

Research Article

Experimental Studies of Neural Network Control for One-Wheel Mobile Robot

P. K. Kim and S. Jung

Intelligent Systems and Emotional Engineering (I.S.E.E.) Laboratory, Department of Mechatronics Engineering, Chungnam National University, Daejeon 305-764, Republic of Korea

Correspondence should be addressed to S. Jung, jungs@cnu.ac.kr

Received 18 July 2011; Revised 28 January 2012; Accepted 14 February 2012

Academic Editor: Haibo He

Copyright © 2012 P. K. Kim and S. Jung. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

This paper presents development and control of a disc-typed one-wheel mobile robot, called GYROBO. Several models of the one-wheel mobile robot are designed, developed, and controlled. The current version of GYROBO is successfully balanced and controlled to follow the straight line. GYROBO has three actuators to balance and move. Two actuators are used for balancing control by virtue of gyro effect and one actuator for driving movements. Since the space is limited and weight balance is an important factor for the successful balancing control, careful mechanical design is considered. To compensate for uncertainties in robot dynamics, a neural network is added to the nonmodel-based PD-controlled system. The reference compensation technique (RCT) is used for the neural network controller to help GYROBO to improve balancing and tracking performances. Experimental studies of a self-balancing task and a line tracking task are conducted to demonstrate the control performances of GYROBO.

1. Introduction

Mobile robots are considered as quite a useful robot system for conducting conveying objects, conducting surveillance, and carrying objects to the desired destination. Service robots must have the mobility to serve human beings in many aspects. Most of mobile robots have a two-actuated wheel structure with three- or four-point contact on the ground to maintain stable pose on the plane.

Recently, the balancing mechanism becomes an important issue in the mobile robot research. Evolving from the inverted pendulum system, the mobile-inverted pendulum system (MIPS) has a combined structure of two systems: an inverted pendulum system and a mobile robot system. Relying on the balancing mechanism of the MIPS, a personal transportation vehicle has been introduced [1]. The MIPS has two-point contact to stabilize itself. Advantages of the MIPS include robust balancing against small obstacles on the ground and possible narrow turns while three- or four-point contact mobile robots are not able to do so.

In research on balancing robots, two-point contact mobile robots are designed and controlled [1–5]. A series of balancing robots has been implemented and demonstrated their performances [3–5]. Currently, successful navigation and balancing control performances with carrying a human operator as a transportation vehicle have been reported [5].

More challengingly, one-wheel mobile robots are developed. A one-wheel robot is a rolling disk that requires balancing while running on the ground [6]. Gyrover is a typical disc-typed mobile robot that has been developed and presented for many years [7–11]. Gyrover is a gyroscopically stabilized system that uses the gyroscopic motion to balance the lean angle against falling as shown in Figure 1. Three actuators are required to make Gyrover stable. Two actuators are used for balancing and one actuator for driving. Experimental results as well as dynamical modeling analysis on Gyrover are well summarized in the literature [12].

In other researches on one-wheel robots, different approaches of modelling dynamics of a one-wheel robot have been presented [13, 14]. Simulation studies of controlling

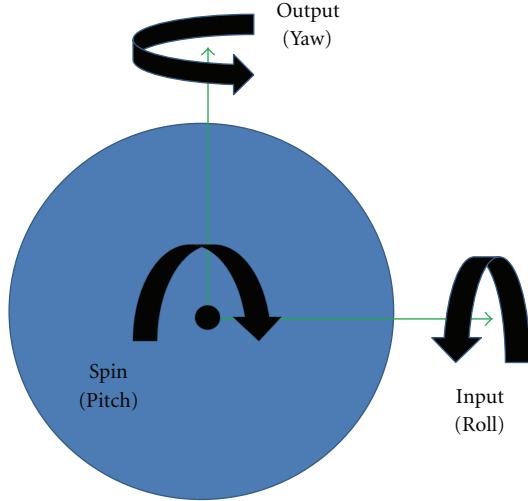


FIGURE 1: Gyro motion of one-wheel robot.

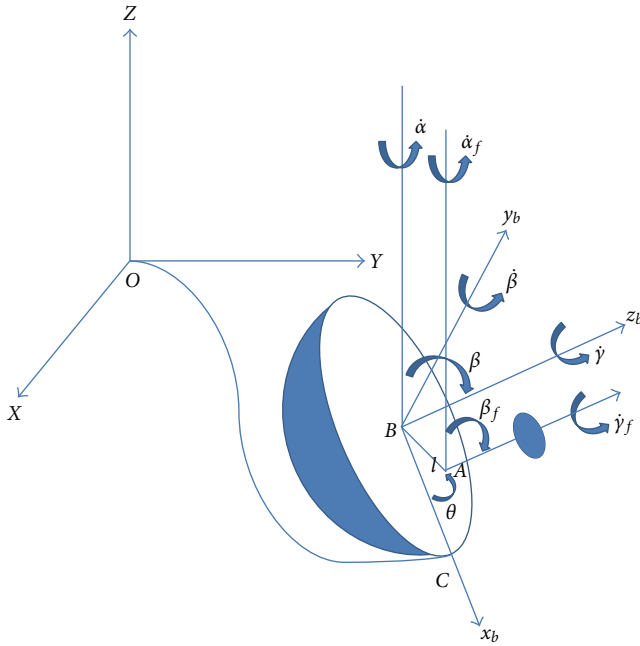


FIGURE 2: GYROBO Coordinates.

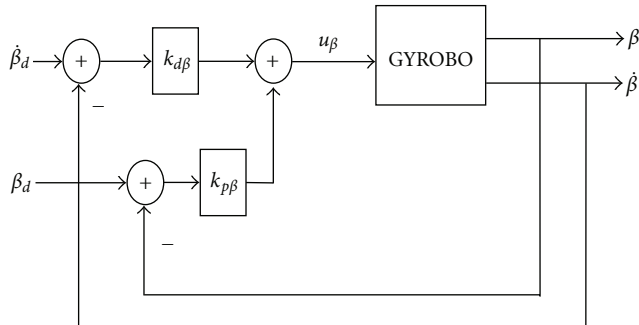


FIGURE 3: Lean angle control for balancing.

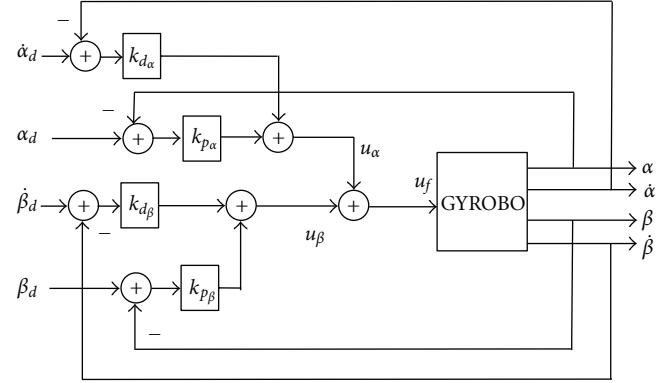


FIGURE 4: Straight line tracking control structure.

