ELSEVIER

Available online at www.sciencedirect.com

SciVerse ScienceDirect

Procedia Engineering 41 (2012) 538 – 546

Engineering Procedia

www.elsevier.com/locate/procedia

International Symposium on Robotics and Intelligent Sensors 2012 (IRIS 2012)

Development of Smart Wheelchair System for a User with Severe Motor Impairment

Mohd Razali Md Tomari[a,c]*, Yoshinori Kobayashi[a,b], Yoshinori Kuno[a]

[a]Graduate School of Science & Engineering, Saitama University, 255 Shimo-Okubo, Sakura-Ku, Saitama-Shi, 338-8570, Japan
[b]Japan Science Technology Agency, PRESTO, 4-1-8 Honcho Kawaguchi, Saitama 332-0012, Japan
[c]Faculty of Electrical & Electronic Engineering, Universiti Tun Hussein Onn Malaysia, 86400 Parit Raja Batu Pahat, Malaysia

Abstract

Users with severe motor impairment may find it difficult to operate a wheelchair when they are in tight space (e.g., passing doorway) or when avoiding obstacles since they cannot command the wheelchair by means of a conventional joystick. Here we propose a framework that can assist users to overcome such circumstances using a hierarchical semi-autonomous control strategy. Initially multimodal user inputs based on momentary switch and yaw head angle are analyzed to decide a maneuvering mode and assign the direction of travel. Then, environmental information is perceived using the combination of a laser range finder and the Kinect sensor for determining safety map around wheelchair's vicinity. Eventually, the user's inputs are provided to the navigation planner along with the safety map to moderate motion that is collision free and the best for user preference. Experimental results demonstrate the feasibility of the proposed approach.

Keywords: Smart wheelchair, multimodal interface, semi-autonomous control, hybrid sensor.

Nomenclature

RGBD	Red-Green-Blue-Depth
HCI	Human computer interface
DOF	Degree of freedom
FOV	Field of view

1. Introduction

In recent years, numerous methods have been introduced for developing smart wheelchairs system to accommodate the disabled people, and some of them can be found in [1]. The development trend can be broadly classed into three main areas [2]: 1) Improvements to the assistive technology mechanics, 2) Improvements to the user-machine physical interface, 3) Improvements to shared control between the user and the machine. One of the key aspects of smart wheelchair is to provide independent mobility for users with severe impairments who cannot control the wheelchair by means of a standard joystick. This may be due to several reasons such as cerebral palsy, cognitive impairment, or fatigue [3]. It has been reported that people with mobility difficulties tend to be more depressed or anxious compared to normal people [4]. Therefore recovering their mobility may significantly improve their quality of life.

* Corresponding author. Tel.: +81-48-858-3585; fax: +81-48-858-3585
E-mail address: mdrazali@cv.ics.saitama-u.ac.jp

1877-7058 © 2012 Published by Elsevier Ltd. Open access under CC BY-NC-ND license.
doi:10.1016/j.proeng.2012.07.209

Generally, the devised smart wheelchair platform highly depends on the user's profile [5] (i.e. abilities and disabilities), and there is no single solution that suitable for all users. Patients with motor impairment (e.g. spinal cord injury) generally lack of muscle control, and in the worst case they are unable to command the movement of arms and legs. To aid mobility of these patients, according to [6] cues or actions generated from the head (e.g. voice, brain, gaze, tongue, and bite) can be possible input devices for all level of injuries. The word `possible` here means that if the level of injury is low, patients can use his/her hand or head to generate input commands; however as the level increased, input sourced from the head is perhaps the only solution. Voice-activated navigation [7] requires quiet ambience and may not be good for use in busy and noisy environments. Furthermore, it is not good and sometimes considered to be impolite to talk aloud in a silent area. As for brain wave [8], recently it has become a subject of interest for controlling machines. To do so, electroencephalography (EEG) signal patterns need to be classified and grouped into the intended actions. The user must have good emotional control and concentration for effective control. This is a burden to the user although this medium may be a good alternative for people with a totally paralyzed body. Gaze input offers good information such as head [9] and eyes [10] directions for manipulation. The basic idea is that an area at which the user gazes represents the intended direction, which imitates human physiological behaviour during walking [11] or driving. While this medium seems a good candidate, it is hard to distinguish between actions for steering the wheelchair or simply looking around. Thus the user has to refrain from viewing surroundings and to concentrate on driving during navigation. Perhaps the best solution for inferring input from the user with severe motor disabilities is relying on multi-user input approaches [12][13], in such a way that multiple possible user's cues are analyzed before providing the desired command. Using this strategy we can assign each controlling task into the different user's input and hence will impose less burden to the user compared to the case that solely relies on a single input.

