Automatic Speed Control for SmartWalker

Jiwon Shin* Ivo Steinmann* Bertrand Meyer*†

*Chair of Software Engineering, Department of Computer Science, ETH Zürich, Switzerland
†Software Engineering Lab, Innopolis University, Kazan, Russia
jiwon.shin@inf.ethz.ch, ivost@student.ethz.ch, bertrand.meyer@inf.ethz.ch

ABSTRACT

Ensuring mobility of the elderly is an important task in our aging society. To this end, this paper presents an automatic speed controller for the SmartWalker – a high-tech extension of a regular walker. The walker locates its user by detecting the user's legs using a laser range scanner. The controller then determines the optimal speed for the walker using the user's location and other sensory data. We evaluated the walker and its speed controller with thirteen residents at three different retirement homes. Our analysis showed that the walker with the controller is slightly more comfortable and easier to maneuver than the walker without the controller and is more liked than traditional walkers.

Categories and Subject Descriptors

H.5.2 [**Information Interfaces and Presentation**]: User Interfaces—*Input devices and strategies, Interaction styles*; I.2.9 [**Artificial Intelligence**]: Robotics

General Terms

Experimentation, Human Factors

Keywords

Ambient assisted living, Robotics

1. INTRODUCTION

As our society ages, ensuring mobility of the elderly gains greater importance. Impaired mobility can decrease independence and life quality and increase institutionalization and mortality of the elderly [11]. Mobility aids can delay mobility impairment, thus prolonging one's independence and ensuring the quality of life. Among various mobility aids – from canes to wheelchairs – rollators, or wheeled walkers, are particularly well-liked because they support natural gait patterns and are easy to use [7].

PETRA '15 July 01 - 03, 2015, Corfu, Greece

ACM 978-1-4503-3452-5/15/07 ...$15.00.
DOI: http://dx.doi.org/10.1145/2769493.2769565.

Smart walkers are rollators equipped with sensors and actuators. They provide better assistance and support to make it easier for the elderly to stay active by offering physical support, sensorial assistance, cognitive assistance, or health monitoring [7]. They can be passive devices, which may steer or brake automatically but require the user to push them to move forward, or active devices, which can actively control the movement. Active devices are of particular interest because they would ease the usage, resulting in an increase in acceptance of the device by the elderly [2]. This paper focuses on an active device that controls its speed based on the leg movements and ground inclination.

Several robotic walkers perform gait analysis for control using Hokuyo [6, 8], Kinect [4], force sensor [3], and/or IMU [10] as the sensor. Among these, JAIST Active Robotic Walker [6] and UFES Smart Walker [10] are particularly relevant. Equipped with two Hokuyo laser sensors, the JAIST walker tracks the legs by Kalman filter and controls the three motorized wheels of its circular walker based on the user's location with respect to the walker. The UFES walker detects the legs using data from a Hokuyo and measures the ground pitch and roll using an IMU. Their controller then combines these data to control the robot on inclined surface.

This paper proposes an automatic speed controller for the SmartWalker. As a natural user interface, the walker supports its user without requiring the user to push the walker. Similar to the aforementioned walkers, the SmartWalker also detects the legs and controls the speed according to the walking speed and the ground inclination, but it achieves this using much more affordable sensors. The walker computes the user's walking speed by detecting the user's leg movements using a laser range scanner and then combines this information with the ground inclination and the state of its brakes in the controller to compute the appropriate speed for the walker. We evaluate the walker with thirteen residents of three different retirement homes.

2. SMARTWALKER

SmartWalker [9] consists of a walker equipped with sensors and actuators and software that controls the walker. The walker that consists of a normal walking frame, two hub engines, a laser range scanner, an inclinometer, and a rotatable camera (Figure 1(a)). Located at the two rear wheels (Figure 1(b)), the hub engines contain a hall effect sensor for measuring the rotational speed of the wheel. The laser range scanner at the bottom center of the walker is a low-cost scanner[1] harvested from Neato XV-11 vacuum

[1]400 CHF/USD for the vacuum cleaner

(a) SmartWalker (b) Hub engines (c) Laser scanner

Figure 1: SmartWalker hardware.

cleaner (Figure 1(c)). It scans 360° at 1° resolution with the speed of 250 ms per one 360° scan. On top of the scanner is a Pewatron PEI-Z100-AL-232-1 360° inclinometer[2] that measures the pitch of the ground. For 3D sensing, the walker has a PrimeSense Carmine 1.08 sensor, mounted on a small servo motor below the handlebar.

