Journal of Ambient Intelligence and Smart Environments 1 (2016) 1–5
IOS Press

SmartWalker: an intelligent robotic walker

Jiwon Shin [a], Andrey Rusakov [a] and Bertrand Meyer [a,b,c]
[a] *Chair of Software Engineering, Department of Computer Science, ETH Zürich, Switzerland*
[b] *Department of Electronics, Information and Bioengineering, Politecnico di Milano, Italy*
[c] *Software Engineering Lab, Innopolis University, Kazan, Russia*
E-mail: jiwon.shin@inf.ethz.ch, andrey.rusakov@inf.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 SMARTWALKER, a high-tech extension of a regular walker that aims to navigate around its environment autonomously and assist its user intelligently. The walker is equipped with sensors and actuators and operates in two modes, autonomous and assistive. In the autonomous mode, the walker accepts gesture commands via its gesture-based interface and navigates around accordingly. The interface uses a k-nearest neighbors classifier with dynamic time warping to recognize gestures and the Viola and Jones face detector to locate the user. In the assistive mode, its automatic speed controller determines the optimal speed for the walker. The walker locates its user by detecting the user's legs using a laser range scanner and combines the information with other sensory data for the speed control. The walker was evaluated with residents of retirement homes in Zürich. The study aimed to understand the usefulness of robotic walker to the elderly, evaluate its potential to replace traditional walkers, and determine the appropriateness of the added functionality. The elderly found the SMARTWALKER's autonomy useful and exciting and the walker with the controller slightly more comfortable and easier to maneuver. They stated that the walker is too big and too heavy but liked it more than traditional walkers in the assistive mode.

Keywords: ambient assisted living, robotics, user study

1. Introduction

The percentage of the global population aged 60 years or over has been increasing steadily, and it is projected to rise further to 21 percent by 2050 [33]. As the population ages, it becomes increasingly important that people continue to stay active and mobile. Impaired mobility of older adults is linked to a loss of independence, decreased quality of life, institutionalization, and a higher risk of mortality. Unfortunately, impaired mobility is prevalent in 44% of older adults [35]. Thus, ensuring mobility of the elderly is critical to our aging society.

Mobility aids range from simple external devices such as canes to mobile vehicles such as wheelchairs. Canes can reduce falls in patients by increasing gait stability, but they provide minimal weight support. Wheelchairs can transport people who cannot move by themselves, but excessive sitting can cause deterioration of health. In between the two lie walkers. Walkers provide support for weight and balance but require patients to use their own locomotion, thus minimizing deterioration of mobility. Rollators, or wheeled walkers, are particularly liked for their usability and support for natural gait patterns [22].

Advancement of technology has given rise to smart walkers – rollators equipped with sensors and actuators for better assistance and support. Most research on smart walkers has been in providing better physical support, sensory assistance, cognitive assistance, health monitoring, or human-machine interface [22]. They can be passive or active. Passive devices may steer or brake automatically but require the user to push them to move forward while active ones 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 [12]. Beyond passive or active support, little attention has been given to smart walker as an autonomous robotic device. In certain situations, however, a non-autonomous walker becomes clumsy to use. For instance, when many walker users gather in

Fig. 1. A resident of a retirement home testing the SMARTWALKER.

one space for a meeting or a meal, placing the walkers without hindering the social situation becomes difficult. Indeed, at retirement homes, where many residents use walkers, each walker has to be moved out of a room when residents gather for a meal so that they can eat more comfortably; at the end of the meal, the walkers are brought back to the room one at a time. An autonomous walker with the ability to park itself and return to its user when required would eliminate such a repetitive and laborious task.

This paper proposes SMARTWALKER that functions as an autonomous robot as well as an assistive device. As a high-tech extension of a regular walker, it is equipped with sensors and actuator. In the autonomous mode, the SMARTWALKER can receive user commands through its real-time gesture-based user interface and navigate around its environment accordingly. The underlying algorithm uses a k-nearest neighbors (k-NN) classifier with dynamic time warping (DTW) to classify gestures. As people can unintentionally make gestures similar to the commands, the interface ignores unintentional gestures by detecting the user's attention along with the gesture recognition. In the assistive mode, the walker supports its user without requiring the user to push the walker by detecting the legs and controls the speed according to the walking speed and the ground inclination. 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. The walker's autonomous mode was evaluated with 23 residents and eight members of the staff at five different retirement homes and the assistive mode with thirteen residents of three different retirement homes in Zürich, Switzerland. In the study, the elderly found the gesture-based interface difficult to use but appreciated the speed controller.

The aims of this research are to understand 1) whether or not a robotic walker an attractive alternative to a traditional walker; 2) if gesture-based interface an acceptable form of user interface for the elderly; and 3) if the ability to move autonomously and the motorized support are useful functionalities. This paper extends the authors' previous work on gesture-based interface for the SMARTWALKER robot [29] with the work on automatic speed control [30].

2. Related work

The first publication on a smart mobility aid was a personal adaptive mobility aid for the infirm and elderly blind (PAM-AID) in 1998 [18]. Since then, researchers have proposed many different smart walkers. Most are purely assistive, offering physical support, sensorial assistance, cognitive assistance, or health monitoring [22]. Few have an additional capability to move around autonomously. Notable work is by Glover *et al.* [15], whose walker can park itself and return to its user according to the user's control via a remote button. This paper presents SMARTWALKER that has two modes of operation. In the autonomous mode, it achieves autonomy without requiring an external controller via gesture-based user interface. The assistive mode focuses on enhanced support and natural interaction via leg detection and tracking.

Smart walker interfaces range from direct interfaces such as a joystick [16] and modified handlebars [21] to indirect interfaces such as a gait detection system [20]. Direct interfaces that do not require any physical contact are particularly relevant to the research presented here. Gharieb [13] proposed a voice-based interface for visually-impaired people that enables its user to command the walker verbally, thus eliminating the necessity for physical contact. Similarly, the gesture-based interface used in this work receives commands from its user located at a distance.

Gesture recognition is a well-researched area within the field of computer vision for robotics. Gleeson *et al.* [14] presented a gestural communication lexicon for human-robot collaboration in assembly tasks. Barattini *et al.* [2] proposed a gesture set for the control of industrial collaborative robots. In service robotics, recognition of pointing and showing gestures for a domestic service robot was studied by Droeschel *et al.* [10]. However, to the authors' knowledge, no work has investigated the viability of a gesture-based interface for smart walkers.