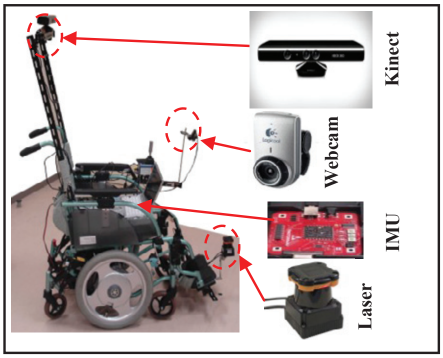
While alternative to joystick mediums enable the target patient to maneuver, the navigation burden is still totally on the user. According to [14], among 200 practicing clinicians in U.S. 40% of their patients find it hard or impossible to control the wheelchair even though with such alternative mediums. Most recently, it has also been reported in [15] that most patients with motor impairment are unable to steer the wheelchair for avoiding obstacles and parallel parking. These clinical findings provide insight for the importance for devising a computer-controlled platform to assist the users by reducing their workload and increasing the safety. In this framework, the user input along with the environmental information will be seamlessly analyzed for performing necessary assistive tasks. The amount of given assistance usually varies depending on how severe the users` impairments. The assistance can be categorized into three main levels; shared-control, semi-autonomous control, and autonomous control [16]. The level of given assistance should be decided by considering the maximum of users' abilities to control the wheelchair first, and the computer will complement the loophole [5]. Shared-control lets the user fully command the wheelchair. The computer only steps in when it is absolutely necessary (e.g. passing through a door or avoiding obstacles) [17][18].The autonomous control lets the wheelchair move automatically on the routs to the final destination pre-selected by the user[19]. Due to such nature, shared-control is more suitable to the users who can provide continuous input command with a joystick. Autonomous control is more adequate to the users who are unable to provide low level orders, who feel fatigue easily, and who have visual impairment [3]. Since people with severe motor disabilities can still command their heads even with limited amount of movement, semi-autonomous control is a proper assistive choice. In this control mode, the computer performs short-term route planning, and the users only intervene it when they wish to deviate from the plan [20][21]. This means that, once a command is issued, the users can relax while the computer is completing the task. Unlike the autonomous control, the semi-autonomous control does not need an actual map of the environment; only a local safety map based on sensors scanning is needed. Hence it can give freedom to the users to move in new environments.

In this paper we propose a semi-autonomous control wheelchair system with a multi-input interface to aid mobility of people with severe motor impairment. Since the user can issue a limited control command within a short period of time, the computer will take over the responsibilities for navigating and avoiding any possible threat, while the user is only responsible for heading the wheelchair into the desired directions. With such setup, the critical and dangerous situation can be effectively overcome, while at the same time the user can still feel driving the wheelchair.

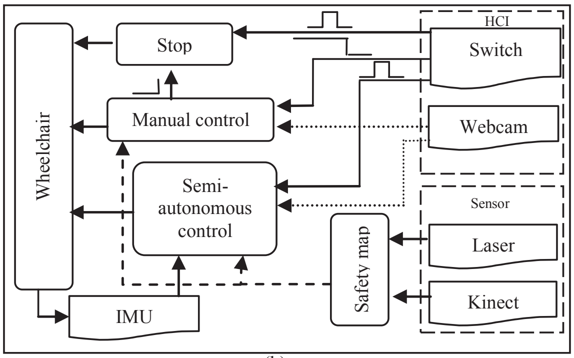
2. System Overview

Our system is implemented on an electric wheelchair (TT-Joy, Matsunaga Corp.). It is equipped with a switch and four types of sensors as shown in Fig. 1 (a): a standard webcam (Logicool), a RGBD Camera (Kinect, Microsoft), a laser range finder (UTM-04LX, Hokuyo Electric Machinery), and an Inertial Measurement Unit sensor (VN-100, Vectornav).

As illustrated in Fig. 1(b) the first HCI input device is the switch. It is a single and momentary type in nature that is responsible for triggering several maneuvering modes (i.e., "stop", "semi-auto" and "manual") depending on how long the user lets it "on." It can be realized by various mediums such as detecting motion of facial parts (e.g. eye blinking or shaking), voice or a button switch. Physiological features will impose much burden to the user when s/he needs to issue a command frequently, especially when s/he navigates in a limited space. Therefore we choose to use a simple bite-like switch button (i.e., not an actual bite switch but the one that we developed imitating the nature of the switch operation). The second HCI input device is the webcam that is utilized to steer the wheelchair in the manual mode. It also provides the general direction to move when facing with obstacles in the semi-autonomous mode.



(a)



(b)

Fig. 1 : (a) Sensor arrangement. (b) General operation flow.

