2014 IEEE International Conference on Robotics & Automation (ICRA)
Hong Kong Convention and Exhibition Center
May 31 - June 7, 2014. Hong Kong, China

Descending Stairs Locomotion and Somatosensory Control for An Erect Wheel-Legged Service Robot

Ren C. Luo, Ming Hsiao, Che-Wei Liu

***Abstract*— In this paper, an erect stair climbing robot, which has been able to ascend stairs, is further developed for descending stairs automatically. The somatosensory control functions are also implemented for a human rider to drive the robot on flat ground as a personal vehicle instinctively. In order to face front whether climbing up or down stairs, the robot is designed with proper mechanical structure and mass distribution. Moreover, the perception and control methods of descending stairs are modified from the methodologies of ascending stairs in our previous works so that the robot can handle both ascending and descending stairs effectively under similar principles. As for the somatosensory control part, the functions of controlling the robot to move back and forth using the rider's body motion is realized, which has more advantages than Segway PT since it can stand still when the rider wants the platform to stop moving, and the robot motion under the control of the human rider is stable enough for indoor applications. In the experiments, the stair-climbing abilities and somatosensory control functions are both successfully demonstrated, which proves that the design and implementation of the robot system are feasible and efficient. We are looking forward for more functions and applications of the robot, including our ultimate goal of carrying a human rider to climb stairs in an erect way with somatosensory control simultaneously.**

I. Introduction

Stair climbing is crucial for service robots or personal vehicles to travel around indoor environment. Many solutions has been discussed, including mechanical design of mobile platforms, perception of terrains, and locomotion planning for better stability and efficiency during stair climbing procedures. In our previous work, we developed an erect stair climbing robot for indoor service applications [1]. Our robot includes a pair of rotational triangular modules, a main body for a human rider to stand on it, and an erect structure with sensors, computing devices, and batteries on it (as shown in Fig. 1). Since the system has integrated effective control strategies and perception methods, the mobile robot has demonstrated the capability of climbing up stairs automatically. However, complete stair climbing functions should include both ascending and descending stairs. Therefore, methodologies to accomplish stair descending function will be introduced in this paper.

Ren C. Luo is with the Center for Intelligent Robotics and Automation Research, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan 106 (corresponding author to provide phone: +886-2-3366-9822; e-mail: renluo@ntu.edu.tw).

Ming Hsiao is with the Center for Intelligent Robotics and Automation Research, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan 106 (e-mail: mhsiao@ira.ee.ntu.edu.tw).

Che-Wei Liu is with the Center for Intelligent Robotics and Automation Research, National Taiwan University, No. 1, Sec. 4, Roosevelt Road, Taipei, Taiwan 106 (e-mail: jwliou@ira.ee.ntu.edu.tw).

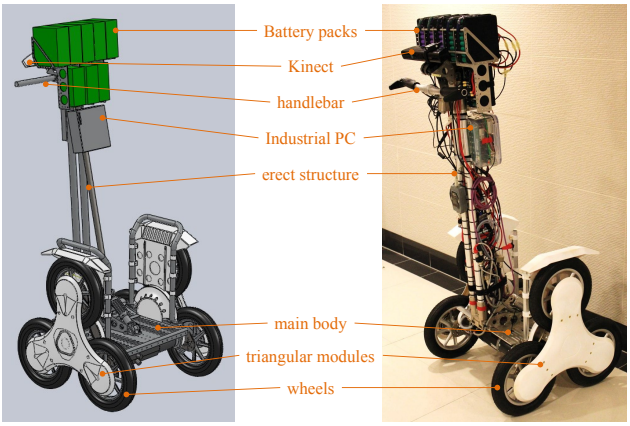


Fig. 1. The general structure of the erect stair-climbing robot

Several research topics has solved the problem of robot climbing down stairs in various ways. Among all, biped robots are the most discussed type of mobile platform that can descend stairs with well-designed control strategies [2][3][4]. Other mobile robots with various mechanical structures are also described in [5][6][7], which includes common track-based robots and some other wheel-legged mobile robots. However, these works cannot offer enough information for us to realize the stair descending function on our robot. Because the center of mass of our robot is much higher than any of these special designed mobile platforms, which results in more stability problems. Also, the stable step-by-step procedure of the biped robots walking down stairs cannot be implemented on our robot either because the degrees of freedom of its locomotion mechanism are much less than biped robots. As a result, our goal is to find a solution for our robot to descending down stairs in a step-by-step but dynamic way.

