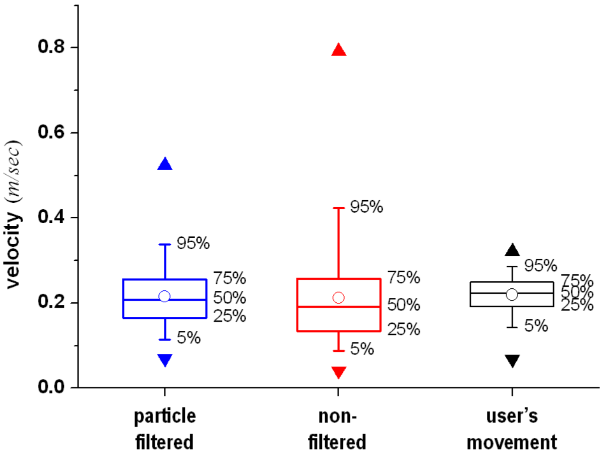


Fig. 11. Statistic data for the comparison results in Figs. 10-(a) and (b)

port. First, we proposed a natural user interface without requiring any user operations, through the use of a pair of LRFs. Secondly, the user's location estimation and prediction scheme employing a particle filter was developed to offer the natural motions of JARoW according to the user walking motions. Thirdly, the feedback motion control adjusted the motions of JARoW corresponding to the actual user walking behaviors. To demonstrate the effectiveness of the proposed control functions, different types of experiments were performed and the results were quite encouraging. When considering the elderly who tend to lean their upper body onto the walker, as our future study, more sophisticated control algorithms will be needed to cope with dynamic and unpredictable changes in the nominal walker parameters.

References

[1] H. Ikeda, Y. Katsumata, M. Shoji, T. Takahashi, and E. Nakano, "Cooperative strategy for a wheelchair and a robot to climb and descend a step," *Advanced Robotics*, 22(13-14):1439-1460, 2008.

[2] C.-H. Kuo, H.-W. Yeh, and C.-E. Wu, "Development of autonomous navigation robotic wheelchairs using programmable system-on-chip based distributed computing architecture," *Proc. IEEE Int. Conf. Systems, Man and Cybernetics*, 2939-2944, 2007.

[3] H. Yu, M. Spenko, and S. Dubowsky, "An adaptive shared control system for an intelligent mobility aid for the elderly," *Autonomous Robotics*, 15(1):53-66, 2003.

[4] A. Veg and D. B. Popovic, "Walkaround: mobile balance support for therapy of walking," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 16(3):264-269, 2008.

[5] Y. Hirata, A. Hara, and K. Kosuge, "Motion control of passive intelligent walker using servo brakes," *IEEE Transactions on Robotics*, 23(5):981-990, 2007.

[6] H. Kobayashi, T. Karato, and T. Tsuji, "Development of an active walker as a new orthosis," *Proc. IEEE Int. Conf. Mechatronics and Automation*, 186-191, 2007.

[7] K. Kong and D. Jeon, "Design and control of an exoskeleton for the elderly and patients," *IEEE/ASME Transactions on Mechatronics*, 11(4):428-432, 2006.

[8] V. Kulyukin, "Human-robot interaction through gesture-free spoken dialogue," *Autonomous Robots*, 16(3):239-257, 2004.

[9] H. Yano, Y. Hosomi, K. Aoki, R. Nanba, T. Hirotomi, and Satoru Okamoto, "Unstable motion detect system for four-casterd walker," *IEICE technical report, Speech*, 107(434):13-18, 2008.

[10] M. Hirasawa, H. Okada, and M. Shimojo "The development of the plantar pressure sensor shoes for gait analysis," *Journal of Robotics and Mechatronics*, 20(3):324-330, 2007.

[11] G. Lee, T. Ohnuma, and N. Y. Chong, "Design and control of JAIST active robotic walker," *Journal of Intelligent Service Robotics*, 3(3):125-135, 2010.

[12] J. M. Burnfield and C. M. Powers, "Normal and pathologic gait" in *Orthopaedic physical therapy secrets*, J. D. Placzek and D. A. Boyce (edt.), Hanley and Belfus (2nd ed.), 2006.

[13] S. Chib and E. Greenberg, "Understanding the Metropolis-Hastings algorithm," American Statistician, 49(4):327-335, 1995.

[14] G. Kitagawa, "Monte Carlo filtering and smoothing method for nonlinear non-Gaussian state space models,"Journal of Computational Graphical Statistics, 5(1):1-25, 1996.

[15] C. M. Bishop, *Pattern Recognition and Machine Learning*, Springer, 2006.