For the sensor part, the combination of the laser and Kinect is used for detecting obstacles and threatening areas in the wheelchair's vicinity. Before using both sensors, a calibration is performed to ensure the reliability of the obtained data. We used the standard stereo calibration procedure for the Kinect, and the linear calibration model for tailoring data from the Kinect and the laser. The IMU sensor provides the information about the wheelchair's current state in the world coordinate and is mainly employed here for correcting the heading orientation. Output from this sensor is calibrated, and the errors are compensated by using Extended Kalman filter. Therefore, the output provides reliable 6 DOF reading.

Basically during full time operation, we do not directly use the HCI inputs to continuously control the wheelchair; instead we only use it for heading the wheelchair into the user's intended direction through the manual mode. In our system, the computer through semi-autonomous controller takes over the responsibilities for navigation and low level control, while the user takes care of only determining the general directions of travel. This results in low user involvement and higher user comfort. During maneuvering, if the user wants to stop or deviate from the planned path, s/he can always interrupt by issuing a "stop" or "manual" command by using the switch. In the manual mode, the gaze direction is used to steer left or right until the desired direction is found. It can be issued several times by repeating the same process until the user is satisfied. Once this operation finished, there are two possibilities; the user already reaches the destination, or s/he wants to travel into the headed direction. Therefore if the user wishes to execute the latter action, s/he needs to issues the "semi-auto" command. The computer sets the current heading orientation as the new direction of travel and resumes navigation while avoiding obstacles.

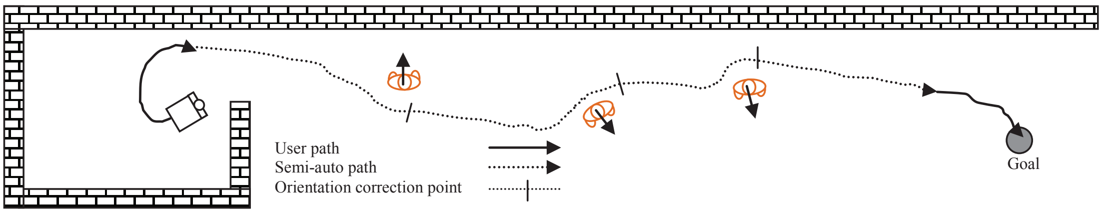


Fig.2. Illustration how the proposed system can assist the user in practical operation.

Fig. 2 illustrates how the proposed system can give the assistance. In this example, the user wants to move from the current position to the goal location. Initially s/he manually alters the heading to face the goal location. When semi-auto control is activated, the wheelchair moves to the desired direction, avoiding obstacles and refines the orientation when necessary. Due to the series of such operations, the final position may be a bit diverse from the goal location. In such cases, the user can manually steer it until the wheelchair completely reaches the goal. If no obstacles appear during travelling, apparently the wheelchair just moves straight to the goal location. We tune our system in this way because usually unlike joystick, any alternative medium offers a few distinct commands and sometimes contains errors during the obtaining process; therefore solely relying on such a medium may lead to an inappropriate motion command. Furthermore, the medium requires more concentration from the user. Consequently, for a long time operation, this may become a burden to him/her, and s/he cannot enjoy the surroundings while maneuvering.

By using this procedure, we can give freedom to the user. S/he can partly maneuver the wheelchair by giving a command to go to the new goal direction as s/he wish in an easy and natural way, while reducing the navigation burden for a long time operation.

3. Human Computer Interface and Manual Control

A conventional joystick is well known for its functionality to control the wheelchair effectively. Therefore, when users are unable to use this medium, the best alternative should be able to imitate the joystick operation. It should be easily adapted to the user and minimize the user's workload. In our system, we use gaze direction for commanding the "left" and "right" since it naturally reflects human behavior when changing direction during walking or driving.

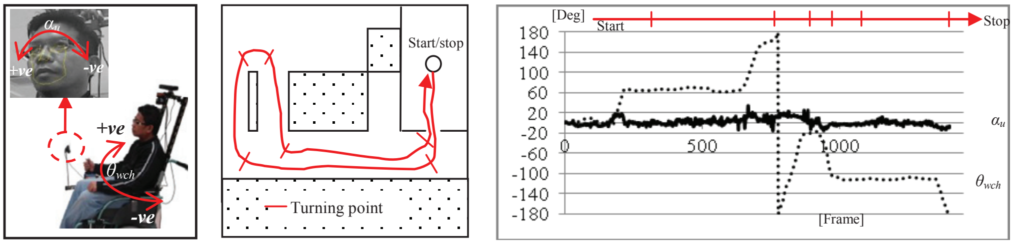


Fig.3. (a) Gaze and wheelchair directions. (b) Manual driving route. (c) Measured directions and wheelchair orientation.

