2014 International Conference on Mechatronics and Control (ICMC)
July 3 - 5, 2014, Jinzhou, China

A Design for Two-Wheeled Self-Balancing Robot Based on Kalman Filter and LQR

Fengxin Sun
Automation Research Institute
BOHAI University
Jin Zhou, China 121000–0086
Email: sunfengxin0511@126.com

Zhen Yu
Automation Research Institute
BOHAI University
Jin Zhou, China 121000–0086
Email:459799767@qq.com

Haijiao Yang
Automation Research Institute
BOHAI University
Jin Zhou, China 121000–0086
Email: bhuyhj51@gmail.com

***Abstract*—This paper introduces a method to control a two-wheeled self-balancing robot based on Kalman filter and LQR algorithm.It is incorporated with hardware design,signal processing and control algorithm design.In this system the angle signals from a gyroscope and an accelerometer,disposed by Kalman filter,are combined with LQR controller to accomplish the controlling of the robot.The purpose of simplifying Kalman filter is to make the signal processing using fewer system resources and reduce the burden of the system.A mathematical model was established according to the physical model.Based on the mathematical model,the LQR controller was designed for the attitude and position controlling of the robot.The experiment and simulation based on the prototype robot make it clear that the system could achieve a state of equilibrium and position tracking with a initial disturbance in a short time.**

I. Introduction

The two-wheeled self-balancing robot is a two-wheeled parallel arranged system which could keep balance based on control algorithm combined with the signals measured by angle senors.It is non-linear,non-stable,multivariable and strong coupled so that it has become a classic device used to test the processing capacities of various control algorithms.Meanwhile the two-wheeled self-balancing robot gains a wide applications in engineering practice. The self-balancing electric vehicle has been widely put into production and has a great commercial success [2] [5].

The study of the two-wheeled self-balancing robot has a rapidly development in recent years.Researchers at home and abroad have gained a series of theoretical achievements such as the applications of robust control,neural network control and fuzzy control [6] [7] [8] [9].The experiments and simulations based on above theoretical achievements have a good performance.But the application of the neural network control and fuzzy control are based on a large number of experiments and data.It influences their application in a certain extent,especially for the system on the basis of low-performance Micro Control Unit(MCU).This passage introduces a simplified Kalman filter to simplify the calculation of signal filtering processing.It also builds up the physical motion model and presents a controller based on LQR algorithm to realize the attitude and position controlling of the robot [10].The experiments and simulations verify that the proposed system not only have a good performance in response speed but also a high accuracy.

The second section of the passage introduces the system modules and hardware design of the robot.Section III presents the signal processing by simplifying Kalman filter.The physical motion model of the robot and the controller based on LQR algorithm are described in section IV.The results of physical experiment and simulation of the system are shown at the end of the passage.

II. Hardware design

The robot system is mainly composed of a plastic chassis,a central control board,a sensor board and two coaxial wheels driven by two DC-servo motors. The central control board has a MCU MK60DN512VLQ10 for signals processing and running control algorithm.Two driver chips,VNH2SP30,which are used to driver two DC-servo motors are also placed on the central control board. The sensor board has an accelerometer and a gyroscope for measuring the slant angle and angular speed of the robot [3]. A battery supplies energy for the central control board,the sensor board and two DC-servo motors.There is a encoder beside the left motor used to measure the velocity of the robot.The hardware module design is shown in *Figure 1* and the prototype robot is shown in *Figure 2* .