In addition, since our erect stair climbing robot is designed as an indoor personal vehicle as well, the functions for the rider to control the motion of the robot is required. For convenience, intuitive operation, and safety reasons, we decide to adopt somatosensory control method to realize the function. The most famous example of somatosensory-controlled vehicles is Segway PT [8]. A human rider can stand on the platform of Segway, hold its handles, and control the motion of the vehicle through body tilting. Therefore, another goal is to plan the reaction of the robot to the motion of the rider so that the rider can feel as if he or she is riding and controlling a Segway, but more stable and safe.

978-1-4799-3684-7/14/$31.00 ©2014 IEEE

II. System Structure

A. Mechanism

The erect stair climbing robot contains a main body, a pair of rotational triangular modules, and an erect structure (see Fig. 1). The surface of the main body is designed with proper space of a human rider to stand on it while riding it. Each triangular module has three active and synchronous wheels with pneumatic tires, which are linked together and driven through gears, sprocket wheels, and chains. A pair of adjustable spring assemblies with pin-and-slot structures are placed in the transmission system of the rotational triangular modules, which enables the rider to tilt the main body together with the erect structure back and forth $\pm 5^\circ$ referring to the vertical direction. A handle bar is on the erect structure for the rider to hold and control the motion of the robot. As for the integration of the devices and the computing units, refers to our previous work [1].

B. Electrical Devices

Inside the main body of the robot are four servo motors with two types of gearheads. Two of the motor sets with the same higher reduction ratio drive the two triangular modules respectively. Similarly, the other two motor sets with the same lower reduction ratios drive the wheels on the two sides of the robot (each side with three wheels) respectively. Therefore the total active degrees of freedom of the robot are four. A Kinect sensor is placed at the top front of the erect structure to get visual data from the surrounding environment. There are two MotionNode inertial measurement units (IMU) in the robot system. One is attached on the top of the Kinect, which can help rectify the observation angle of the sensor. The other is inside the main body to detect the pose and motion of the robot. As for the integration of the devices and the computing units, please again refer to our previous work [1].

III. Locomotion Principles of Descending Stairs

A. Concepts

As shown in Fig. 2, the concept of the method of descending stairs is that the robot rotates both of its triangular modules (cyan) to climb down stairs and keep balance at the same time. The active wheels (green) on the vertices of the triangular modules can adjust the position of the robot to the stair steps and also help keep balance of the robot and the rider (if there is one). The whole idea is similar to the locomotion of climbing up stairs. However, there are some differences and extra challenges when we implement the stair descending locomotion. Followings are the discussions and analysis.

B. Dynamic Analysis and Comparison

Even though we apply the same planer inverse pendulum model, zero moment line method together with virtual slope concept [9], and the idea of "capture point" [10] to analyze and realize the locomotion of descending stairs as we discussed about the locomotion of ascending stairs in

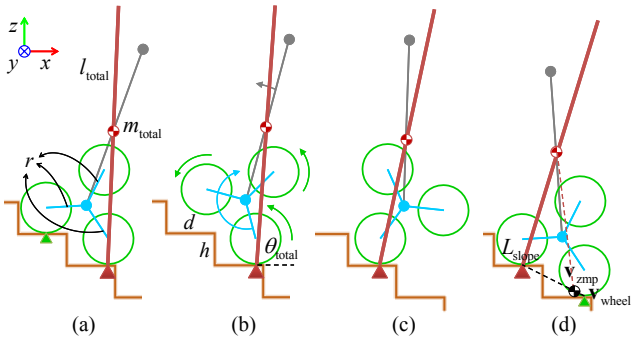


Fig. 2. The dynamic model of descending stairs

our previous work [1], the parameters can be different and result in stability problems. As shown in Fig. 3, we still use a uniform mass stick with length l_{total} and mass m_{total} to model the total mass of the robot, and therefore the dynamic model of the robot climbing down a step of stair can be written as a falling inverse pendulum:

$$I_{\text{total}}\ddot{\theta}_{\text{total}} = -m_{\text{total}} g \frac{l_{\text{total}}}{2} \cos\theta_{\text{total}}, \tag{1}$$

where θ_{total} is the angle of the stick with respect to horizontal direction, g is the gravitational acceleration constant, and I_{total} is the equivalent rotational inertia of the whole robot (also include the rider if there is one), which can be approximated as a rod of length l_{total} and uniform mass m_{total} with axis of rotation at the end of the rod even though I_{total} can change slightly during climbing.

Until now, the analysis are almost the same as ascending stairs. But unfortunately, because of the structure of the stair and the triangular module, the stable range for virtual zero moment point (ZMP) while descending stairs is smaller than in ascending stairs under the same falling speed of the inverse pendulum model. The main reason is that the virtual slope lines are different in the two cases. We can extend from our previous analysis (equation (2)-(6) in [1]) that the virtual zero moment line

$$L_{\text{zmp}} : x_{\text{zmp}} = x_{\text{cog}} + \frac{\ddot{x}_{\text{cog}}}{\ddot{z}_{\text{cog}} + g}(z_{\text{zmp}} - z_{\text{cog}}), \tag{2}$$

and the virtual slope line of descending stairs

$$L_{\text{slope}} : z = -\frac{h}{\sqrt{3r^2 - h^2}} x \tag{3}$$

intersect with each other at the point

$$(x_d, z_d) = \left(\frac{x_{\text{cog}} - k_a z_{\text{cog}}}{1 + k_a k_s}, \frac{-k_s x_{\text{cog}} + k_a k_s z_{\text{cog}}}{1 + k_a k_s}\right) \tag{4}$$

where $k_a = \frac{\ddot{x}_{\text{cog}}}{\ddot{z}_{\text{cog}}+g}$ and $k_s = \frac{h}{\sqrt{3r^2-h^2}}$ are the simplified notations of the complex items in the results. The intersection point (x, z) is actually the ZMP point of the robot in the descending case. Comparing with the virtual ZMP point of the robot in the ascending case with the same stair rise:

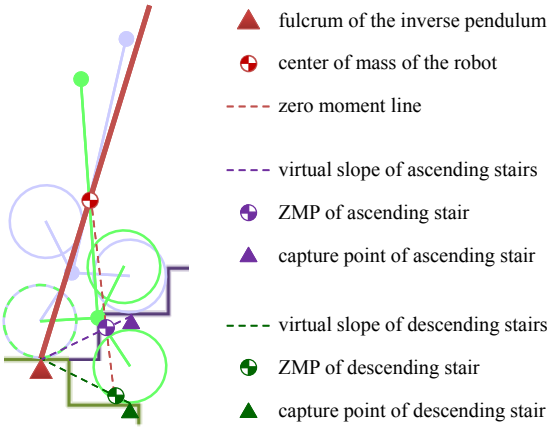


Fig. 3. Comparison of virtual ZMP points in the two cases of locomotion

$$(x_a, z_a) = \left(\frac{x_{\text{cog}} - k_a z_{\text{cog}}}{1 - k_a k_s}, \frac{k_s x_{\text{cog}} - k_a k_s z_{\text{cog}}}{1 - k_a k_s}\right) \tag{5}$$

we can find that the virtual ZMP of the robot in the descending case is further away from the fulcrum point than it is in the ascending case since $k_a < 0$ and $k_s > 0$ are always true. We can also observe the effect from the geometry as shown in Fig. 3. Therefore, descending stair is actually more difficult than ascending stairs for this robot.