For obtaining the gaze data, we use FaceAPI software [22] that is capable to supply such information in real time. In our work only yaw angle is considered since it is enough to characterize where the current gaze is directed. Fig. 3(a) shows an example of the inferred gaze where the yaw angle is denoted by α_u , and the wheelchair orientation is labeled by θ_{wch}. In order to gain insight on how head's gazing naturally behaves while the user is driving the wheelchair, we recorded values of α_u and θ_{wchr} when the user manually steered the wheelchair using the joystick in the environment illustrated in Fig. 3(b). The measurement results are shown in Fig. 3(c). Based on the results, we empirically classify the yaw angle (α_u) values into three possible actions: { $\alpha_u \geq 15^\circ$="turn right"}, {$\alpha_u \leq -15^\circ$="turn left"}, and {$-15^\circ > \alpha_u > 15^\circ$="move forward"}. For the turn action, the greater amount of α_u with respect to the predefined threshold (i.e.$\pm 15^\circ$), the sharper turning curve will be generated. However, the gaze commands are executed only when the user presses and holds the button switch. With such setup, the user`s actions which intend to steer the chair or just look around can be easily distinguished by the controller.

The switch button triggers different functions depending on how it is pressed. When receives a cycle of momentary pattern (low→high→low), the stop or semi-auto command is executed depending on the current state (i.e., when the current state is stop, the system executes semi-auto, and vice versa). When continuously receives high signal, the system enters full user control mode until the low signal is issued. Every time the user exits from the manual mode, the system stops and waits for the next command.

4. Safety Map

In our system the top priority is the user safety, and hence in all operating modes the safety map is supplied to ensure that the generated motion is free from collision. In the semi-autonomous mode, the map guides the computer to generate an optimal motion while seeking the goal and avoiding obstacles. In the manual mode, the map helps the user to brake when the given command is subject to collision.

To allow reasoning about the safety map development, we make use of the combination of the laser sensor and Kinect camera. The former is used for perceiving solid obstacle placement, and the latter is used for detecting any obstacles that are not uniform in dimension of the shape (e.g. table, chair). Both sensors complement each other in a sense that the laser has a wide FOV but is unable to precisely portray the surrounding. On the other hand, although the Kinect has a small FOV, it can recover the 3D information.

The Kinect is fully calibrated and mounted at 1.3meters high from the ground plane [23]. Since we use it for detecting obstacles, the camera is tilted downwards and hence most likely the ground information will be partly visible. In order to prune out such information we rely on the height map. As illustrated in Fig. 4(a), P is the camera point, D is the ground plane, C is the camera centre location, and d is a series of height thresholds. When the camera is facing forward (C=C_s), we can easily determine the ground plane using the simple threshold of d_1 which is the actual height where the camera is installed. However when the camera is tilted down (C=C_t), this assumption will not work since the farther the ground plane from the camera is, the smaller the height threshold is. To overcome this problem, we modelled the ground plane as Δ: $Z= \alpha X + \beta Y + \gamma$, where (X, Y, Z) is the 3D point's location in the world coordinate and α, β, and γ are constant parameters to represent the 3D plane of the ground. The parameters are estimated by selecting non-collinear points belonging to the floor and performing the least square fitting method. Fig. 4(b) shows a sample of the detected ground plane (highlighted by white pixels in the RGB image) and the resultant range map. From these, we can see that the removed ground plane does not cut too much information of the obstacles, and the range map can give the precise obstacle shape and placement.

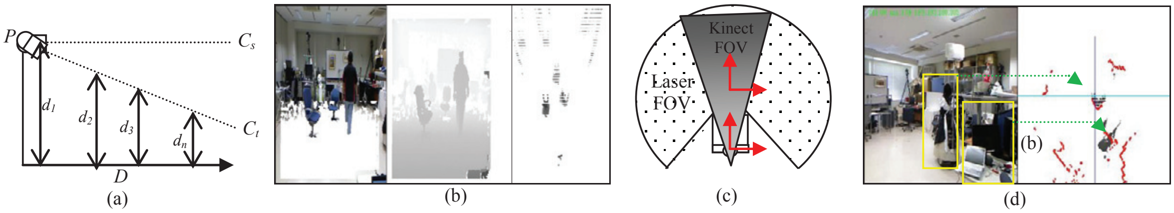


Fig.4. (a) Camera geometry. (b) Kinect images after ground plane removal: *(left)* RGB, (*middle*) Depth, (*right*) Range Map. (c) Arrangement of the Kinect and laser sensors. (d) Sample of perceived data: Kinect (*grey*) and laser (*red*).