The proposed gesture-based interface uses an RGBD camera as the input sensor. Using an RGBD camera for image processing has gained momentum since the release of affordable devices. In terms of hand gesture recognition, Suarez and Murphy [31] categorize different techniques that use RGBD data as input. Work of particular relevance is a probability-based dynamic time warping approach for gesture recognition using RGBD data [3]. The authors employ a Gaussian Mixture Model to model the variance within a training set. The proposed work also uses dynamic time warping on RGBD data, but does not explicitly model the variance; instead, *k*-NN is used to classify gestures. In this way, current work is more similar to Ten Holt *et al.* [32], who expand dynamic time warping to a multi-dimensional space and classify gestures using *k*-NN. The recognition system described in this paper computes the warping distance from three-dimensional feature vectors with *k*-NN.

The interface also has face detection and recognition. The face detector ensures that the system classifies gestures only if the user is looking at the walker whereas the face recognizer ensures that the walker accepts commands only from its owner. The face detection approach is similar to Monajjemi *et al.* [24] in that it computes a face score for detected faces and accepts only those faces with face scores higher than a threshold as frontal faces. The face recognition uses Local Binary Patterns Histograms (LBPH) Face Recognizer [1].

The assistive mode detects and tracks legs to control the walker's speed. Several researchers have proposed leg detection and tracking methods for robotic systems. Kim, Chung, and Yoo [17] propose a human leg detection and tracking method for a mobile robot. In the same way, presented work also filters out data that are outside of the detection area and validate potential leg clusters based on their size. Cifuentes *et al.* [7] present gait detection method that combines tracking data from a laser range finder with data from a wearable internal measurement unit. Current work also takes data from an inclinometer as input to the control loop, but as the sensor is integrated into the walker, it does not require the user to wear any external gadget.

Gait analysis has also been used for rehabilitation. The UFES walker [11] has a three-dimensional force sensor at the handlebars and a laser scanner between the wheels and provides user-walker interaction data for generating navigation commands. The data are then used for clinical analysis and for fine-tuning of different walking training and rehabilitation programs. The smart walker of Postolache *et al.* [26] is designed to assist in physiotherapy sessions for gait analysis and recovery. They use handlebars with piezo-resistive force sensors for measuring the force applied by the user to the walker.

Fig. 2. SMARTWALKER.

The walker of particular relevance is the JAIST Active Robotic Walker [19,20]. The walker tracks its user's leg movement using Kalman filter and controls its velocity accordingly. SMARTWALKER presented here also detects the legs by EM clustering and uses this information and other sensory data to control the walker's speed and direction. Moreover, a modified standard walking frame is used instead of a circular frame of the JAIST walker with three motorized wheels. The modified frame has two back wheels which are controlled as a differential drive.

3. SmartWalker

The SMARTWALKER is composed of a walker frame enhanced with sensors, actuators, and appropriate software to control these hardware components.

3.1. Hardware

The walker consists of a normal walking frame, two hub engines, a laser range scanner, an inclinometer, and a rotatable camera (Figure 2). The front wheel has no motor and is for stability and maneuverability. The rear wheels are powered with e-bike motors and can be controlled to move the walker as desired. Located at the two rear wheels, 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

[1] 400 CHF/USD for the vacuum cleaner

Fig. 3. SMARTWALKERś tablet-PC running a graphical application with a touch user interface.

Neato XV-11 vacuum cleaner[2] and intended for obstacle avoidance. 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[3] that measures the pitch of the ground. The walker also has a PrimeSense Carmine 1.08 RGBD camera[4], which is attached to a motor for 360° rotation and placed below the handlebar, for gesture recognition. It has operation range of 0.8 to 3.5 meters and works indoors only.

The SMARTWALKER uses two processing units connected via the local network. The first one is a tablet-PC, where most of the computation is run. The tablet also provides the user with graphical applications and a touch interface (Figure 3). The second device is a BeagleBone Black[5] single-board computer for message passing between the tablet and the sensors and the actuators.

3.2. Software

The SMARTWALKER software is distributed between the tablet-PC and the single-board computer (Figure 4). The tablet runs the "brain" of the system – the main control application and the gesture recognition module – whereas the single-board computer receives raw data from the sensors and forwards commands from the tablet to the actuators. The SMARTWALKER control application is written in Roboscoop [28], a robotics programming framework with concurrency support. Built on Simple Concurrent Object-Oriented Programming (SCOOP) model, which provides simple and safe concurrency features, Roboscoop is a robotics library written in Eiffel [23]

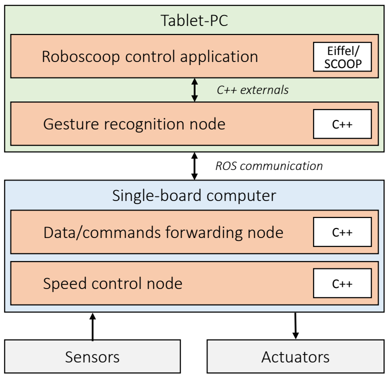


Fig. 4. SMARTWALKER software distribution: orange rectangles show the executable nodes, arrows specify the data flow direction.

and has tools for integration with other frameworks and libraries via C/C++ external interface. The gesture and face recognition are implemented in C++ as nodes in Robot Operating System (ROS) [27]. ROS is a popular middleware in robotics and in addition to network communication, it provides libraries for image processing among others.

3.3. Modes of operation

The SMARTWALKER can function in two different modes: assistive mode and autonomous mode. In the assistive mode, the frame is driven by the user and it provides support to the user during the movement. In addition to regular walkers' functionality, the SMARTWALKER minimizes a pushing effort by automatically keeping the optimal speed. A slight push from the user is still required, but the robot carries its own weight with support of the motors. This becomes very helpful when going uphill with a walker or having some load in a walker's basket. In the autonomous mode, the SMARTWALKER operates as an autonomous robot, without any physical exertion by the user. It navigates around its environment and executes the user's commands given through gestures. Switching between the two modes requires just a touch of a button in the graphical touch interface on the tablet-PC. The user can also activate the autonomous mode by giving a gesture command.