C. Simulation

The simulation of descending stairs goes similar as ascending stairs (Fig. 3 in [1]), while the only difference is that the stable time range of falling becomes $t < 600$ms if the initial tilting angle of the stick model is set as 88°. Therefore, the average rotation speed of the triangular modules have to be larger than $\frac{1/3}{0.6} \times 60 \doteqdot 33(\text{rpm})$, which is a little faster than it is in the case of ascending stairs, but still within the speed limit of the adopted motors.

IV. Implementation of Descending Stairs

A. Mechanical Constraints and Solutions

In order to ascend as well as descend stairs, the mechanical structure of the robot has to match with several requirements. First of all, the mass distribution together with the shape of the chassis of the main body and the triangular modules have to enable the robot to move its center of mass back and forth for a certain range without colliding into various terrains. As a result, we place the heavy batteries high up on top of the erect structure so that the robot can change the position of its center of mass by slightly rotating the two triangular modules together. As shown in Fig. 4-a, when the robot is climbing up stairs, the center of mass has to move beyond the front fulcrum point on the front wheels while the front part of the chassis can still keep a safe distance form colliding into the next stair step (red dashed line). Similarly, during the stair descending procedure (see Fig. 4-b), the center of mass of the robot should be kept within the stable range between the front and back wheels that touch two adjacent stair steps

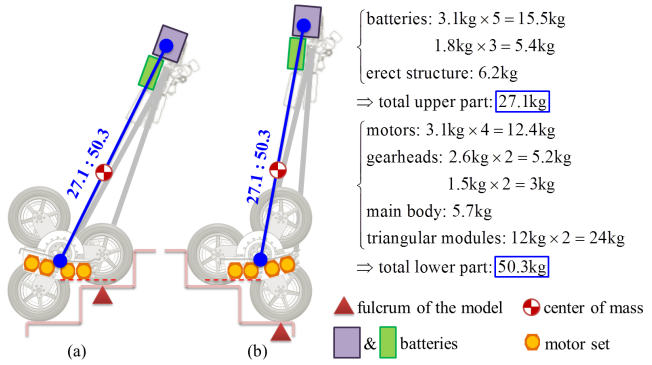


Fig. 4. Mechanical constraints on the center of mass and the shape of chassis

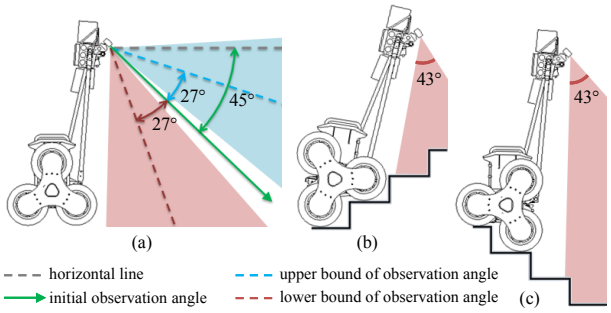


Fig. 5. Available observation angles of the Kinect on the robot for various cases

while the back part of the chassis should also keep away from colliding into the previous stair step (red dashed line).

Another problem is the observation angle of the sensor in descending cases. The Kinect has to look down a certain angle to observe as large stair surfaces as possible for better parameters measurement, while it should also be able to observe horizontal direction in normal applications. From the specifications of Kinect, we know that the vertical open angle of its infrared depth sensor is 43°, and rotation range of the small motor that control the tilting angle of Kinect is ±27°. Therefore, the base of the Kinect is set to be 45° leaning forward (see Fig. 5) so that when the Kinect rotates its neck to look up, enough horizontal scene can be covered within the vertical open angle of observation of Kinect. And also, when the Kinect rotates its neck to look down, its view can cover even the area near its front wheels, which can give the robot sufficient information about the terrain right in front of it.