Since the Kinect and laser sensors are located in different axes, we calibrated both data by minimizing the reading error in each FOV angles. The calibration process makes use of the Kinect's axis as the reference. Empirically we found that the linear transformation model is enough to attain high precision of the fitted data with a low computational complexity. Fig.4.(c) illustrates the arrangement of both sensors on the wheelchair and how the information is overlaid, and Fig. 4(d) shows a sample of the obtained information consisting of RGB image (*left*) and safety map *(right*). In the safety map, we can see that both data (i.e. Kinect in grey and laser in red) are well aligned and the Kinect successfully detects the table placed on the right side, which the laser is unable to correctly sense. This example shows one of the reasons why sensor integration is important. With more information we can model the detailed safety map of the local surrounding. With a more accurate map, the risk of collision can be reduced, especially when obstacles in irregular shapes exist in cluttered environments.

5. Semi-autonomous Control

To allow reasoning of the navigation planning in the semi-autonomous control, we adopt an extension of Vector Field Histogram (VFH+) [24] with a slight modification. VFH uses the 2D Cartesian histogram known as the active window that is built around the wheelchair with a square shape. Each active cell in the window grid is transformed into the 1D histogram representation known as Polar Obstacle Density (*POD*) in the form of $POD(\omega_{i,j}) = m_{i,j}$ using equation (1), where ($x_{i,j}$, $y_{i,j}$) is the coordinate of the active cell , (x_0,y_0) is the coordinate of the sensor origin (SO), $C_{i,j}$ is the certainty value of the obstacle in the cell, d is the distance to the cell from SO, and a and b are design parameters.

$$\omega_{i,j} = \tan^{-1}\left(\frac{y_{i,j}-y_0}{x_{i,j}-x_0}\right), m_{i,j} = c_{i,j}^2 \times \left(a - bd_{i,j}^2\right) \quad (1)$$

For developing the primary *POD* we use the information provided by the safety map, which is visualized in Fig.4 (d). To convert this *POD* into Binary representation (POD_b), a threshold is needed. We define two values namely τ_c and τ_d in which $\tau_d > \tau_c$. The former is used for discriminating high and low POD, and the latter is used for determining high risk obstacle (*HRO*) locations.

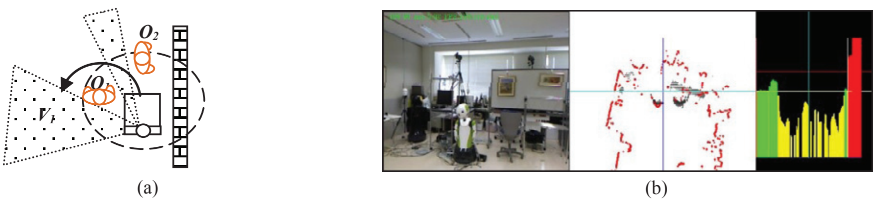


Fig. 5. (a) High risk obstacle encountering situation. (b) System monitoring view: (*left*) RGB Image. (*middle*) Safety map. (*right*) POD histogram.

During straight movement, obstacles to the right and left sides of the wheelchair do not explicitly affect the movement. However, when the wheelchair needs to initiate an avoiding action, such obstacles may become dangerous. As illustrated in Fig.5 (a), two people O_1 and O_2 may block the wheelchair movement. In this scenario opening V_1 meets a condition to be selected and the movement to the left is suggested. However, if the wheelchair proceeds with such action, clearly it will bump into O_1 even though the wide opening is available. To overcome such an issue, we search for *HRO* in the primary *POD* and block any movements towards such obstacles. For each hill in *HRO-POD* location, if the hill lies in the left-side or right-side of center *POD*, then *all* POD_b to the left of $HRO\text{-}POD_{left}$ or to the right of $HRO\text{-}POD_{right}$ should be high. As for normal POD (POD $(\omega_i,j)<\tau_d$), the POD_b is updated using equation (2). In the condition where the *HRO* is found only in front of the wheelchair, since it is not safe to perform any further movement we block all *PODs* to the left and to the right of this *HRO-POD*, and hence the wheelchair will stop moving. A sample of the generated POD is given in Fig. 5(b) where the RGB image, the safety map, and the POD image are shown in the left, middle and right, respectively. In the POD image, the yellow and green bars represent the low and high POD_b, respectively, while the red bars indicate *HRO-POD*.

$$POD_b(\omega_{i,j}) = \begin{cases} 1 & if \quad POD(\omega_{i,j}) > \tau_c \\ 0 & elsewhere \end{cases} \tag{2}$$

To determine the steering angle, we need to analyze the best opening in the POD_b that must be wide and safe (i.e. V_{wide}). Also, we must consider the user intention to moderate the motion signal to the wheelchair. For these reasons, we use a hierarchical strategy and set up a prioritize framework for selection of the future direction according to the goal, the user, and the safety in an orderly manner. If the goal angle is found in the wide opening as seen in Fig.6 (a), then it is selected to satisfy the goal-oriented requirement. Note that, the goal direction can always be set and reset by the user at any time by pushing and holding the switch as explained previously. When the goal requirement cannot be satisfied, i.e., when the goal is blocked, the wheelchair is under the collision/obstacle avoidance mode. In this situation, the next priority for selection of the candidates, V_{wide}, is based on the user's gaze direction (under this mode the user does not have to press the switch). Considering this, when the wheelchair is at the hallway T-intersection (i.e., Fig. 6 (b)), it can instantaneously steer to the user-driven direction. However when the user points at the direction that is subject to collision as shown in Fig. 6(c), the wheelchair neglects the gaze direction and steers to the safest zone. Once reached the zone, the wheelchair then fixate its orientation towards the goal.