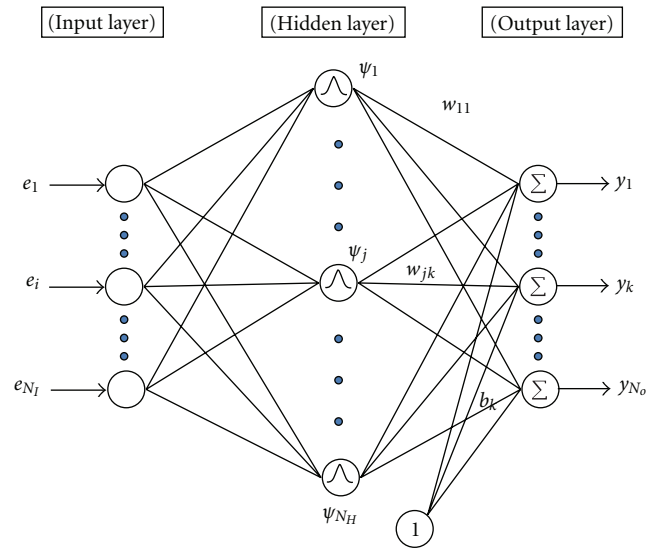


FIGURE 5: Neural network structure.

a Gyrobot along with design and fabrication are presented [15, 16]. An interesting design of a single spherical-wheel robot has been presented [17–19]. Successful balancing control of the spherical-wheel robot has been demonstrated.

In this paper, a one-wheel robot called GYROBO is designed, implemented, and controlled. GYROBO has three actuators to move forward and backward and to balance itself. Although aforementioned research results provide dynamics models for the one-wheel robot, in real physical system, it is quite difficult to control GYROBO based on dynamic models due to several reasons. First, modeling the system is not accurate. Second, it is hard to describe coupling effects between the body wheel and the flywheel since the system is nonlinearly coupled. Third, it is a nonholonomic system.

Therefore, we focus on optimal mechanical design rather than modeling the system since the dynamic model is quite complicated. Although we have dynamic models, other effects from uncertainties and nonmodel dynamics should be considered as well for the better performance.

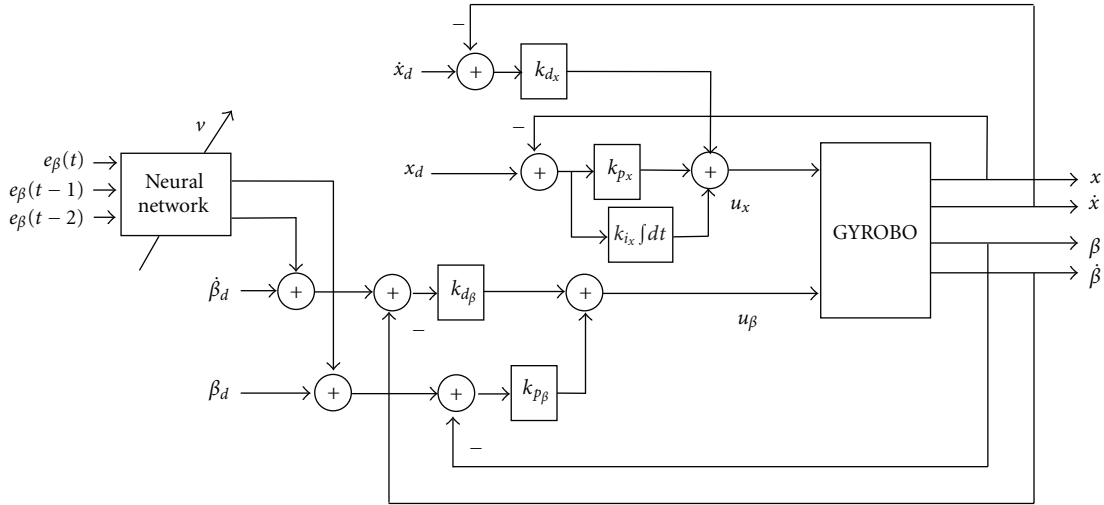


FIGURE 7: Sphere robot with links.

After several modifications of the design, the successful design and control of GYROBO are achieved. Since all of actuators have to be housed inside the one wheel, design and placement of each part become the most difficult problem.

After careful design of the system, a neural network control scheme is applied to the model-free-controlled system to improve balancing performance achieved by linear controllers. First, a simple PD control method is applied to the system and then a neural network controller is added to help the PD controller to improve tracking performance. Neural network has been known for its capabilities of learning and adaptation for model-free dynamical systems in online fashion [2, 20].

Experimental studies of a self-balancing task and a straight line tracking task are conducted. Performances by a PD controller and a neural controller are compared.

2. GYROBO Modelling

GYROBO is described as shown in Figure 2. Three angles such as spin, lean, and tilt angle are generated by three corresponding actuators. The gyro effect is created by the combination of two angle rotations, flywheel spin and tilt motion.

TABLE 1: Parameter definition.

α, α_f	Precession angle of a wheel and a flywheel
β	Lean angle of a wheel
β_f	Tilt angle of a flywheel
γ, γ_f	Spin angles of a wheel and a flywheel
θ	Angle between l and x_b axis
m_w, m_f	Masses of a wheel and a flywheel
m	Mass of whole body
R	Radius of a wheel
I_{xw}, I_{yw}, I_{zw}	Wheel moment of inertia about X, Y, and Z axes
I_{xf}, I_{yf}, I_{zf}	Flywheel moment of inertia about X, Y, and Z
g	Gravitational velocity
u_1, u_2	Drive torque and tilt torque
l	Distance between A and B

The dynamics of one-wheel robot system has been presented in [7, 11, 13]. Table 1 lists parameters of GYROBO. This paper is focused on the implementation and control of GYROBO rather than analyzing the dynamics model of the system since the modeling has been well presented in the literature [11].

Since the GYROBO is a nonholonomic system, there are kinematic constraints such that the robot cannot move in the lateral direction. GYROBO is an underactuated system that has three actuators to drive more states.

We follow the dynamic equation of the GYROBO described in [11]:

$$M(q)\ddot{q} + F(q, \dot{q}) = A^T \lambda + Bu, \quad (1)$$



FIGURE 8: Models of GYROBO using gyro effects.