B. Perception Methods

The perception of stairs is another challenge when the robot is going to climb down stairs. Unlike climbing up stairs, the observation angle and space of the stairs are both much more limited. Even if the robot moves to the edge of a downward staircase, upper steps can still occlude the lower steps because of the limited observation angle. Other researchers have discussed about this problem in [11][12]. However, since we have already develop the system to

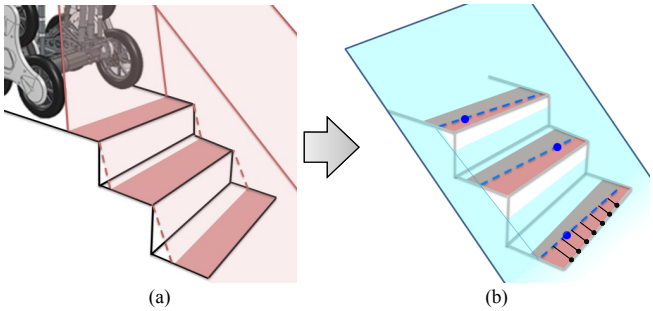


Fig. 6. (a) Limited observation and (b) the applied RANSEC method to find a slope plane to represent the general tendency of the stair

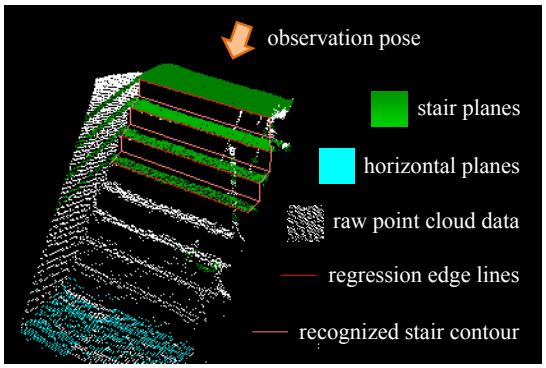


Fig. 7. Result of downward stair recognition

recognize stairs and measure their parameters during stair ascending in our previous work [13], we just have to modify it for descending stairs.

To recognize a downward stair and measure its parameters from lateral top view, we first rectify the observation angle of the sensor on the robot in the program during data preprocessing step, which uses the method of finding the cluster of possible normal vectors of horizontal surfaces all over the 3D scene. And then, the horizontal planes with enough area are marked up as possible surfaces of stair steps through analyzing the vertical distribution of the point cloud in the 3D scene. By far, the process is almost the same as it is in the method of recognizing upward stairs, only that the defined interval of valid planes is set to be broader since the observed surfaces of the stair step planes could be occluded by others (see Fig.6-a). Fortunately, because the step surfaces of a downward staircase are lower than ground level, which means that we can ignore all the structures locating above the ground and focus on the horizontal planes below the ground. In application, the height of ground can be roughly determined by the structure of the robot and the pose of the Kinect sensor.

However, the next step is different from the original method of finding edge points from each point cluster on the same plane because the occlusion problem results in the large variation of the observed width of the planes. Therefore, another idea is implemented in the latter part of the downward stair recognition method, which is to use a RANSAC method to find a slope plane to describe the general slope trend of the downward staircase, as shown in Fig. 6-b. Three points (blue dots) from three of the point clusters of possible stair surface planes are randomly picked out as the samples to form a new plane (cyan plane), which represents a possible slope trend of the whole stair structure. The process will be repeated again and again until the generated possible slope plane has reasonable slope of stairs and can fit enough raw point cloud data within certain distance. Once the slope is determined, we calculate the intersection lines (blue dashed lines) of the slope and all the possible stair step planes, and find the nearest sample edge points away from the intersection lines on the perpendicular direction (black lines and dots). These sample edge points can describe the rough contour of the edge of the stair steps, which are the most important features for a robot to recognize stairs and measure their parameters. For our application, the most crucial information for the robot to determine proper control parameters to climb down stairs are the rises and runs of the stair steps, which can be found from the relationships among the regression lines of the sample edge points. As a result, the robot can acquire necessary information of the stair through the above procedure before climbing down it (see Fig. 7).