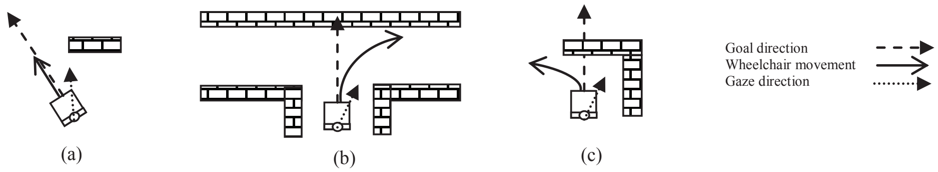


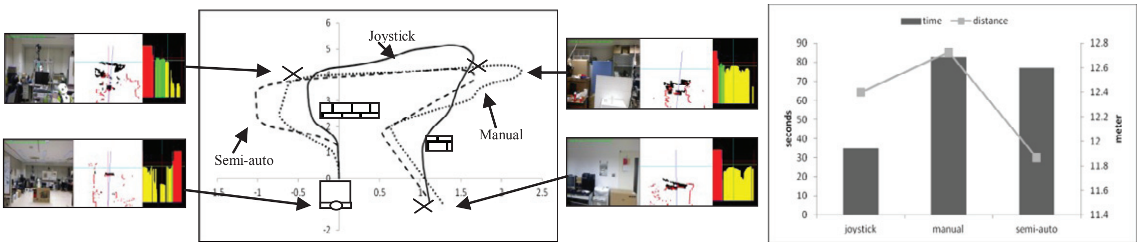
Fig. 6. Possible occurrences during navigation. (a) Free maneuver. (b) Hallway T-intersection. (c) Avoiding obstacles.

However, if the user presses the switch button while navigating, VFH algorithm still assists the user in such a way that it terminates any user command that may hurt him/her. For example if the user commands the wheelchair to the right in such a case as shown in Fig. 6(c), the computer overrides its command and brakes the wheelchair. However in this mode, it does not steer away as the behavior found in the semi-autonomous control; instead it keeps stopping until the user issues a safe command.

6. Experimental Results

In this section we analyze the system performance in real operation environments. The first experiment was conducted to evaluate the system ability for responding to the user command via the designed HCI. A user drove the wheelchair into three predefined locations in the cluttered lab environment by the full manual control and the semi-auto control. For the sake of comparison, the user was also asked to control the wheelchair by means of a standard joystick.

The resultant paths are shown in Fig. 7(a) where the checkpoints are marked by 'X'. Apparently we can see that by using all interfaces, the user is able to reach the checkpoints and successfully completed the maneuvering task without collision. In terms of task completion time, as given in Fig. 7(b), the joystick is the fastest, and both manual and semi-autonomous modes required almost double of the joystick case time. This may happen because, unlike the joystick, both mediums required a bit of time to reinitiate the user's command and re-orientate to the desired direction. As for the distance of travel, generally there were no significant differences among all mediums although the semi-autonomous mode showed shorter distance of travel than the other two. This indicates that the user can reach his/her goal by using the proposed HCI in almost the same travel distance as by using the joystick. This also shows its reliability as an alternative interface.



(a) (b)

Fig. 7. Sample paths by three different modes.

One way to measure the system performance is using the complexity of operating the wheelchair. It can be measured by counting the time that the user needs to interact with the system to complete the task. Apparently, the semi-autonomous mode requires less effort from the user than the manual since s/he needs to interact only when wishing to change the direction to travel. From our testing, in the semi-autonomous mode the user only needs to issue as low as one third commands compared to the manual mode. Along with less workload, the semi-autonomous mode still gives freedom to the user for changing the path in a simple and natural way. Therefore it is a better option, provided that the "manual part" is easy to use.

In the second experiment we investigated the feasibility of the safety map; whether the map can prevent the wheelchair from collision in the manual mode and steer away from obstacles in the semi-autonomous mode. In Section 4 we described how the sensor integration can be beneficial for producing the safety map. As given in Fig. 8 we can see the differences in the safety map and POD histogram when we use only the laser and when we combine the laser with the Kinect. In *scenario 1*, since the laser is unable to correctly detect the whole table shape, the resultant POD suggests a motion towards the table, which is dangerous to the user. However, when integrated with the Kinect, the POD successfully blocks any movement towards the table and provides a steer clear motion to the left. The same situation happens in *scenario 2* where the combined sensor data successfully hinder the wheelchair from colliding with the iron rack, whereas the laser still suggests the next motion. These examples provide insight for the versatility to fuse multiple sensors to visualise the surrounding more properly compared to relying on a single perception tool.