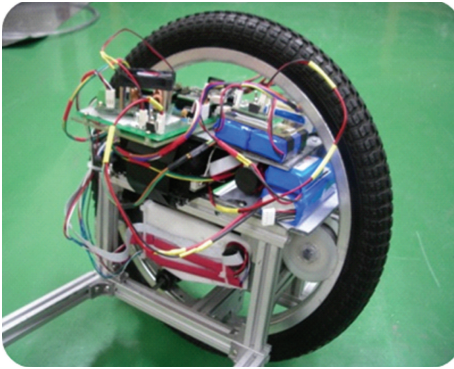


FIGURE 9: Real design of GYROBO I.

where

$$M(q) = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0 \\ 0 & m & 0 & 0 & 0 & 0 \\ 0 & 0 & M_{33} & 0 & I_{zw}S\beta & 0 \\ 0 & 0 & 0 & I_{xw} + I_{xf} + mR^2S^2\beta & 0 & I_{xf} \\ 0 & 0 & I_{zw}S\beta & 0 & I_{zw} & 0 \\ 0 & 0 & 0 & I_{xf} & 0 & I_{xf} \end{bmatrix},$$

$$M_{33} = I_{yw}C^2\beta + I_{zw}S^2\beta + I_{yf}C^2(\beta + \beta_f) + I_{zf}S^2(\beta + \beta_f),$$

$$F(q, \dot{q}) = \begin{bmatrix} 0 \\ 0 \\ F_3 \\ F_4 \\ I_{zw}C\beta\dot{\alpha}\dot{\beta} \\ F_6 \end{bmatrix},$$

(2)

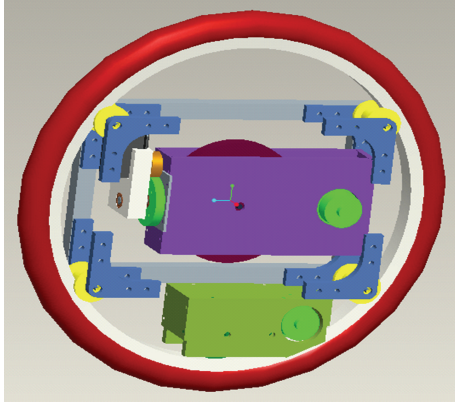
where

$$\begin{aligned} F_3 &= 2(I_{zw} - I_{yw})C\beta S\beta\dot{\alpha}\dot{\beta} + I_{zw}C\beta\dot{\beta}\dot{\gamma} \\ &\quad + 2(I_{zf} - I_{yf})C(\beta + \beta_f)S(\beta + \beta_f)(\dot{\beta} + \dot{\beta}_f)\dot{\alpha} \\ &\quad + I_{zf}C(\beta + \beta_f)(\dot{\beta} + \dot{\beta}_f)\gamma_f, \end{aligned}$$

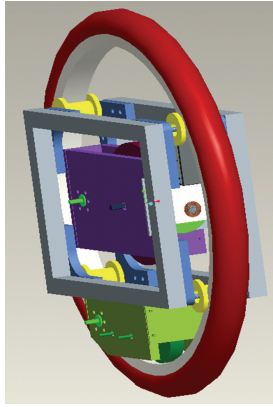
$$\begin{aligned} F_4 &= mR^2S\beta C\beta\dot{\beta}^2 + (I_{yw} - I_{zw})C\beta S\beta\dot{\alpha}^2 - I_{zw}C\beta\dot{\gamma}\dot{\alpha} \\ &\quad + (I_{yf} - I_{zf})C(\beta + \beta_f)S(\beta + \beta_f)\dot{\alpha}^2 \\ &\quad - I_{zf}C(\beta + \beta_f)\gamma_f\dot{\alpha} - mgRS\beta, \end{aligned}$$

$$F_6 = (I_{yf} - I_{zf})C(\beta + \beta_f)S(\beta + \beta_f)\dot{\alpha}^2 - I_{zf}C(\beta + \beta_f)\gamma_f\dot{\alpha}, \quad (3)$$

$$A = \begin{bmatrix} 1 & 0 & -RC\alpha C\beta & RS\alpha S\beta & -RC\alpha & 0 \\ 0 & 1 & -RC\beta C\alpha & -RC\alpha C\beta & -RS\alpha & 0 \end{bmatrix}, \quad (4)$$



(a)



(b)

FIGURE 10: Design of GYROBO II.

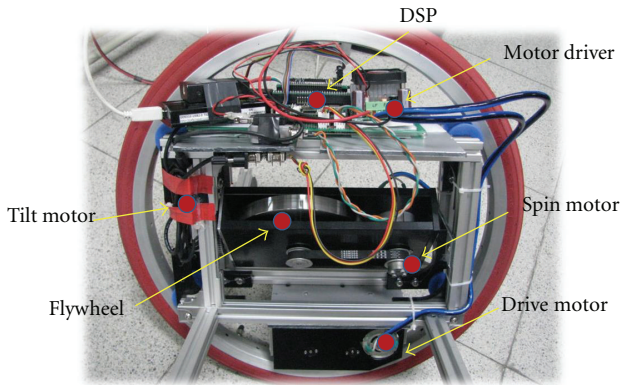


FIGURE 11: Real GYROBO II.

$$q = \begin{bmatrix} X \\ Y \\ \alpha \\ \beta \\ \gamma \\ \beta_f \end{bmatrix}, \quad \lambda = \begin{bmatrix} \lambda_1 \\ \lambda_2 \end{bmatrix}, \quad B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ k_1 & 0 \\ 0 & 1 \end{bmatrix}, \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}, \quad (5)$$

where $C\alpha = \cos(\alpha)$, and $S\alpha = \sin(\alpha)$.

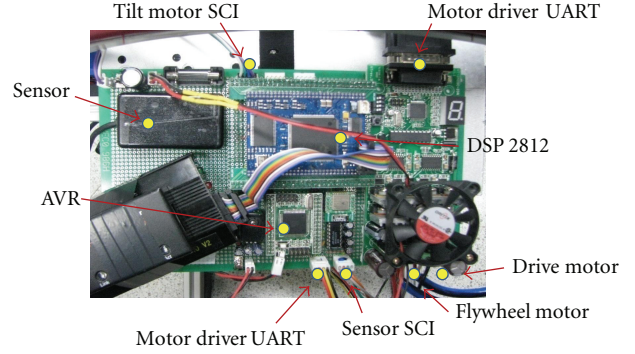


FIGURE 12: Control hardware.