4. Autonomous mode

The gesture-based interface takes images from the RGBD camera as input and processes the information in three steps: gesture detection, face detection, and

[2] http://www.neatorobotics.com/
[3] 200 CHF/USD
[4] Discontinued
[5] http://beagleboard.org/bone

face recognition. The gesture detection accepts predefined gestures as commands and rejects unknown gestures. The face detection localizes the user's position more accurately and ensures that the commands are processed only when the user intends to, i.e., when the user is looking at the walker. Finally, the face recognition ensures that the walker only responds to the commands given by its owner. The face detection and recognition modules are independent of the gesture detection module and can be individually deactivated through the graphical touch interface.

4.1. Commands

The gesture detector works with three different predefined gestures and interprets them as commands – "come here", "go back", and "stop". In addition, the detector gives voice feedback to the user when a gesture is detected. The "come here" command requires the user to move a hand up and down, and when detected, the SMARTWALKER says "I am coming to you" and moves to the user with its handle bars towards the user. The "go back" gesture requires moving a hand sideways, either to left then right or to right then left, and the robot says "Going back to the charging station" and moves back to a predefined location. The "stop" command requires the hand to push forward, and it causes the robot to beep and stop. Figure 5 shows how the distance to the starting position of the hand changes during the execution of each gesture command.

4.2. Gesture detection

The gesture detector is built on hand tracking to handle various input data. Given that the walker is primarily for elderly people in need of walkers, the user would usually be sitting and could be partially occluded by a table, for instance, when they execute commands. The detector, thus, must impose minimal restrictions on the user's pose. In addition, the detector must be able to cope with variability in the gesture execution that stems from different levels of cognitive and motor skills of the elderly and various distances between the user and the walker. As hand tracking requires only a hand to be visible, it imposes minimal restrictions.

Hand waving signals the system to start OpenNI's hand tracking [8] and thus the gesture recognition. The hand tracker returns 3-D coordinates $\mathbf{p}_h = \{p_x^h, p_y^h, p_z^h\}$ of the hand, and it marks the 3-D coordinates where the hand tracking began as $\mathbf{p}_s = \{p_x^s, p_y^s, p_z^s\}$. From $\mathbf{p}_h$ and $\mathbf{p}_s$, the feature vector $\mathbf{f}_h = \{f_x^h, f_y^h, f_z^h\}$ can be computed as the absolute distance between the two in each dimension, i.e.,

$$f_d^h = \|p_d^h - p_d^s\|, \tag{1}$$

where d indicates the three dimensions, x, y, and z. One feature vector per frame is extracted and a gesture g is represented as a time series F_g of $N \in \mathbb{N}$ feature vectors,

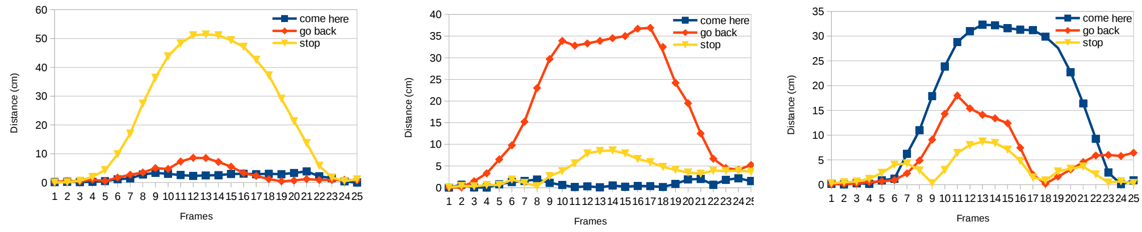
$$F_g = \{\mathbf{f}_1, \mathbf{f}_2, ..., \mathbf{f}_N\}. \tag{2}$$

Once the time series F_g is extracted from the input gesture g, the gesture recognition system classifies it using a k-NN classifier. The classifier requires a training set, which is built by collecting gestures from seven people. Every person performed the three gesture commands 5 times per command, resulting in 105 gestures. Using this training set, the classifier computes the distance from a test time series F_g to the training data and selects k training data items $F_{(t)}, t = 1, ..., k$ that are closest to F_g. It then assigns a class y_i to F_g that is most represented by the k neighbors. As F_g may be spurious, a threshold d_{thresh} introduced such that no neighboring point in $F_{(t)}$ lies farther than d_{thresh} away. For this work, the authors set $k = 5$ and assign a class y_i to the gesture g if 3 or more points in $F_{(t)}$ have the same class label y_i. Performing K-fold cross validation, with K $= 7$ for the seven people, showed that the accuracy remains at 99.1% for $k = 1$ to $k = 12$ and drops to 98.5% for $k = 20$ and to 96.4% for $k = 30$.

To compute the distance between the test time series F_g and the training time series $F_{(t)}$, DTW [25] is used. DTW is a well-known technique in signal processing and has also been successfully applied to gesture recognition [3,32,5]. Given F_g of length N and F_t of length M, DTW finds an optimal match p between F_g and F_t that aligns the two time series with the minimum cost under the following conditions:

1. The match starts at $(1, 1)$ and ends at (N, M).
2. The match can move one step in F_g, F_t, or both but cannot go back in either.

Using these constraints and Euclidean distance as the distance metric between two feature vectors, one can find the minimum cost path using dynamic programming [4]. Dynamic programming constructs the N by M cost matrix that contains the lowest possible matching cost between the two time series F_g and F_t up to the time index i in F_g for the i^{th} row and up to the time



(a) Change in x-direction (depth). (b) Change in y-direction (width). (c) Change in z-direction (height).

Fig. 5. Change in distance with respect to the starting position for the three gestures – “come here” (blue), “go back” (red), “stop” (yellow).

index j in F_t for the j^{th} column. At position (N, M), F_g and F_t are completely matched with the minimal matching cost.

The length N of the test time series is based on the length M of the training set. On average, the gestures of the training set were performed in 17 ± 3.5 frames. The gesture recognition uses the capture window of 50 frames to provide the user sufficient amount of time to execute a gesture. It matches the first 50 frames after the hand tracking starts to the time series stored in the training set.

The RGBD camera has operation range to 3.5 meters. The training set was gathered at 3 meters from the camera and the gesture recognition ran also at the same distance. The maximum tested distance for the gesture recognition is 6 meters. Due to constraints of the underlying technology, the particular RGBD camera device can only be used indoors.

4.3. Face detection

The gestures defined as commands can also be performed and detected when the user does not intend to command the SMARTWALKER. To ignore unintentional gesture commands, the interface uses a face detector. Under the assumption that the user making a gesture to command the walker would look at it, the face detection finds frontal faces in the input stream. The system then processes the detected gesture command only if the frontal face detection is successful.