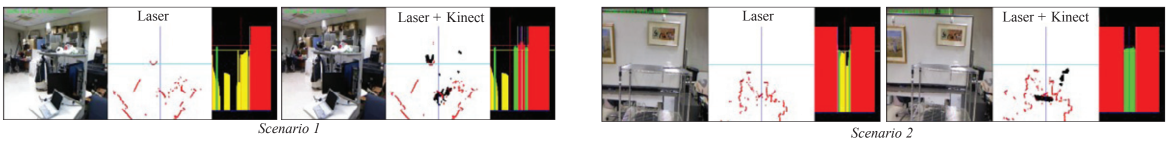
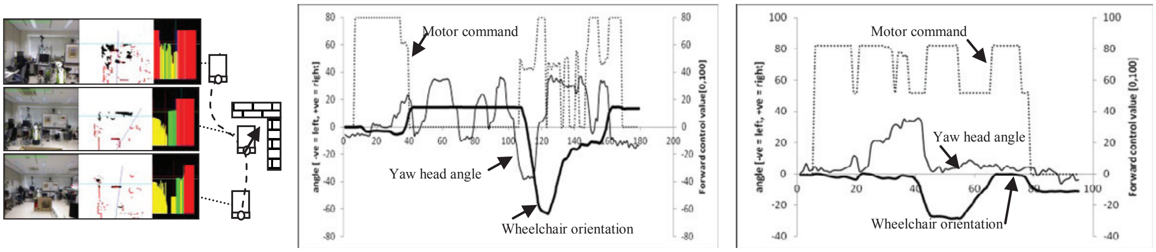


Fig. 8. Two scenarios to illustrate the resultant safety maps and POD histograms when using only the laser, and when combining the laser and the Kinect.

To evaluate the safety map, the user steered the wheelchair through the environments as illustrated in Fig. 9(a) and intentionally issued a command that would be subject to collision. In the manual mode, as shown in Fig. 9(b), the driving signal that was sent to the wheelchair usually followed the user's gaze direction (e.g., when the user looked to the left, the wheelchair also turned left). However as seen in frames #40 to #100, the system did not execute the user's command (i.e., motor command = 0) since it was not safe and the wheelchair remained stationary. Later on, the user commanded to turn left (frames #100 to #110). Since the steering command showed no possibility to collision, the system properly responded and executed it. Such a verification process is continuously carried out as long as the wheelchair is under the manual mode.

The wheelchair response may be a bit different in the semi-autonomous mode since the head direction is effective only when facing with an obstacle (i.e., the goal being blocked). In this mode, the direction to avoid an obstacle will also closely follow the gaze direction. As shown in Fig. 9 (c), when the wheelchair faced with an obstacle in frame #40, the user gave suggestion to the right. However, since it was dangerous, the planner rejected the command and instead steered to the left to maintain safety. Once reached an appropriate space, in frames #50 to #70, the wheelchair corrected the heading direction to face the goal location. This result highlights the benefit of the semi-autonomous controller for providing help to the user by automatically passing through obstacles safely while seeking the goal. Besides that, when it is appropriate, the planner can move the wheelchair by following the user preference while avoiding obstacles.

For real implementation, execution time is crucial since the system needs to respond as soon as any threat is detected. For our system, the vision part requires around 83ms for accomplishing the 3D data mapping. By integrating the IMU sensor and the laser range finder, overall execution time increases since the laser requires 25ms to complete a cycle of scanning and the IMU required 5ms acquisition time. To sum up, the execution time of the system is 113ms, that is, about 8 frames per second. For the current application, this processing speed is enough for the wheelchair to receive, evaluate, and execute the user command promptly.



(a) (b) (c)

Fig. 9. System evaluation for avoiding collision. (a) Environment snapshots. (b) Manual mode. (c) Semi-autonomoud mode.

7. Conclusion

In this paper, we have proposed a smart wheelchair framework for catering the need of users with severe motor disabilities. The wheelchair has two control modes: the manual mode and the semi-autonomous mode. The user can change the modes freely by manual switch operation. By incorporating the safety map, the wheelchair can avoid collision in both modes, and hence can reduce the user's burden of continuously monitoring the surrounding while maneuvering. Operation experiments in actual cluttered environments have confirmed that the wheelchair can move efficiently without any collision.

In future work, we will evaluate the system with more participants to gain the generality of the overall system reliability. Moreover, we will measure the cognitive complexity of users by introducing extra work load during navigating.