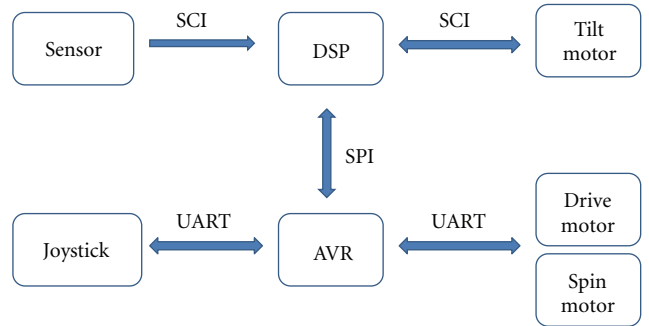


FIGURE 13: Control hardware block diagram.

3. Linear Control Schemes

The most important aspect of controlling GYROBO is the stabilization that prevents from falling in the lateral direction. Stabilization can be achieved by controlling the lean angle β . The lean angle in the y -axis can be separately controlled by controlling the flywheel while the flywheel rotates at a high constant speed. Spinning and tilting velocities of the flywheel induce the precession angle rate that enables GYROBO to stand up. Therefore, a linear controller is designed for the lean angle separately to generate suitable flywheel tilting motions.

3.1. PD Control for Balancing. The PD control method is used for balancing of GYROBO. The lean angle error is defined as

$$e_\beta = \beta_d - \beta, \quad (6)$$

where β_d is the desired lean angle value which is 0 for the balancing purpose and β is the actual lean angle. The angle error passes through the PD controller and generates the tilt torque u_β for the flywheel:

$$u_\beta = k_{p\beta} e_\beta + k_{d\beta} \dot{e}_\beta, \quad (7)$$

where $k_{p\beta}$ and $k_{d\beta}$ are controller gains. Figure 3 shows the PD control block diagram.

3.2. Straight Line Tracking Control. For the GYROBO to follow the straight line, a torque for the body wheel and a torque

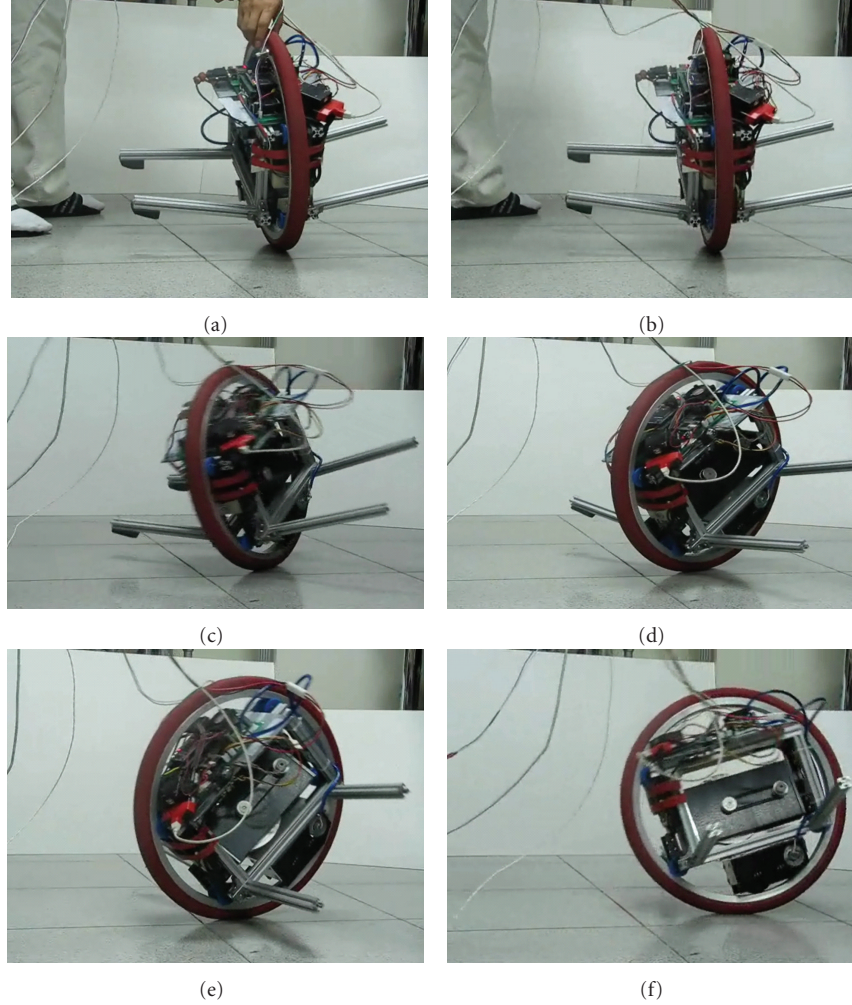


FIGURE 14: PD control: balancing task (a-b-c-d-e-f in order).

for the flywheel should be separately controlled. The body wheel rotates to follow the straight line while the lean angle is controlled to maintain balancing.

The position error detected by an encoder is defined by

$$e_p = x_d - x, \quad (8)$$

where x_d is the desired position value and x is the actual position. The detailed PID controller output becomes

$$u_x = k_{px}e_x + k_{dx}\dot{e}_x + k_{ix} \int e_x dt, \quad (9)$$

where k_{px} , k_{dx} , and k_{ix} are controller gains.

Thus line tracking control requires both position control and angle control. The detailed control block diagram for the straight line following is shown in Figure 4.

4. Neural Control Schemes

4.1. RBF Neural Network Structure. The purpose of using a neural network is to improve the performance controlled by linear controllers. Linear controllers for controlling

GYROBO may have limited performances since GYROBO is a highly nonlinear and coupled system.

Neural networks have been known for their capabilities of learning and adaptation of nonlinear functions and used for nonlinear system control [20]. Successful balancing control performances of a two-wheel mobile robot have been presented [2].

One of the advantages of using a neural network as an auxiliary controller is that the dynamic model of the system is not required. The neural network can take care of nonlinear uncertainties in the system by an iterative adaptation process of internal weights.

Here the radial basis function (RBF) network is used for an auxiliary controller to compensate for uncertainties caused by nonlinear dynamics of GYROBO. Figure 5 shows the structure of the radial basis function.