Main processing units are a BeagleBone motherboard and a tablet computer, both running Ubuntu. The BeagleBone receives sensory data and controls actuators, and the tablet processes computationally intensive algorithms and acts as a display for the user interface. The key software components are a leg detection package for extracting the user's position and a controller for controlling the wheel speed, and both run on the BeagleBone.

The walker operates in two modes – assistive and autonomous. In the assistive mode, the walker functions as a smart mobility aid and supports its user when walking. In the autonomous mode, the walker functions as an autonomous device and navigates around its environment. The automatic speed controller is part of the assistive mode.

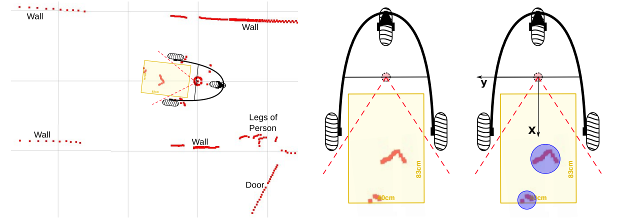
3. LEG DETECTION

The goal of leg detection is to locate the user so that the walker can keep a steady distance away from the user when the user is walking and stop when no user is detected. The leg detection takes laser scan data as input and searches for two clusters that represent the two legs. Treating the center of the two legs as the user's position, we calculate the user's distance to the walker and feed this distance and the mean distance over 90 seconds into the controller as input.

The laser scanner scans the area around the walker and provides 360° scans, ranging between $0.02m$ and $4m$ (Figure 2(a)). The leg detection, however, only needs a subset of the data that falls into the area where people could be when walking behind the walker. Similar to the tracking algorithm that defines a search area in front of the robot [5], we define a walking area behind the walker and filter out the points that lay outside of this area (Figure 2(b)); the area is set to $40cm$ by $83cm$ based on our analysis of the walking patterns of twenty adults. From the filtered data, the algorithm then detects two legs using an expectation-maximization (EM) algorithm [1] (Figure 2(c)).

The EM algorithm is an iterative method for finding the parameters θ that maximize the log likelihood of the observed data $\mathbf{X}$ without knowing their labels $\mathbf{Z}$. In the leg detection, θ are the means and variances of the legs, $\mathbf{X}$ are

[2] 200 CHF/USD



(a) A sample 360° scan (b) Filtered (c) Detected

Figure 2: Laser scan data (red) for leg detection with the walking area (yellow) and legs (blue).

the filtered data, and $\mathbf{Z}$ are the leg labels – *left* and *right*. Given initial guesses $\theta^{(0)}$, the algorithm repeats the *E-step* of finding the expectation $Q(\theta, \theta^{(t)}) = \mathrm{E}_{\mathbf{Z}|\mathbf{X},\theta^{(t)}}[\log p(\mathbf{X}, \mathbf{Z}|\theta)]$ using the current parameters $\theta^{(t)}$ and the *M-step* of computing new parameters $\theta^{(t+1)} = \arg\max_\theta Q(\theta, \theta^{(t)})$. The iteration continues until it converges or reaches the maximum number of iterations.

With the assumption that only one person who has two legs is present behind the walker, the leg detection algorithm searches for two clusters. As the initial guesses $\theta^{(0)}$, it takes either the leg positions of the previous data, or if this information is too old or unavailable, predefined initial leg positions. It iteratively searches for the cluster means and variances until termination. To avoid flickering of data points between the two clusters, we introduce a threshold as an additional termination criterion so that the algorithm terminates if the change between two consecutive iterations is below the threshold.

Once the clusters are found, the algorithm validates the clusters based on their size and if the validation passes, it computes the distance of the user with respect to the laser scanner. The validation step discards any cluster that is too small (fewer than 7 data points) or too big (more than 35 data points) to be a leg. If the validation is successful, the cluster with a larger y component is assumed to be the left leg and the other is assumed to be the right leg. From the two leg cluster centers $\mathbf{p}_l$ and $\mathbf{p}_r$, we compute the user's body center $\mathbf{c} = \frac{\mathbf{p}_l + \mathbf{p}_r}{2}$ as the mean of the two and the user's distance d to the walker as the Euclidean distance $d = \sqrt{\mathbf{c}_x^2 + \mathbf{c}_y^2}$. In addition, we compute the average distance $\widetilde{d}$ over the past 90 seconds to determine the user's position with respect to the recent history.