The face detector uses OpenCV’s Viola and Jones face detector [34]. The detector computes Haar features from the input image and trains the features on positive and negative images to select most distinctive features, i.e., features that have the highest variance between the positive and the negative ones. For fast detection, the detector uses a cascade of classifiers, where each stage uses one or few features to determine if the input image could contain a face. Only those images

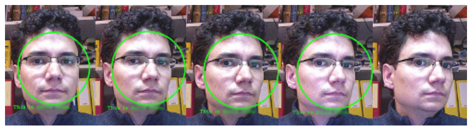


Fig. 6. Face detection and recognition. Only the frontal face and slightly turned faces are detected. The text below the green circle indicates recognition of the detected face.

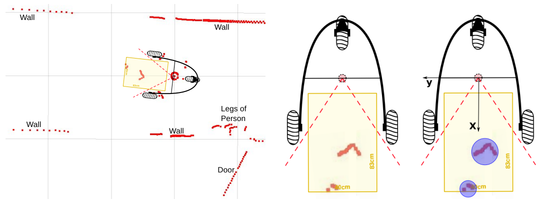
that pass all the stages are positively classified. For further speed improvement, the detector uses depth cues given by the RGBD camera to restrict the search area, as proposed by Burgin *et al.* [6]. In addition, the detector uses depth information to reduce false detection by creating a bounding box around the detected face and validating its size. Any face whose horizontal or vertical dimension is smaller than 8cm or greater than 23cm is deemed unlikely to be a face.

To determine if the user is looking at the walker, the aforementioned classifier is trained only on frontal faces. The detector puts a bounding box around every plausible frontal face. In turn, a higher number of bounding boxes indicates a higher probability of a frontal face. Similar to the approach of Monajjemi *et al.* [24], which selects the most attentive face from many faces using a face score, the detector uses the number of bounding boxes as the face score; however, as there is only one face, it introduces a threshold and accepts only those faces whose face scores are higher than the threshold as frontal faces. Figure 6 shows how turning of a face affects the detection.

4.4. Face recognition

The face recognition ensures that the SMARTWALKER only responds to the commands of its owner. Once a face is detected, the face recognition is activated to determine if the detected face is the owner. The face recognition system is based on the LBPH Face Recognizer [1], provided by the OpenCV library.

LBPH takes a detected face – a cropped image of a face from the face detector – as input and constructs histograms of local binary pattern operators. The operators capture fine grained details such as spots, lines, edges and corners in an image and are invariant to different lighting conditions. LBPH recognizes the input face by comparing its histograms to those of the face training set. It ensures scale invariance by applying local binary patterns operators of different sizes.



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

Fig. 7. Laser scan data (red) for leg detection with the walking area (yellow) and legs (blue).

5. Assistive mode

The assistive mode aims to support the user without requiring them push the weight of the walker. To this end, the walker moves itself at the speed of the user's walking speed, thus minimizing the need for lateral force to move the walker. The developed algorithm combines the user's speed in the walker's reference frame with the wheel speed, the brake, and the ground inclination and computes the walker's speed. Just a small push is all that is needed to start the motorized support.

5.1. 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, the algorithm calculates 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 range scanner scans the area around the walker and provides 360° scans, ranging between $0.02m$ and $4m$ (Figure 7(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 [17], a walking area behind the walker can be defined and points that lay outside of this area can be filtered out(Figure 7(b)); the area is set to $40cm$ by $83cm$ based on performed analysis of the walking patterns of twenty adults. From the filtered data, the algorithm then detects two legs using an expectation-maximization (EM) algorithm [9] (Figure 7(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 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, a threshold is introduced 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 $\mathbf{p}_r$, the user's body center is computed $\mathbf{c} = \frac{\mathbf{p}_l + \mathbf{p}_r}{2}$ 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, the average distance $\widetilde{d}$ over the past 90 seconds is computed to determine the user's position with respect to the recent history.

5.2. Control

The controller takes various sensory information as input and controls the speed of the wheels. It 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 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

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

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 ($v > 0 \land \alpha_{pitch} > 0$) or backward downhill movement ($v < 0 \land \alpha_{pitch} < 0$) and to

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

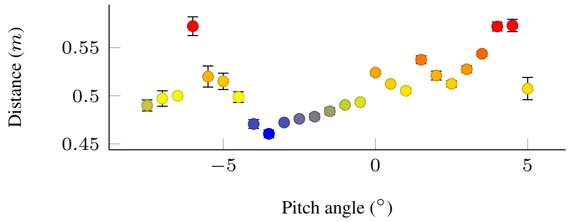


Fig. 8. Ground inclination vs. walking distance.

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

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

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, there is a pitch threshold $\gamma = 3$ and the algorithm considers 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 8). 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.

6. Study setup

The goals of the study are to understand the acceptance and usefulness of a robotic walker for the elderly

Table 1

Respondents' background: age groups, usage of mobility aid devices and modern computing devices.

Categories	Responses				
Age group	Under 70 (1)	$70-79$ (3)	**$80-89$ (11)**	90 and over (6)	
Mobility aid usage	None (3)	Cane (9)	**Rollator (11)**	Wheelchair (3)	
Years of usage	Zero (3)	<1 year (5)	$1-2$ years (3)	**$3-5$ years (8)**	>5 years (4)
Frequency of usage	Never (3)	Rarely (2)	Sometimes (3)	Often (1)	**Always (14)**
Outside usage	Never (4)	Rarely (0)	Sometimes (1)	Often (4)	**Always (14)**
Usage of computing devices	**Never (16)**	Rarely (1)	Sometimes (3)	Often (2)	Always (1)

and to determine the appropriateness of the gesture-based interface and the speed controller. The study was conducted with residents of retirement homes in two parts. The first experiment evaluated the usefulness and appropriateness of the SMARTWALKER and its gesture-based interface of the autonomous mode. The second experiment was conducted six months later and evaluated the automatic speed controller of the assistive mode.

Testing of the autonomous mode required a room with about $5m$ by $5m$ of free space and was conducted either in an activity room or a dining room of the homes. Given the sensor limitation, the gesture-based interface was tested exclusively indoors. Evaluation of the assistive mode was also conducted in an activity room or a dining room, but in addition, the participants were free to go out of the room and test the device anywhere they wanted. Despite this, most participants stayed indoors and only one resident ventured outside with the walker during the test.