The Gaussian function used in the hidden layer is

$$\psi_j(e) = \exp\left(-\frac{|e - \mu_j|^2}{2\sigma_j^2}\right), \quad (10)$$

where e is the input vector, $e = [e_1 e_2 \cdots e_{N_I}]^T$, μ_j is

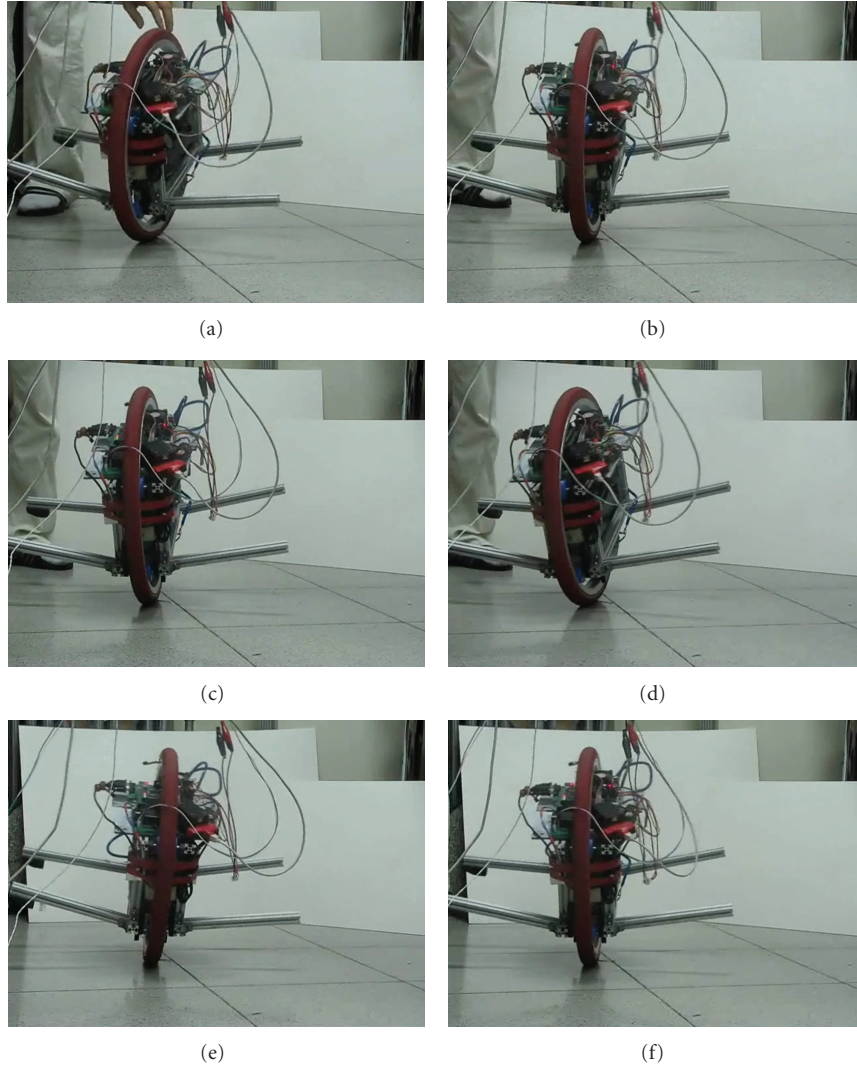


FIGURE 15: Neural network control: balancing task.

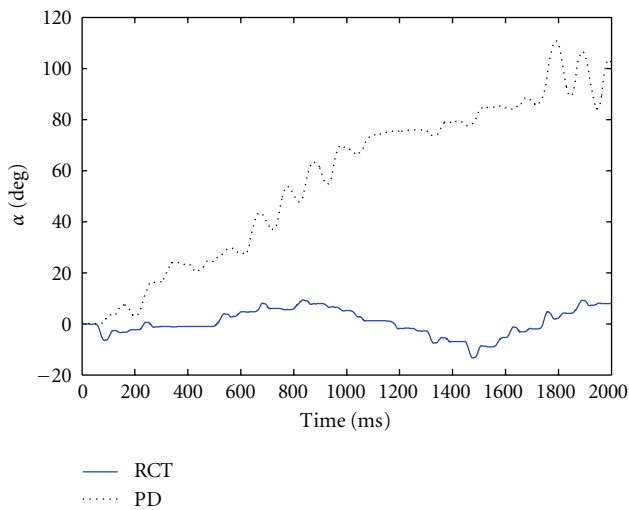


FIGURE 16: Heading angle comparison of PD (black dotted line) and RCT control (blue solid line).

the center value vector of the j th hidden unit, and σ_j is the width of the j th hidden unit.

The forward k th output in the output layer can be calculated as a sum of outputs from the hidden layer:

$$y_k = \sum_{j=1}^{N_H} \psi_j w_{jk} + b_k, \quad (11)$$

where ψ_j is j th output of the hidden layer in (11), w_{jk} is the weight between the j th hidden unit and k th output, and b_k is the bias weight.

4.2. Neural Network Control. Neural network is utilized to generate compensating signals to help linear controllers by minimizing the output errors. Initial stability can be achieved by linear controllers and further tracking improvement can be done by neural network in online fashion.

Different compensation location of neural network yields different neural network control schemes, but they eventually perform the same goal to minimize the output error.

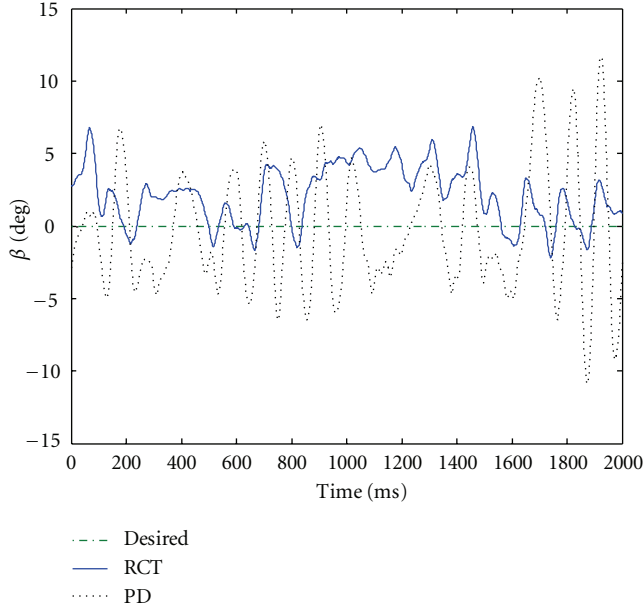


FIGURE 17: Lean angle comparison of PD (black dotted line) and RCT control (blue solid line).

The reference compensation technique (RCT) is known as one of neural network control schemes that provides a structural advantage of not interrupting the predesigned control structure [3, 4]. Since the output of neural network is added to the input trajectories of PD controllers as in Figure 4, tracking performance can be improved. This forms the totally separable control structure as an auxiliary controller shown in Figure 6:

$$u_\beta = k_{p\beta}(e_\beta + \phi_1) + k_{d\beta}(\dot{e}_\beta + \phi_2), \quad (12)$$

where ϕ_1 and ϕ_2 are neural network outputs.