4. CONTROL

The controller takes various sensory information as input and controls the speed of the wheels. The controller consists of a wheel controller and a power controller. The wheel controller is a safety authority between an active controller mode and the motor driver. In addition to setting the driver to the right power, the wheel controller stops the engines if it does not receive messages regularly. This automatic stoppage prevents the wheels from turning continuously when a parent controller hangs or a message does not get delivered due to an interruption in the network connection.

The power controller takes the speed of the walker, the inclination of the road, the state of the brakes, and the distance of the user as input and adjusts the engine speed accordingly. The power controller stops the engines if any of the sensors fails to deliver data or the leg detection does not detect anyone behind the walker. Otherwise, the power for

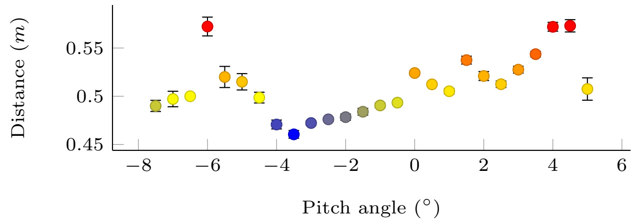


Figure 3: Ground inclination vs. walking distance.

the wheels are computed as a combination of the sensory values. The power for the left wheel is

$$p_l = p_{s_l} + p_{b_l} + p_i + p_d,$$

a sum of the power due to the the left wheel speed p_{s_l}, the left brake p_{b_l}, the inclination p_i, and the distance to the user p_d. Similarly, the power for the right wheel is set to

$$p_r = p_{s_r} + p_{b_r} + p_i + p_d.$$

The four components are computed as follows: The speed components are proportional to the wheel speeds v_l and v_r and set to $p_{s_l} = k_v \cdot v_l$ for the left wheel and $p_{s_r} = k_v \cdot v_r$ for the right wheel. The brake components are proportional to the brake states b_l and b_r and inversely proportional to v_l and v_r so that they act in the opposite direction of motion. They are set to $p_{b_l} = k_b \cdot b_l \cdot -v_l$ for the left brake and $p_{b_r} = k_b \cdot b_r \cdot -v_r$ for the right brake. The inclination component depends on the pitch angle α_{pitch} and the walker's speed $v = \frac{v_l + v_r}{2}$. It is set to

$$p_i = |v| \cdot \sin(\alpha_{pitch}) \cdot k_{ascend}$$

for forward uphill ($v > 0 \wedge \alpha_{pitch} > 0$) or backward downhill movement ($v < 0 \wedge \alpha_{pitch} < 0$) and to

$$p_i = |v| \cdot \sin(\alpha_{pitch}) \cdot k_{descend}$$

otherwise. Lastly, the distance component depends on the speed v, the difference in distance $\Delta d = \widetilde{d} - d$ between the current distance d and the mean distance $\widetilde{d}$, and the pitch angle α_{pitch}. It is set to

$$p_d = k_d \cdot v \cdot \begin{cases} -\Delta d & \text{if } \alpha_{pitch} > \gamma \\ 0 & \text{if } \alpha_{pitch} < -\gamma \wedge \Delta d \geq 0 \\ \Delta d & \text{otherwise.} \end{cases}$$

The coefficients are initialized to $k_v = 6.1$, $k_{ascend} = 114.0$, $k_{descend} = 40.0$, $k_b = 6.6$, and $k_d = 5.1$.

As the terrain is almost never perfectly flat, we introduce a pitch threshold $\gamma = 3$ and consider any terrain with $|\alpha_{pitch}| < \gamma$ as flat ground. Knowing the terrain is important because the user's distance to the walker depends on the terrain (Figure 3). On uphill ($\alpha_{pitch} > \gamma$), the distance to the walker d is longer than on flat terrain. In turn, Δd is negative, and the resulting p_d is also negative, meaning that the walker's support would be reduced. On uphill, however, the walker should provide more support. Therefore, Δd is negated. On downhill ($\alpha_{pitch} < \gamma$), the distance to the walker is shorter than on flat terrain, resulting in a positive Δd. This causes the walker to accelerate, which is dangerous. Therefore, p_d is set to zero.