C. Control Strategies

Suppose the parameters of the stairs measured by our method are precise enough (this will be verified in the latter section), the locomotion control parameters can be determined based on them. From the previous dynamic analysis, we know that the rise of the stair step that the robot is going to climb on can decide the rotation speed and angle of the triangular modules, and the run of the step determines whether there is enough room for the robot to put its wheels on it safely. To find out the required rotational speed ω_{motor} and angle θ_{motor} of the motors which drive the triangular modules, we apply the equations:

$$\theta_{\text{motor}} = \left(\frac{2\pi}{3} + \theta_{\text{initial}} + \arctan k_s \right) \times \gamma_{\text{total}} \tag{6}$$

and

$$\omega_{\text{motor}} = \theta_{\text{motor}} / t_{\text{limit}} \tag{7}$$

where θ_{initial} is the relative angle of the front branch of the triangular modules to horizontal direction, which is always positive in stair ascending cases but negative in descending cases. k_s is the parameter of the geometric relationship between the robot and the stair structure, which is as defined in Section III. γ_{total} is the cumulative reduction ratio from one motor to its corresponding triangular module. t_{limit} is the limit time range of recovering falling state to stable state and could be decided from simulations (as introduced Section III) or experiments.

Stairs will also be measured whether they are able to be climbed or not based on its parameters. The robot will only climb up or down stairs with their rises

$$0 < h < \frac{\sqrt{3}}{2} r \tag{8}$$

and also their runs

$$\frac{\sqrt{3r^2 - h^2} + r_{\text{wheel}} + d_{\text{s}}}{2} < d < 2\sqrt{3r^2 - h^2} - r_{\text{wheel}} - d_{\text{s}} \tag{9}$$

or

$$d > \sqrt{3r^2 - h^2} + \sqrt{3} r \tag{10}$$

where r_{wheel} is the radius of the active wheels, d_{s} is the buffer region between the touch point of the wheels to the edge of the step surface added for safety concern. Equation (9) describes the type of stairs that the robot can climb with the same successive motions, while (10) is the type of step with very large depth so that the robot has to climb it as if to move to another plane surface in stead of a stair structure.

V. Somatosensory Control Methodologies

In this section, we introduce the methods of how to design the control and motion of the robot when a human rider is interacting with it. Two spring assemblies are designed in the transmission system of the rotational triangular modules so that the rider can lean forward or backward freely within $\pm 5^\circ$. The robot has to move forward when the rider leans forward, move backward when the rider leans backward, and stop when the rider stand still straightly.

A. Sensor Data Processing

As described in section II, the system includes a set of IMUs to measure the motion of the main body of the robot. Because the sampling rate of the IMU sensors is 30Hz, which is much faster than human reaction, a filter can be applied to smooth the input sensory data without delaying the reaction of the robot too much to be noticed by the human rider. In our implementation, a simple filter of calculating the average within a fixed interval is used to smooth the acceleration and angular speed data from the IMU in the main body of the robot. Next step, we find the direction of the relative acceleration, which consists of current acceleration of the robot and gravity. Finally, the direction of the relative acceleration of the system is taken as the input to the control algorithm. The reason why we do not have to find the actual tilt angle of the main body is that both current pose and current motion of the robot have to be compensated by the response action of the robot determined by some control methods. Moreover, the data acquired from the IMU cannot be analyzed easily to find each of them respectively. That is:

$$\mathbf{a}_{\text{total}} = \mathbf{g} + \mathbf{a}_{\text{robot}} = a_x \mathbf{e}_x + a_y \mathbf{e}_y + a_z \mathbf{e}_z \tag{11}$$

where $\mathbf{a}_{\text{total}}$ is the relative acceleration of the whole system, $\mathbf{a}_{\text{robot}}$ is the acceleration of the robot, $\mathbf{g}$ is the gravity, and a_x, a_y, a_z are the measurements of acceleration from the IMU on the three main axis $\mathbf{e}_x$, $\mathbf{e}_y$ of its own coordinates. Obviously, it is hard to extract $\mathbf{g}$ and $\mathbf{a}_{\text{robot}}$ respectively since we do not know the direction of $\mathbf{g}$ and the value of $\mathbf{a}_{\text{robot}}$. Fortunately, the two factors can be handled together with their combinational form since they actually influence the system in the same way and their effects to the state of the system are cumulative. Thus, offering the information of the direction of the relative acceleration and the angular speed of the main body to the control block is sufficient for a complete sensory feedback control system.