Here, the training signal v is selected as a function of tracking errors such as a form of PD controller outputs:

$$v = k_{p\beta}e_\beta + k_{d\beta}\dot{e}_\beta. \quad (13)$$

Then (12) becomes

$$v = u_\beta - (k_{p\beta}\phi_1 + k_{d\beta}\phi_2). \quad (14)$$

To achieve the inverse dynamic control in (14) such as satisfying the relationship $u_\beta = \tau$ where τ is the dynamics of GYROBO, we need to drive the training signal of neural network to satisfy that $v \rightarrow 0$. Then the neural network outputs become equal to the dynamics of the GYROBO in the ideal condition. This is known as an inverse dynamics control scheme. Therefore learning algorithm is developed for the neural network to minimize output errors in the next section.

4.3. Neural Network Learning. Here neural network does not require any offline learning process. As an adaptive controller, internal weights in neural network are updated at each sampling time to minimize errors. Therefore, the selection of

an appropriate training signal for the neural network becomes an important issue in the neural network control, even it determines the ultimate control performance. The objective function is defined as

$$E = \frac{1}{2}v^2. \quad (15)$$

Differentiating (15) and combining with (14) yield the gradient function required in the back-propagation algorithm:

$$\frac{\partial E}{\partial w} = v \frac{\partial v}{\partial w} = -v \left(k_{p\beta} \frac{\partial \phi_1}{\partial w} + k_{d\beta} \frac{\partial \phi_2}{\partial w} \right). \quad (16)$$

The weights are updated as

$$w(t+1) = w(t) + \eta v \left(k_{p\beta} \frac{\partial \phi_1}{\partial w} + k_{d\beta} \frac{\partial \phi_2}{\partial w} \right), \quad (17)$$

where η is the learning rate.

5. Design of One-Wheel Robot

Design of the one-wheel robot becomes the most important issue due to the limitation of space. Our first model is shown in Figure 7. The original idea was to balance and control the sphere robot by differing rotating speeds of two links. Thus, two rotational links inside the sphere are supposed to balance itself but failed due to the irregular momentum induced by two links.

After that, the balancing concept has been changed to use a flywheel instead of links to generate gyro effect as shown in Figure 8. Controlling spin and tilt angles of the flywheel yields gyro effects to make the robot keep upright position.

We have designed and built several models as shown in Figure 8. However, all of models are not able to balance itself long enough. Through trial and error processes of designing models, we have a current version of GYROBO I and II as shown in Figures 9 and 10. Since the GYROBO I in Figure 9 has a limited space for housing, the second model of GYROBO has been redesigned as shown in Figure 11.

6. GYROBO Design

Figure 10 shows the CAD design of the GYROBO II. The real design consists of motors, a flywheel, and necessary hardware as shown in Figure 11.

GYROBO II is redesigned with several criteria. Locating motors appropriately in the limited space becomes quite an important problem to satisfy the mass balance. Thus the size of the body wheel is increased.

The placement of a flywheel effects the location of the center of gravity as well. The size of the flywheel is also critical in the design to generate enough force to make the whole body upright position. The size of the flywheel is increased. The frame of the flywheel system is redesigned and located at the center of the body wheel.

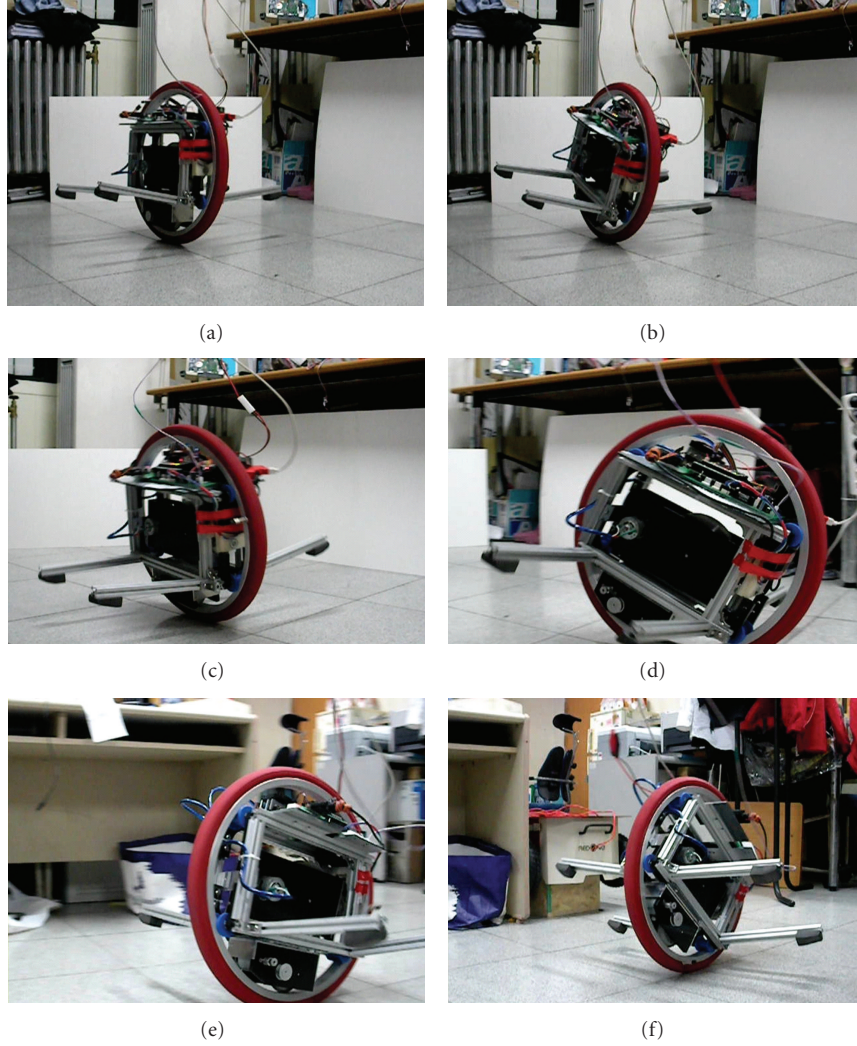


FIGURE 18: PD control: line following task.

The drive motor is attached to the wheel so that it directly controls the movement of the wheel. A tilt motor is mounted on the center line to change the precession angle of the gyro effect. A high-speed spin motor is located to rotate the flywheel through a timing belt.

Control hardware includes a DSP2812 and an AVR as main processors. The AVR controls drive and spin motors while DSP controls a tilt motor. The detailed layout is shown in Figure 12. To detect the lean angle, a gyro sensor is used. Interface between hardware is shown in Figure 13.