Each experiment consisted of each participant filling out a questionnaire and testing the SMARTWALKER and its interface. The experiment took about 30 minutes per person. For every test, the authors followed the participant, suggested them to try various movements and answered their questions. To ease the data-gathering process, an interviewer read out every question and possible answers to each participant and marked the responses on the participant's behalf; reading out the questions was especially important for the elderly with limited or no vision.

Both studies had three parts. After a brief introduction to the SMARTWALKER and the study, the authors first collected each participant's background information such as the age and usage of mobility aid. The participants then walked around the testing room with the walker and answered the questions about its functionality as a regular mobility aid. The third part evaluated the new functionality – the gesture-based interface for the autonomous mode and the speed controller for the assistive mode.

There were also discussions with the staff of the retirement homes. The staff were present for part or all of the experiments, and after the experiments, they shared their opinion with us. Their responses varied from general comments about the visit to specific suggestions for improvement of the walker. There was no particular format to the discussion; the authors simply noted their remarks.

6.1. Autonomous mode

The authors contacted 28 retirement homes via e-mail and received five positive responses. The homes included one male-only place and one for visually-impaired and blind people. Three of the five places made an announcement of the visit to their residents on bulletin boards and allowed people to join the experiment freely. The other two places had contacted individual residents in advance and brought the interested people one by one. In total, 23 residents (14 men and 9 women) and eight members of the staff participated in the study. Four of the residents were visually-impaired or blind. Each evaluation took on average 30 minutes.

The study evaluated the appropriateness and usefulness of the SMARTWALKER using a questionnaire. The research questions investigated whether or not a robotic walker an attractive alternative to a traditional walker, if gesture-based interface an acceptable form of user interface for the elderly, and if the ability to move autonomously a useful functionality. The questionnaire had 23 questions, divided into three groups. The first group was about gender, age, usage of mobility aids, activity level, and familiarity with technology. The second part contained questions about their impression of the SMARTWALKERas a mobility aid such as how they like the walker and how big, heavy, and maneuverable the walker is. The last section had questions related to the usefulness of the

SMARTWALKER's autonomy, the easiness and appropriateness of its gesture-based interface, and their general impression and opinion of autonomous robotic walkers. The questionnaire ended with a free response question about desired functionality of the SMARTWALKER. Appendix A lists a subset of the interview questions.

Table 1 shows the background information of the study participants. The majority of the participants were 80 and above. Most used a rollator or a cane, with three using both. Many stated that they use their mobility aids regularly, both inside and outside, though outside is often limited to the garden area of the retirement homes. Most were unfamiliar with technology, never using a computer or a smart phone.

6.2. Assistive mode

The evaluation of the autonomous mode showed that support in the assistive mode is much-needed functionality. To this end, a speed controller for uphill assistance was added. Six months after the initial visit, the authors contacted the five retirement homes plus three additional places for the evaluation of the assistive mode. Positive feedback was received from three retirement homes, including one that had participated in the initial study. The evaluation was divided into three parts: background information, evaluation of the walker without the controller, and that with the automatic speed controller. The information was gathered using a questionnaire. Appendix B shows a subset of the interview questions. Given the participants' limited motor skills and vision, every question and possible answers were read out to the participants. Thirteen elderly residents participated in the study, and each evaluation took about 20 minutes per participant.

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 day (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.

Table 2
Evaluation of the SMARTWALKER as a mobility aid device.

	Great	Good	Okay	Bad	Terrible	Total
Overall liking	4	7	0	4	2	17
Comfort	10	4	2	3	2	21
Size	0	0	7	7	7	21
Weight	0	0	6	9	6	21
Maneuverability	9	4	3	4	1	21

Table 3
Evaluation of the SMARTWALKER as an autonomous robot.

	Great	Good	Okay	Bad	Terrible	Total
Overall liking	6	6	0	2	4	18
Comfort	16	3	0	1	0	20
Stopping distance	12	3	0	4	1	20
Usefulness	12	5	2	2	2	23
Operability	8	8	2	2	3	23

7. Results

7.1. Autonomous mode

Twenty-one participants evaluated the SMARTWALKER's potential as a mobility aid by walking around with it; two of the wheelchair users could not participate in this portion of the test. Table 2 shows the responses of the participants after testing the device. Eleven of the 17 who were asked how much they like the walker[6] stated that they like the SMARTWALKER. Fourteen out of the 21 found the walker comfortable to use and 13 found it easy to control. Many, however, found it big (14) and heavy (15).

All 23 participants tested the SMARTWALKER's gesture-based interface by calling it towards them and sending it back to a predefined location. Given the camera's operation range, the walker was set at about 3 meters from the user for optimal operation. As an indoor sensor, the camera the had trouble detecting hands in rooms that let too much sunlight in, e.g., with many large windows. The results are shown in Table 3. The interface failed to recognize gestures of three of the participants because they performed the gestures too slowly, and some questions for those participants were omitted.

After testing the interface, 12 out of 18 stated that they like the walker. Nineteen out of 20 said that they were comfortable with the SMARTWALKER coming towards them, and 15 said that its stopping location,

[6]The four visually-impaired people were not asked this question.

Table 4

Evaluation of the gesture-based interface.

	Great	Good	Okay	Bad	Terrible	Total
Gesture execution	19	2	0	0	2	23
Gesture understood	9	7	0	2	5	23
Feedback	10	1	1	7	4	23

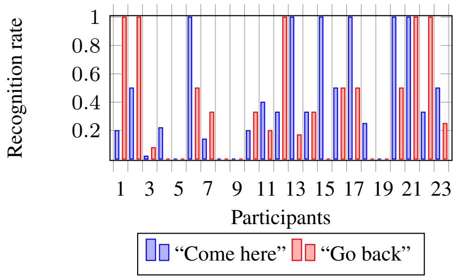


Fig. 9. Recognition rate of the two gesture commands. The participants gave the same gesture command until it was recognized or until they no longer wished to try.

set at $40cm$ from the user, is a good distance. Seventeen out of the 23 found the SMARTWALKER's ability to move by itself useful, and sixteen people said that the walker's gesture-based interface is easy to operate.

In terms of the interface, 21 out of the 23 said that the gestures are easy to execute, and 16 said that their commands were well-understood (Table 4). The recognition rate was 0.41, i.e., the participants had to execute the same command on average 2.4 times before the interface recognized the command (Figure 9). The low recognition rate is partially due to the mismatch between the training set and the test set; the training set was built using healthy adults in their 20's and 30's while the test was performed by elderly people. Lastly, eleven participants said that the voice feedback that the robot gives when it recognizes a gesture is sufficient whereas eleven said that the voice feedback is insufficient due to the low volume.