B. Control Design and Parameter Tuning

A basic PD control with boundary limits is adopted for forward and backward motions as below:

$$\omega_{\text{wheel}} = k_p \theta_{\text{body}} + k_d \omega_{\text{body}} \tag{12}$$

where ω_{wheel} is the rotational speed of the active wheels that should be within the speed limits for safety concern. k_p and k_d are the constant parameters of proportional (θ_{body}) and derivative (ω_{body}) terms. θ_{body} is decided by the difference between the direction of acceleration and the vertical direction, while ω_{body} is the integrated result of the measured rotational speed from IMU and the difference of two body angles calculated in two successive system cycles. In addition, the robot will stand still if the tilting angle of the main body is between $\pm 1.5^\circ$. The whole method is simple but effective for our application.

VI. Experimental Results

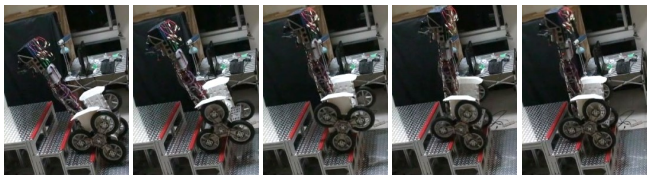
A. Procedures

The original mechanical design enables the robot to climb up and down every indoor stairs which match with the public building codes of our government. However, due to the limited size of the tires we could find, the wheels we adopted are slightly larger than they should be, and the robot cannot climb every indoor stairs because some of their stair runs are too small for the robot to place its reluctantly larger wheels safely on it. As a result, we built an experimental staircase with wider stair runs to fit the size of the actual wheels of the robot.

The first experiment is to retry the automatic stair ascending ability of the robot. After the original functions are tested well in the new experimental environment, we conduct the main experiment of the robot descending stairs on its own. At last, we fine tune the parameters of the somatosensory control system and test its performance with a human rider standing on the robot.

B. Results

Fig. 8 and Fig. 9 show the sequential photos of the robot climbing up and down the experimental stairs on its own. The time interval between each two images is about 0.12 second in both cases. Even though an aluminum frame with hanger is placed on the experimental stairs to prevent the robot from falling during the descending procedure, the robot does not have to be hung onto it in this experiment.



(a) (b) (c) (d) (e)

Fig. 8. The sequential photos of the robot climbing up a stairway

(a) (b) (c) (d) (e)

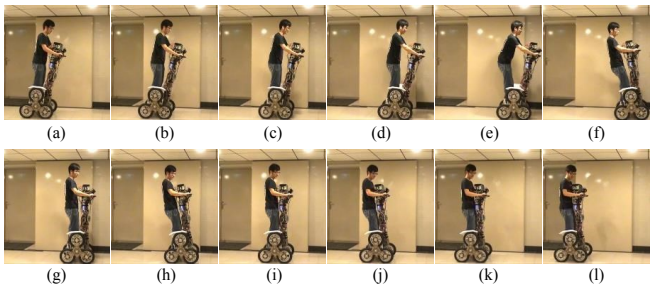
Fig. 9. The sequential photos of the robot climbing down a stairway

Fig. 10 shows the sequential photos of the robot under the body motion control of a rider. The time interval between each two images is about 0.67 second. The rider in the photos controls the robot to move forward and latter backward. In the duration the rider stops the robot in Fig. 10-g and 10-i.