7. Experimental Studies

7.1. Balancing Test at One Point

7.1.1. Scheme 1: PD Control. The speed of the flywheel is set to 7,000 rpm. The control frequency is 10 ms. When the PD controller is used, the GYROBO is able to balance itself as shown in Figure 14. However, the heading angle also rotates as it balances. Figures 14(d), 14(e), and 14(f) show the

deviation of the heading angle controlled by a PD controller whose gains are selected as $k_{p\beta} = 10$, and $k_{d\beta} = 15$. PD gains are selected by empirical studies.

7.1.2. Scheme 2: Neural Network Compensation. The same balancing test is conducted with the help of the neural network controller. The learning rate is set to 0.0005, and 3 inputs, 4 hidden units, and 2 outputs are used for the neural network structure. Neural network parameters are found by empirical studies. The GYROBO balances itself and does not rotate much comparing with Figure 14. It stays still as shown in Figure 15. The heading angle is not changed much when neural network control is applied while it deviates for PD control. The larger heading angle error means rotation of GYROBO while balancing which is not desired.

Further clear comparisons of performances between PD control and neural network control are shown in Figures 16 and 17, which are the corresponding plots of Figures 14 and 15. We clearly see that the heading angle of GYROBO is

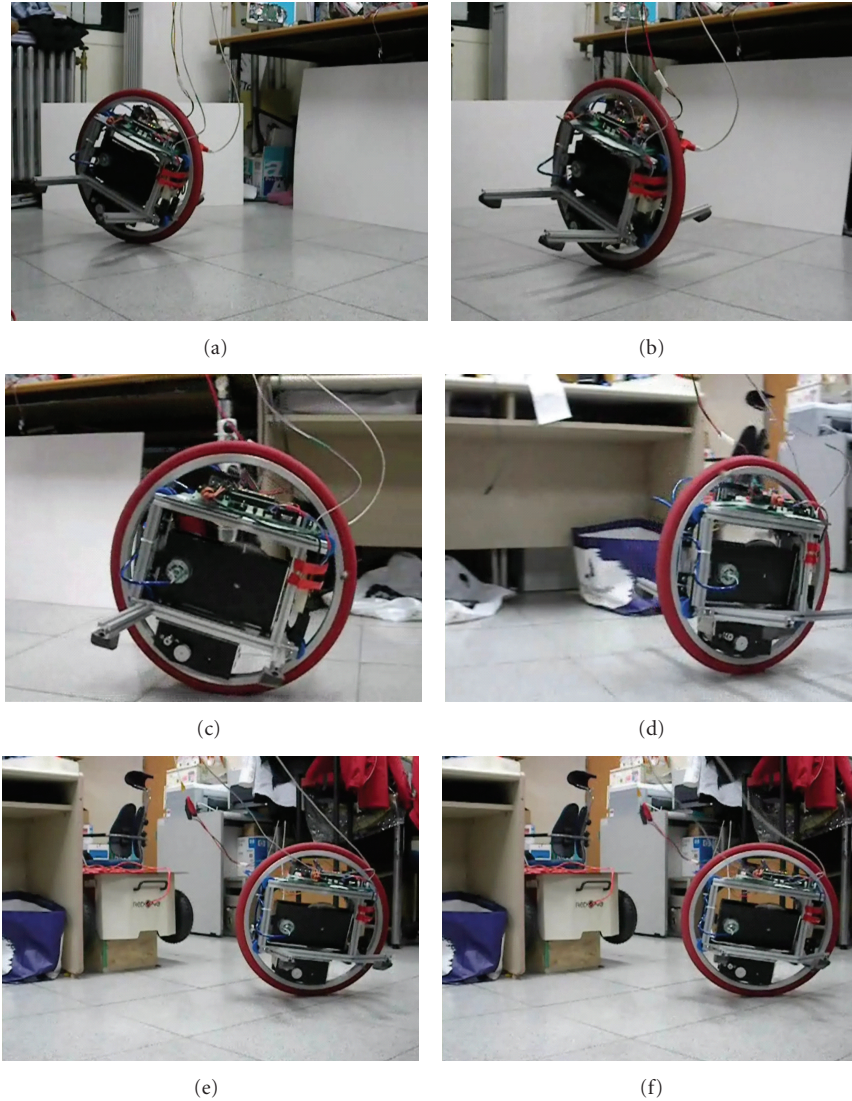


FIGURE 19: Neural network control: line following.

deviating in the PD control case. We clearly see the larger oscillation of PD control than that of RCT control.

7.2. Line Tracking Test

7.2.1. Scheme 1: PD Control. Next test is to follow the desired straight line. Figure 18 shows the line tracking performance by the PD controller. GYROBO moves forward about 2 m and then stops.

7.2.2. Scheme 2: Neural Network Compensation. Figure 19 shows the line tracking performance by the neural network controller.

We see that the GYROBO moves forward while balancing. Since the GYROBO has to carry a power line, navigation is stopped within a short distance about 2 m.

Clear distinctions between PD control and neural network control are shown in Figures 20 and 21. Both controllers maintain balance successfully. GYROBO controlled by

a PD control method deviates from the desired straight line trajectory further as described in Figure 20. In addition, the oscillatory behaviour is reduced much by the neural network controller as shown in Figure 21.

8. Conclusion

A one-wheel robot GYROBO is designed and implemented. After several trial errors of body design, successful design is presented. An important key issue is the design to package all materials in one wheel. One important tip in controlling GYROBO is to reduce the weight so that the flywheel can generate enough gyro effect because it is not easy to find suitable motors. Although balancing and line tracking tasks are successful in this paper, one major problem has to be solved in the future. The power supply for the GYROBO should be independent. Since the stand-alone type of GYROBO is preferred, a battery should be mounted inside

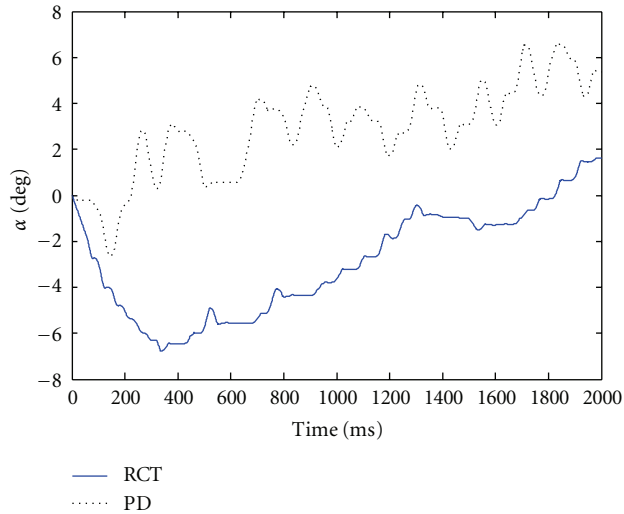


FIGURE 20: Heading angles of PD (black dotted line) and RCT control (blue solid line).