Acknowledgments

This paper was supported in part by JST PRESTO, A-Step and KAKENHI (22243037, 24700157)

References

[1] Deal, B., Grove, A., 1965. General Relationship for the Thermal Oxidation of Silicon, Journal of Applied Physics 36, p. 3770.

[1] Simpson RC. Smart wheelchairs: a literature review. *J of Rehab. Research and Development* 2005; **42(4)**: 423–436.

[2] Cowan RE, Fregly BJ, Boninger ML, Chan L, Rodgers MM, Reikensmeyer DJ. Recent trends in assistive technology for mobility. *J of NeuroEngineering and Rehab.* 2012;**9**:20

[3] Simpson RC. Lo Presti EF, Cooper RA. How many people would benefit from a smart wheelchair?. *J of Rehab. Research and Development* 2008; **45(1)**: pp. 53–72.

[4] Iezzoni L, McCarthy E, Davis R, Siebens H. Mobility difficulties are not only a problem of old age. *J Gen Intern Med.* 2001; **16(4)**:35-43.

[5] Nisbet PD. Who's intelligent? Wheelchair, driver or both?. IEEE Intl. Conf. on Control and App. 2002; 760–765.

[6] Bates R. A computer input device selection methodology for users with high-level spinal cord injuries. The 1st Cambridge Workshop on Universal Access and Assistive technology 2002.

[7] Rockland RH, Reisman S. Voice activated wheelchair controller. IEEE 24th Annual Bioengineering Conf. 1998. 128–129.

[8] Blatt R, Ceriani S, Dal Seno B, Fontana G, Matteucci M, Migliore D. Brain control of a smart wheelchair. Intl. Conf. on Intelligent Autonomous System 2008; 221–228.

[9] Kuno Y, Yoshimura T, Mitani M, Nakamura A. Robotic wheelchair looking at all people with multiple sensors. IEEE Conf. on Multisensor Fusion and Integration for Intelligent System 2002; 341–346.

[10]Matsumoto Y, Ino T, ogasawa T. Development of Intelligent Wheelchair System with Face and Gaze Based Interface. IEEE Intl. Workshop on Robot and Human Communication 2001; 262-267.

[11]Hollands MA, Patla AE, Vickers JN. Look where you're going! : gaze behaviour associated with maintaining and changing the direction of locomotion. *Experimental Brain Research 2002*; 14: 221–230.

[12]Batolein C, Wagner A, Jipp M, Badreddin E. Easing Wheelchair Control by Gaze-based Estimation of Intended Motion. The 17th World Congress the Intl. Federation of Automatic Control 2008; 9162-9167.

[13]Reis LP, Braga RAM, Sousa M, Moreira AP. IntellWheels MMI: A flexible interface for an intelligent wheelchair. LNAI 2009; **5949**: 296–307.

[14] Fehr L, Langbein W, Skaar S. Adequacy of power wheelchair control interfaces for persons with severe disabilities: A clinical survey. *J of Rehabilitation Research and Development* 2000; **37**(3):353–360.

[15] Wan LM, Tam E. Power wheelchair assessment and training for people with motor impairment. 12th Intl. Conf. on Mobility and Transport for Elderly and Disabled Person 2010.

[16]Perrin X. Semi-autonomous navigation of an assistive robot using low throughput interfaces. *PhD Thesis ETH Zurich 2009*.

[17]Carlson T, Demiris Y. Increasing robotic wheelchair safety with collaborative control: Evidence from secondary task Experiments. In Proc. Of Intl. Conf. on Robotics and Automation 2010; 5582–5587.

[18]Levine SP, Bell DA, Koren Y. NavChair: An example of a shared-control system for assistive technologies. In Proc. of Intl. Conf. on Computers for Handicapped Persons 1994.

[19]Gomi T, Griffin A. Developing intelligent wheelchairs for the handicapped. Assistive Technology and Artificial Intelligence 1998; **1458:**150-178.

[20]Parokh S, Grassi V, Kumar V, Okamoto J. Usability study of a control framework for an intelligent wheelchair. In Proc. of Intl. Conf. on Robotics and Automations 2005; 4745–4750.

[21]Lopes AC, Nunes U, Vaz L. Assisted navigation based on shared-control using discrete and sparse human-machine Interfaces. Annual Intl. Conf. of the IEEE Engineering in Medicine and Biology Society 2010; 471–474.

[22]http://www.seeingmachines.com

[23]Tomari R, Kobayashi Y, Kuno Y. Multi-view head detection and tracking with long range capability for social navigation planning. LNCS 2011; **6939(II)**: 418–427.

[24]Ulrich I, Borenstein J. VFH+: Reliable obstacle avoidance for fast mobile robots. IEEE Intl. Conf. on Robotics and Automation 1998; 1572–1577.