After the computation of p_l and p_r, the two power values are set as power metric to the wheel controller. To prevent the engines from turning on at slow speed, the values are set to zero when they are below p_{thresh}. In addition, to avoid high frequency changes, the new power values to the wheel controller remain unchanged if the change between two consecutive values is below p_{delta}. The thresholds are initialized to $p_{thresh} = 0.05$ and $p_{delta} = 0.01$. All coefficients and thresholds are experimentally determined and are dynamically adjustable.

5. EVALUATION

We evaluated the SmartWalker at three different retirement homes in Zürich, Switzerland to better understand the effect of the automatic speed control on the elderly's acceptance of the walker. The evaluation was divided into three parts: background information, evaluation of the walker without the controller, and evaluation of the walker with the automatic speed controller. The information was gathered using a questionnaire that had multiple choice questions. Given the participants' limited motor skills and vision, every question and possible answers were read out to the participants. An evaluation took on average 20 minutes per participant. In total, thirteen elderly residents participated in the study, and the information was gathered in German.

Of the 13 participants, six were men, and seven were women. Four were visually-impaired or blind. Seven were aged between 80 and 89, four were 90 or over, and two were between 70 and 79. All participants used either a wheeled walker (10) or a cane (3). The frequency of the usage ranged from daily (10) and four to six times a week (2) to less than once a week (1). Some went outside daily (5) or four to six times a week (3), but others mostly stayed inside and did not go outside (5). Most people (8) were unfamiliar with computers, smart phones, or other technological devices, but some were daily (4) or frequent (1) users of such devices.

After answering the first part of questions, the participant walked around the premise of the retirement homes with the walker. For this portion, the speed controller was turned off, and therefore, the participants felt the full weight and resistance of the walker. Given that not everyone is in equal physical shape, we did not define an exact course to follow; instead, each participant decided for him-/herself the distance and duration of the walk. After the walk, most participants said that the walker is heavy (5) or too heavy (4) and too big (8) or big (2). Only a minority of people said that the walker's weight is comfortable (4) and its size is good (3). Interestingly, some participants found the walker's heaviness to be an advantage because they felt that it provided them additional security and stability.

In the third part, the participants walked around with the SmartWalkeronce again but with the controller turned on. There were some minor changes in their responses. In terms of the level of comfort in walking, most found the walker comfortable (6) or very comfortable (4) to walk with the

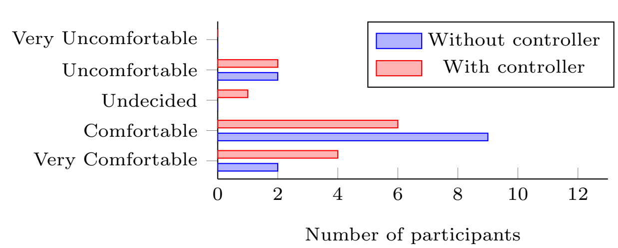


Figure 4: Walking quality.

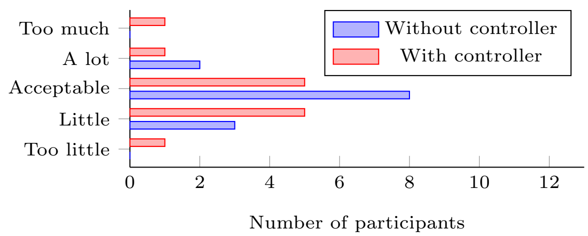


Figure 5: Required effort for pushing the walker.

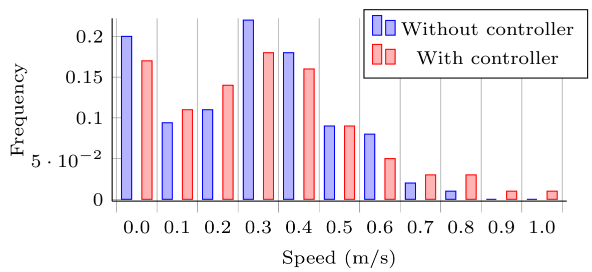


Figure 6: Walking speed.

controller, which is similar to the level of comfort they felt when walking without it (Figure 4). In terms of the pushing effort, most said that the amount of effort required to push the walker is acceptable (8) or little (3) when walking without the controller. Their response was more evenly distributed between acceptable (5) or little (5) for walking with the controller (Figure 5). On average, the participants found the walker with the controller slightly more comfortable and easier to manipulate; only one participant, who relies heavily on the walker for support, stated that the speed controller made him feel less stable.