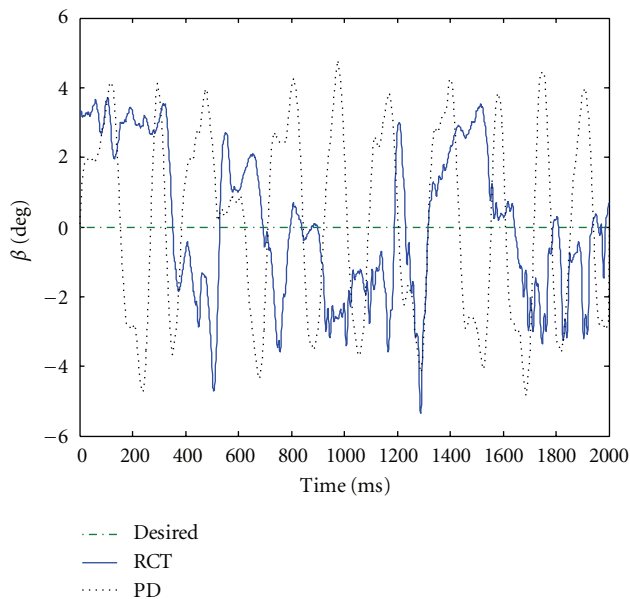


FIGURE 21: Lean angles of PD (black dotted line) and RCT control (blue solid line).

the wheel. Then a room for a battery requires modification of the body design again.

Acknowledgments

This work was financially supported by the basic research program (R01-2008-000-10992-0) of the Ministry of Education, Science and Technology (MEST) and by Center for Autonomous Intelligent Manipulation (AIM) under Human Resources Development Program for Convergence Robot Specialists (Ministry of Knowledge Economy), Republic of Korea.

References

- [1] "Segway," <http://www.segway.com/>.
- [2] S. H. Jeong and T. Takayuki, "Wheeled inverted pendulum type assistant robot: design concept and mobile control," in *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS '07)*, p. 1937, 1932, 2007.
- [3] S. S. Kim and S. Jung, "Control experiment of a wheel-driven mobile inverted pendulum using neural network," *IEEE Transactions on Control Systems Technology*, vol. 16, no. 2, pp. 297–303, 2008.
- [4] J. S. Noh, G. H. Lee, H. J. Choi, and S. Jung, "Robust control of a mobile inverted pendulum robot using a RBF neural network controller," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics, (ROBIO '08)*, pp. 1932–1937, February 2009.
- [5] H. J. Lee and S. Jung, "Development of car like mobile inverted pendulum system : BalBOT VI," *The Korean Robotics Society*, vol. 4, no. 4, pp. 289–297, 2009.
- [6] C. Rui and N. H. McClamroch, "Stabilization and asymptotic path tracking of a rolling disk," in *Proceedings of the 34th IEEE Conference on Decision and Control*, pp. 4294–4299, December 1995.
- [7] G. C. Nandy and Y. Xu, "Dynamic model of a gyroscopic wheel," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 2683–2688, May 1998.
- [8] Y. Xu, K. W. Au, G. C. Nandy, and H. B. Brown, "Analysis of actuation and dynamic balancing for a single wheel robot," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1789–1794, October 1998.
- [9] Y. Xu, H. B. Brown, and K. W. Au, "Dynamic mobility with single-wheel configuration," *International Journal of Robotics Research*, vol. 18, no. 7, pp. 728–738, 1999.
- [10] S. J. Tsai, E. D. Ferreira, and C. J. Raredis, "Control of the gyrover: a single-wheel gyroscopically stabilized robot," in *Proceedings of the IEEE International Workshop on Intelligent Robots and Systems (IROS '99)*, pp. 179–184, 1999.
- [11] Y. Xu and S. K. W. Au, "Stabilization and path following of a single wheel robot," *IEEE/ASME Transactions on Mechatronics*, vol. 9, no. 2, pp. 407–419, 2004.
- [12] Y. S. Xu and Y. S. Ou, *Control of One-Wheel Robots*, Springer, 2005.
- [13] W. Nukulwuthiopas, S. Laowattana, and T. Maneewarn, "Dynamic modeling of a one-wheel robot by using Kane's method," in *Proceedings of the IEEE International Conference on Industrial Technology, (IEEE ICIT '02)*, pp. 524–529, 2002.
- [14] A. Alasty and H. Pendar, "Equations of motion of a single-wheel robot in a rough terrain," in *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 879–884, April 2005.
- [15] Z. Zhu, A. Al Mamun, P. Vadakkepat, and T. H. Lee, "Line tracking of the Gyrobot—a gyroscopically stabilized single-wheeled robot," in *Proceedings of the IEEE International Conference on Robotics and Biomimetics, (ROBIO '06)*, pp. 293–298, December 2006.
- [16] Z. Zhu, M. P. Naing, and A. Al-Mamun, "Integrated ADAMS+MATLAB environment for design of an autonomous single wheel robot," in *Proceedings of the 35th Annual Conference of the IEEE Industrial Electronics Society, (IECON '09)*, pp. 2253–2258, November 2009.
- [17] T. B. Lauwers, G. A. Kantor, and R. L. Hollis, "A dynamically stable single-wheeled mobile robot with inverse mouse-ball

- drive,” in *Proceedings of the IEEE International Conference on Robotics and Automation, (ICRA '06)*, pp. 2884–2889, May 2006.
- [18] U. Nagarajan, A. Mampetta, G. A. Kantor, and R. L. Hollis, “State transition, balancing, station keeping, and yaw control for a dynamically stable single spherical wheel mobile robot,” in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA'10)*, pp. 998–1003, 2009.
- [19] U. Nagarajan, G. Kantor, and R. L. Hollis, “Trajectory planning and control of an underactuated dynamically stable single spherical wheeled mobile robot,” in *Proceedings of the IEEE International Conference on Robotics and Automation, (ICRA '09)*, pp. 3743–3748, May 2009.
- [20] P. K. Kim, J. H. Park, and S. Jung, “Experimental studies of balancing control for a disc-typed mobile robot using a neural controller: GYROBO,” in *Proceedings of the IEEE International Symposium on Intelligent Control, (ISIC '10)*, pp. 1499–1503, September 2010.