C. Discussions

By far, several important functions of the erect stair climbing robot are completed. However, there are some details and more other advanced functions for us to discuss and develop. The experimental results are the same as we conclude in Section III, the stair descending process is actually more unstable than ascending stairs. There is only a small tolerance range for a successful descending motion, and therefore we have to tune the parameters in the control rules precisely. Other possible reasons to cause the unstable problem in the descending case are the effect of the spring assemblies and the release of potential energy. Even though the spring assemblies are necessary for the somatosensory control function, it might result in extra vibration of the robot during stair descending. Moreover, if the released potential energy is not well absorbed by the tires, the vibration will be even more severe. Therefore, we are planning to increase the stiffness of the springs and find suitable inner pressure for the tires.

Even though the rider can control the motion of the robot effectively through body leaning motions, the functions for the robot to become an indoor vehicle are not completed yet. We still have to add an angular sensor on the rotational axis of the handle bar and integrate it with the system so that the rider can rotate the handle bar to control the turning of the robot platform. Moreover, we have to pay more attention in our future works on how to decide when to switch the robot from flat ground traveling mode to stair climbing mode or switch back based on the comprehensive analysis of the command of the rider, the state of the robot platform, and the perceptual data from the environment.



(a) (b) (c) (d) (e) (f)

(g) (h) (i) (j) (k) (l)

Fig. 10. The sequential photos of body motion control of the robot

VII. Conclusions

In this paper, we develop an erect mobile robot system and realize its stair ascending and descending abilities with somatosensory control functions. The experimental results successfully demonstrate that the applied theories and methods are feasible and efficient. In the future, we will test more challenging motions of the robot, such as climbing spiral staircases, traveling on other complicated uneven terrains, or even carrying a human rider to travel across floors in a building. We will integrate more control methods and perception modules to make the robot safer and more robust when interacting with people.

References

[1] R. C. Luo, M. Hsiao, T.-W. Lin, "Erect wheel-legged stair climbing robot for indoor service applications," *IEEE International Conference on Intelligent Robots and Systems (IROS)*, 2013.

[2] C.-L. Shih, "Ascending and descending stairs for a biped robot," *IEEE Transactions on Systems, Man and Cybernetics*, 1999.

[3] G. Figliolini, M. Ceccarelli, M. Di Gioia, "Descending stairs with EP-WAR3 biped robot," *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2003.

[4] C. Fu, K. Chen, "Gait synthesis and sensory control of stair climbing for a humanoid robot," *IEEE Transactions on Industrial Electronics*, 2008.

[5] D. Campbell, M. Buehler, "Stair descent in the simple hexapod 'RHex'," *IEEE International Conference on Robotics and Automation (ICRA)*, 2003.

[6] G. Quaglia, W. Franco, R. Oderio, "Wheelchair.q, a mechanical concept for a stair climbing wheelchair," *IEEE International Conference on Robotics and Biomimetics (ROBIO)*, 2009.

[7] H. Mo, P. Huang, S. Wu, "Study on dynamic stability of a tracked robot climbing over an obstacle or descending stairs," *IEEE International Workshop on Electronic Design, Test and Applications*, 2006.

[8] Segway Inc. (2013). *The Segway PT: An Overview* [Online]. Available: http://www.segway.com/about-segway/learn-how-PTs-work.php

[9] T. Sato, S. Sakaino, E. Ohashi, K. Ohnishi, "Walking trajectory planning on stairs using virtual slope for biped robots," *IEEE Transactions on Industrial Electronics*, 2011.

[10] J. Pratt, J. Carff, S. Drakunov, A. Goswami, "Capture point: A step toward humanoid push recovery," *IEEE-RAS International Conference on Humanoid Robots*, 2006.

[11] J. Hesch, G. Mariottini, and S. Roumeliotis, "Descending-stair detection, approach, and traversal with an autonomous tracked vehicle," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010

[12] J. A. Delmerico, D. Baran, P. David, J. Ryde, J. J. Corso, "Ascending stairway modeling from dense depth imagery for traversability analysis," *IEEE International Conference on Robotics and Automation (ICRA)*, 2013.

[13] R. C. Luo, M. Hsiao, C.-W. Liu, "Multisensor integrated stair recognition and parameters measurement system for dynamic stair climbing robots," *IEEE International Conference on Automation Science and Engineering (CASE)*, 2013.