The change in walking speed between walking without and with the controller was more noticeable (Figure 6). People walked slightly faster when the walker was automatically adjusting its speed. This does not necessarily mean that the controller made them suddenly move faster, as rehabilitation devices may do to encourage recovery. It could also be because the walker without the controller is heavy and thus they may have walked slower than their usual speed. Further study is needed to better understand the phenomenon.

Overall, the participants' impression of the walker with the controller was positive. Eleven said that the automatically adjusted speed of the walker is good; only two said that it is fast (1) or too fast (1). Seven participants stated that they prefer the SmartWalkerwith its controller while four preferred their current mobility aid and two were undecided. Most participants were very interested in our project, and some even wanted to know the approximate price of the device and when the prototype would be ready for purchase. The most frequent complaints were its weight, maneuverability, and width. In particular, several stated that its width is not suitable for small elevators and doors.

6. CONCLUSION

This paper presented an automatic speed controller for the SmartWalker and evaluated it with thirteen residents of three different retirement homes. Our study showed that people find the walker with the speed controller slightly more comfortable, slightly easier to maneuver, and more attractive to own. Given that the study was conducted with a small group of people in retirement homes, it is difficult to generalize the findings. We are thus interested in conducting an in-depth study, involving more elderly living in different situations.

Acknowledgments

We thank Marcel Mathis for the hardware support. This work was partially supported by the European Research Council under the European Unions Seventh Framework Programme (ERC Grant agreement no. 291389) and the Hasler Foundation under the SmartWorld programme.

7. REFERENCES

[1] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *J. of the Royal Statistical Society. Series B (methodological)*, pages 1–38, 1977.

[2] P. Flandorfer. Population ageing and socially assitive robots for elderly persons: The importance of sociodemographic factors for user acceptance. *Interantional Jouranl of Population Research*, 2012.

[3] S. L. Grondin and Q. Li. Intelligent control of a smart walker and its performance evaluation. In *2013 IEEE Int. Conf. on Rehabilitation Robotics*, 2013.

[4] C. Joly, C. Dune, P. Gorce, and P. Rives. Feet and legs tracking using a smart rollator equipped with a kinect. In *Workshop on "Assistance and Service Robotics in a Human Environment" in conjunction with IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2013.

[5] H. Kim, W. Chung, and Y. Yoo. Detection and tracking of human legs for a mobile service robot. In *2010 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics (AIM)*, pages 812–817, 2010.

[6] G. Lee, T. Ohnuma, N. Y. Chong, and S.-G. Lee. Walking intent-based movement control for jaist active robotic walker. *IEEE Trans. on Systems, Man, and Cybernatics: Systems*, 44(5):665–672, May 2014.

[7] M. M. Martins, C. P. Santos, A. Frizera-Neto, and R. Ceres. Assistive mobility devices focusing on smart walkers: Classification and review. *Robotics and Autonomous Systems*, 60(4):548–562, 2012.

[8] X. S. Papageorgiou, C. S. Tzafestas, P. Maragos, G. Pavlakos, G. Chalvatzaki, G. Moustris, I. Kokkinos, A. Peer, B. Stanczyk, E.-S. Fotinea, et al. Advances in intelligent mobility assistance robot integrating multimodal sensory processing. In *Universal Access in Human-Computer Interaction. Aging and Assistive Environments*, pages 692–703. Springer, 2014.

[9] J. Shin, D. Itten, A. Rusakov, and B. Meyer. Smartwalker: Towards an intelligent robotic walker for the elderly. In *The 11th Int. Conf. on Intelligent Environments*. IEEE, 2015. To appear.

[10] L. Tausel, C. A. Cifuentes, C. Rodriguez, A. Frizera, and T. Bastos. Human-walker interaction on slopes based on lrf and imu sensors. In *2014 5th IEEE RAS & EMBS Int. Conf. on Biomedical Robotics and Biomechatronics*, pages 227–232. IEEE, 2014.

[11] H. A. Yeom, J. Fleury, and C. Keller. Risk factors for mobility limitation in community-dwelling older adults: a social ecological perspective. *Geriatric Nursing*, 29(2):133–140, 2008.