Nineteen of the 23 participants found the idea of a robotic walker very exciting (11) or exciting (8); however, eleven said that they would rather not (4) or definitely not (7) replace their traditional mobility aids with robotic walkers. Only a small portion said that they would definitely (6) or likely (1) replace their traditional aids. The low acceptance may be due to the walker's size and weight and the elderly's unfamiliarity with technology.

The elderly suggested several additional features for the walker. They include a seat, the ability to identify its owner, which was implemented but not tested with them as it required collecting their face data first, a better feedback system, a reduction in size and weight, assistance for uphill and stair climbing, and safety assurance.

Comments from retirement home employees

All eight personnel (5 women, 3 men) from the five retirement homes had a positive impression of the SMARTWALKER and the authors' visit. They shared their comments about the device, recommending new features for the walker. Their suggestions were uphill/downhill support, a parking brake for safety, and an ergonomically designed tiltable hand grip. The retirement home for visually-impaired and blind additionally wanted obstacle and stair recognition for warning, more audible warnings when the walker is in action, a physical interface (e.g. buttons) rather than a touch screen or a gesture-based one, and an emergency alarm.

Everyone said that an autonomously-moving robotic walker would be particularly useful at mealtimes. Moving walkers in and out of a dining room is a laborious process, requiring a lot of resources. Currently, the staff parks the walkers outside of the dining room at the beginning of each meal and brings them back to the residents one by one when the mealtime is over. A robotic walker would eliminate this laborious process. Moreover, it would enable the residents to have a meal and leave the room when they wish, and this prospect was particularly well-received by the residents.

7.2. *Assistive mode*

Evaluation of the assistive mode was conducted six months after the evaluation of the autonomous mode. After sharing their background information, the participant walked around the premise of the retirement homes with the walker. For this portion, the speed controller/motor support 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, there was not defined 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 SMARTWALKER once again but with the controller

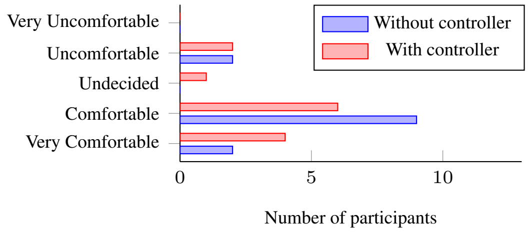


Fig. 10. Walking quality.

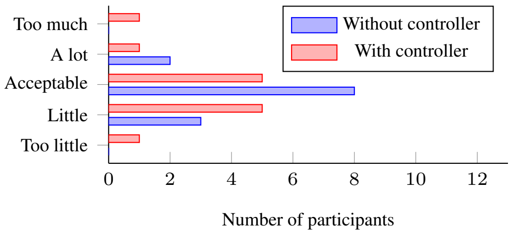


Fig. 11. Required effort for pushing the walker.

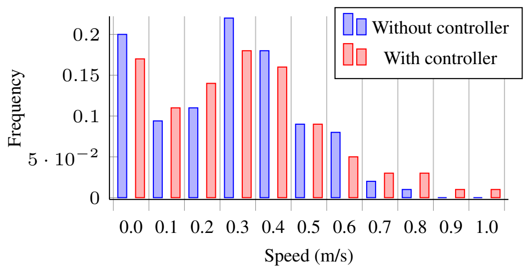


Fig. 12. Walking speed.

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 controller, which is similar to the level of comfort they felt when walking without it (Figure 10). 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 11). 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.

There was a change in walking speed between without and with the controller (Figure 12). 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.

Table 5

Evaluation of the SMARTWALKER's size and weight and willingness to replace traditional walker to the SMARTWALKER.

Would replace	Too big	Big	Perfect	Too heavy	Heavy	Perfect
Definitely (6)	0	**4**	2	0	3	3
Probably (1)	0	1	0	0	1	0
Not sure (2)	0	1	1	0	1	1
Probably not (4)	2	0	2	1	**3**	0
Definitely not (7)	**4**	1	2	**5**	0	2

8. Discussion

8.1. *Autonomous mode*

The overall impression of the SMARTWALKER by the residents and the staff was positive. Most residents found the walker exciting, and the staff showed their interest in the device. As a prototype, the SMARTWALKER, however, was not yet deemed an acceptable replacement of a traditional walker. This may be because the walker is too big and heavy and the elderly are unfamiliar with technology.

8.1.1. *Acceptance of robotic walker*

More than half of the participants said that they would rather stay with a traditional walker. This was especially true for current walker users, with nine out of the 11 stating that they would definitely not (6) or probably not (3) change to a robotic walker. This is surprising given that five of the 11 liked the walker and eight found it exciting. Unfortunately, their positive review did not translate into their willingness to switch to the SMARTWALKER.

A possible cause is the SMARTWALKER's size and weight. Equipped with extra hardware, the walker is bigger and heavier than normal walkers. While none of the seven people who would consider changing found the SMARTWALKER too heavy and three even found the weight just right, only two of the 11 who would not change found the weight acceptable; six of the 11 found the walker too heavy (Table 5). The stuff explained that many residents have trouble going over a curb using a traditional walker and thus stay within the perimeter of the retirement homes. The residents may have felt that the SMARTWALKER would hinder their autonomy even further.

Another cause may be the unfamiliarity with technology. As most participants did not use any computing devices, they could have found the SMARTWALKER overwhelming to use. Indeed, those who use technology regularly were more willing to switch to a robotic walker than those who do not (Table 6).

8.1.2. Appropriateness of gesture-based user interface

The evaluation of the gestured-based interface was positive despite the low recognition rate. In contrary to their positive feedback that the gestures were easy to perform, the authors noticed that the participants had a hard time executing gesture commands to the walker. Many participants had limited fine-motor skills due to old age, and some participants even suffered from movement disorders such as Parkinson's disease. The visually-impaired and blind people had extra difficulty with the gesture commands as they had to conceptually translate a verbal explanation of the gestures to physical movements.

As an alternative to gesture, several staff members suggested using a button and integrating it into the medical alert system. As most residents have alarm buttons, integrating the interface into the system would make it easier for the residents to use the interface. Interestingly, they also pointed out that many residents do not wear their alarm button despite owning one. They generally prefer simple and physical interface, but determining the most appropriate interface requires a further study.

8.1.3. Usefulness of autonomous walker

Most residents found the walker's autonomy useful, but many were initially unsure of its actual use case. The staff at the retirement homes saw its usefulness more readily, stating mealtimes as the main use case. An autonomous walker would eliminate the laborious process of parking and fetching walkers. The elderly, once told of the scenario, were also excited about it. In a longer term, several imagined the walker developing into a butler.

Table 6

Factors influencing the evaluation.

	Average recognition rate	Male (14)	Female (9)	Technology user (7)	Non-tech. user (16)
Replace (7)	0.31 ± 0.27	**7**	0	**4**	3
Not replace (11)	0.40 ± 0.36	4	**7**	2	**9**

8.1.4. Factors influencing the evaluation

Flandorfer's meta study [12] showed that different sociodemographic factors influence the elderly's acceptance of socially-assistive robots. The authors analyzed the influence of the walker's performance and the elderly's gender and experience with technology on their willingness to switch to the SMARTWALKER (Table 6). In terms of the gesture interface's recognition rate, the authors noticed no significant difference between those who are willing to replace and those who are reluctant. In terms of gender, about half of men were interested in switching while no women were interested. Similarly, more than half of the technology users were willing to replace to the SMARTWALKER whereas many non-technology users were unwilling to change. Although the small sample size makes it difficult to draw any statistically significant conclusions, the results show that gender and technology may have some influence on the participants' acceptance of the SMARTWALKER.

8.2. Assistive mode

Overall 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 SMARTWALKER with its controller while four preferred their current mobility aid and two were undecided. Most participants were very interested in the 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.

8.3. Limitations

This study was conducted with residents of retirement homes and would thus reflect the preference of those who live in community-living environments. No participant regularly carried out outside activities such as grocery shopping; in fact, although most participants went outside regularly, they mostly stayed in the garden area and rarely left the perimeter of the retirement homes. Given that many walker users do live independently, the findings of this study may not be applicable to the general population of current and potential walker users. Moreover, walkers provide temporary support for those in rehabilitation, but this study did not investigate the appropriateness of the device for the group.

9. Conclusion

This paper introduced the SMARTWALKER, an autonomous robotic walker for the elderly, with a gesture-based user interface and an automatic speed controller. Equipped with low-cost, off-the-shelf sensors and actuators, the SMARTWALKER can intelligently interact with its environment and support its user. The device was evaluated in two stages; first, twenty-three residents and eight staff members at five different retirement homes evaluated the gesture-based interface. Later, thirteen residents of three different retirement homes evaluated the speed controller. The initial study showed that although the residents liked the SMARTWALKER and found its interface easy to use, they preferred to stay with traditional walkers, possibly because the SMARTWALKER is bulky and heavy and many elderly are unfamiliar with technology. The follow-up study with the speed controller was better received in that the participants found the walker with the speed controller slightly more comfortable, 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. The authors are thus interested in understanding the demand for robotic walker among a wider group of people, in particular, walker users who live independently and those in rehabilitation.

Appendix

A. Autonomous mode: selected questions from the questionnaire

Part one (1–7):
4) Do you use?
☐ nothing ☐ cane ☐ walker/rollator ☐ wheelchair
5) How often do you use the device?
☐ never ☐ seldom ☐ sometimes ☐ often ☐ always
7) How often do you use a smartphone, a computer, ...?
☐ never ☐ seldom ☐ sometimes ☐ often ☐ always

Part two (8–12):
8) Do you like the rollator?
☐ a lot ☐ somewhat ☐ neutral ☐ not really ☐ not at all
11) How comfortable was it to walk with the rollator?
☐ very comfortable ☐ comfortable ☐ neutral ☐ uncomfortable ☐ very uncomfortable

Part three (13–23):
14) Is it useful that the rollator can come and go away?
☐ very ☐ somewhat ☐ neutral ☐ not really ☐ not at all
18) Were the hand gestures difficult for you to execute?
☐ very difficult ☐ difficult ☐ neutral ☐ easy ☐ very easy
19) Did the robot understand your hand gestures well?
☐ always ☐ mostly ☐ neutral ☐ sometimes not ☐ never
22) Would you change to a robotic walker, or would you rather stay with a traditional one?
☐ definitely change ☐ probably change ☐ neutral ☐ probably not change ☐ definitely not change

B. Assistive mode: selected questions from the questionnaire

Part one – questions about the user (1–7):
3) Do you use a walking aid?
☐ no ☐ cane ☐ walker/rollator ☐ wheelchair
4) When yes, how long have you been using the device?
☐ <1 year ☐ 1–2 years ☐ 3–5 years ☐ >5 years
5) How often do you use the device?
☐ daily ☐ 4–6x/week ☐ weekly ☐ less
7) How often do you use a computer or similar device (smartphone)?
☐ daily ☐ 4–6x/week ☐ weekly ☐ less

Part two – use of the rollator without the motor support (8–11):
8) How comfortable was it to walk with the rollator? ☐ very uncomfortable ☐ uncomfortable ☐ comfortable ☐ very comfortable ☐ I don't know
9) How do you find the size of the rollator? ☐ too big ☐ big ☐ comfortable ☐ small ☐ too small
11) How do you feel about the effort required to push the rollator? ☐ too big ☐ big ☐ comfortable ☐ small ☐ too small

Part three – use of the rollator with the motor support (12–17):
12) How comfortable was it to walk with the rollator with the motor support?
☐ very uncomfortable ☐ uncomfortable ☐ comfortable ☐ very comfortable ☐ I don't know
13) How do you find the speed of the rollator?
☐ too fast ☐ fast ☐ comfortable ☐ slow ☐ too slow
15) Do you find it more comfortable to walk with or without the motor support?
☐ without the support ☐ with the support ☐ I don't know
17) Do you prefer our rollator to your current rollator?
☐ yes ☐ rather yes ☐ rather no ☐ no ☐ I don't know

Acknowledgment

We thank Marcel Mathis, David Itten, and Ivo Steinmann for their involvement in the project and the residents and the staff for participating in the study. This work was partially supported by the European Research Council under the European Union's Seventh Framework Programme (ERC Grant agreement no. 291389) and the Hasler Foundation under the SmartWorld programme.

References

[1] T. Ahonen, A. Hadid, and M. Pietikäinen. Face recognition with local binary patterns. In T. Pajdla and J. Matas, editors, *Computer Vision - ECCV 2004*, volume 3021 of *Lecture Notes in Computer Science*, pages 469–481. Springer, Berlin & Heidelberg, 2004.

[2] P. Barattini, C. Morand, and N. M. Robertson. A proposed gesture set for the control of industrial collaborative robots. In *RO-MAN, 2012 IEEE*, pages 132–137. IEEE, 2012.

[3] M. A. Bautista, A. Hernández-Vela, V. Ponce, X. Perez-Sala, X. Baró, O. Pujol, C. Angulo, and S. Escalera. Probability-based dynamic time warping for gesture recognition on rgb-d data. In *Advances in Depth Image Analysis and Applications*, pages 126–135. Springer, 2013.

[4] R. Bellman. On the theory of dynamic programming. *Proceedings of the National Academy of Sciences of the United States of America*, 38(8):716, 1952.

[5] V. Bloom, D. Makris, and V. Argyriou. Clustered spatio-temporal manifolds for online action recognition. In *22nd Int. Conf. on Pattern Recognition (ICPR)*. IEEE, 2014.

[6] W. Burgin, C. Pantofaru, and W. D. Smart. Using depth information to improve face detection. *6th Int. Conf. on Human-Robot Interaction*, 2011.

[7] C. A. Cifuentes, A. Frizera, R. Carelli, and T. Bastos. Human–robot interaction based on wearable imu sensor and laser range finder. *Robotics and Autonomous Systems*, 62(10):1425–1439, 2014.

[8] A. Davison. *Kinect Open Source Programming Secrets: Hacking the Kinect with OpenNI, NITE*. McGraw Hill Professional, 2012.

[9] 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.

[10] D. Droeschel, J. Stückler, D. Holz, and S. Behnke. Towards joint attention for a domestic service robot-person awareness and gesture recognition using time-of-flight cameras. In *2011 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1205–1210. IEEE, 2011.

[11] A. Elias, A. Frizera, T. F. Bastos, and C. Valadão. Robotic walkers from a clinical point of view: Feature-based classification and proposal of the ufes walker. In *2012 ISSNIP Biosignals and Biorobotics Conf. (BRC)*, pages 1–5, 2012.

[12] P. Flandorfer. Population ageing and socially assistive robots for elderly persons: The importance of sociodemographic factors for user acceptance. *Int. J. of Population Research*, 2012: 13 pages, 2012.

[13] W. Gharieb. Intelligent robotic walker design. In *Int. Conf. on Automation, Robotics and Autonomous Systems*, 2006.

[14] B. Gleeson, K. MacLean, A. Haddadi, E. Croft, and J. Alcazar. Gestures for industry: intuitive human-robot communication from human observation. In *Proceedings of the 8th ACM/IEEE international conference on Human-robot interaction*, pages 349–356. IEEE Press, 2013.

[15] J. Glover, D. Holstius, M. Manojlovich, K. Montgomery, A. Powers, J. Wu, S. Kiesler, J. Matthews, and S. Thrun. A robotically augmented walker for older adults. Technical report, CS Department, Carnegie Mellon University, USA, 2004.

[16] H. Hashimoto, A. Sasaki, Y. Ohyama, and C. Ishii. Walker with hand haptic interface for spacial recognition. In *9th IEEE Int. Workshop on Advanced Motion Control*, 2006.

[17] 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.

[18] G. Lacey and K. Dawson-Howe. The application of robotics to a mobility aid for the elderly blind. *Robotics and Autonomous Systems*, 23(4):245–252, July 1998.

[19] G. Lee, E.-J. Jung, T. Ohnuma, N. Y. Chong, and B.-J. Yi. Jaist robotic walker control based on a two-layered kalman filter. In *2011 IEEE Int. Conf. on Robotics and Automation*, pages 3682–3687, May 2011.

[20] 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 Cybernetics: Systems*, 44

(5):665–672, May 2014.

[21] M. Martins, C. Santos, E. Seabra, A. Frizera, and R. Ceres. Design, implementation and testing of a new user interface for a smart walker. In *IEEE Int. Conf. on Autonomous Robot Systems and Competitions*, 2014.

[22] 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.

[23] B. Meyer. *Touch of Class: learning to program well with objects and contracts*. Springer Science & Business Media, 2009.

[24] V. M. Monajjemi, J. Wawerla, R. Vaughan, and G. Mori. Hri in the sky: Creating and commanding teams of uavs with a vision-mediated gestural interface. In *2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 617–623, 2013.

[25] M. Müller. *Information Retrieval for Music and Motion*, chapter Dynamic Time Warping. Springer, 2007.

[26] O. Postolache, P. Girao, J. D. Pereira, J. Pincho, C. Moura, and G. Postolache. Smart walker for pervasive healthcare. In *2011 Fifth Int. Conf. on Sensing Technology (ICST)*, pages 482–487, 2011.

[27] M. Quigley, B. Gerkey, K. Conley, J. Faust, T. Foote, J. Leibs, E. Berger, R. Wheeler, and A. Ng. ROS: an open-source robot operating system. In *2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009.

[28] A. Rusakov, J. Shin, and B. Meyer. Simple concurrency for robotics with the Roboscoop framework. In *2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2014.

[29] J. Shin, D. Itten, A. Rusakov, and B. Meyer. Smartwalker: Towards an intelligent robotic walker for the elderly. In *2015 International Conference on Intelligent Environments (IE)*, pages 9–16. IEEE, 2015.

[30] J. Shin, I. Steinmann, and B. Meyer. Automatic speed control for SmartWalker. In *Proceedings of the 8th ACM International Conference on Pervasive Technologies Related to Assistive Environments (PETRA)*, page 9. ACM, 2015.

[31] J. Suarez and R. R. Murphy. Hand gesture recognition with depth images: A review. In *2012 IEEE Int. Symposium on Robot and Human Interactive Communication*, 2012.

[32] G. Ten Holt, M. Reinders, and E. Hendriks. Multi-dimensional dynamic time warping for gesture recognition. In *13th annual Conf. of the Advanced School for Computing and Imaging*, 2007.

[33] United Nations, Department of Economic and Social Affairs, Population Devision. Population ageing and sustainable development, August 2014. URL `www.unpopulation.org`. Accessed.

[34] P. Viola and M. Jones. Rapid object detection using a boosted cascade of simple features. In *2001 IEEE Conf. on Computer Vision and Pattern Recognition*, volume 1, 2001.

[35] 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.

The final publication is available at IOS Press through http://dx.doi.org/10.3233/AIS-160383.