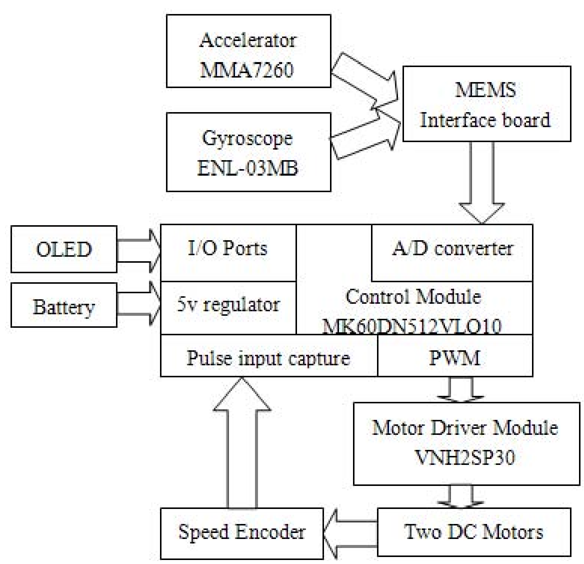

Fig. 1: Hardware module description

Fig. 2: The prototype robot

III. Signal processing

A. Kalman filter

Kalman filter is an optimal recursive data processing algorithm. It could estimate the state of the dynamic system from a series of incomplete measurement which contains noise. The measurement of robot slant angle is accomplished by the accelerometer and the gyroscope.We could got higher accuracy and reliability slant angle of the robot by using Kalman filter to compensate the dynamic error of accelerometer and the drift error of gyroscope.The process of Kalman filtering can be decomposed into time update and measurement update.The time update equations use the previous state and error covariance estimations to estimate the current state.The measurement update equations combine the estimated current state and the measured current state to get the optimal state estimation of the system.The time update equations:

$$X(k|k-1) = AX(k-1|k-1) + BU(k) \tag{1}$$

$$P(k|k-1) = AP(k-1|k-1)A^T + Q \tag{2}$$

The measurement update equations:

$$X(k|k) = X(k|k-1) + K(k)(Z(k) - HX(k|k-1)) \tag{3}$$

$$K(k) = P(k|k-1)H^T(HP(k|k-1)H^T + R))^{-1} \tag{4}$$

$$P(k|k) = (I - K(k)H)P(k|k-1) \tag{5}$$

B. Simplification of Kalman filter

Establish the state equation and the estimate equation of the system based on the derivative relation of slant angle and angular velocity.

$$\begin{bmatrix} \dot{\varphi} \\ \dot{g}_e \end{bmatrix} = \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ g_e \end{bmatrix} + \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_{gyro} \\ \omega_g \end{bmatrix} \tag{6}$$

$$\varphi_{acc} = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ g_e \end{bmatrix} + \omega_a \tag{7}$$

The parameters are defined in *Table I*

TABLE I. PARAMETERS SETTING OF THE ROBOT

Symbols	Parameter	Unit
ω_{gyro}	Pitch angle measured by gyro	rad/s
φ_{acc}	angular velocity measured by acce	rad
ω_g	Measurement noise of gyro	rad/s
φ_a	Measurement noise of acce	rad
g_e	Drift error of gyro	rad/s

Define: $x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \varphi \\ g_e \end{bmatrix}$, $u = \begin{bmatrix} \omega_{gyro} \\ \omega_g \end{bmatrix}$, $A = \begin{bmatrix} 0 & -1 \\ 0 & 0 \end{bmatrix}$, $B = \begin{bmatrix} 1 & 1 \\ 0 & 0 \end{bmatrix}$, $C = \begin{bmatrix} 1 & 0 \end{bmatrix}$.

The equations are written as:

$$\dot{x} = Ax + Bu \tag{8}$$

$$\varphi_{acce} = Cx + \omega_a \tag{9}$$

Sampling period is T.Discretize the above two equations:

$$\begin{aligned} x((k+1)T) &= \phi((k+1)T - kT)x(kT) \\ &\quad + \textstyle\int_{kT}^{(k+1)T} \phi((k+1)T - \tau)Bu(\tau)d\tau \end{aligned} \tag{10}$$

$$\varphi_{acce}(kT) = Cx(kT) + \omega_a(kT) \tag{11}$$

While

$$\phi((k+1)T - kT) = e^{A((k+1)T - kT)} = e^{AT}$$

$$\phi((k+1)T - \tau) = e^{A((k+1)T - \tau)}$$

Define:$\sigma = (k+1)T - \tau$,then

$$x((k+1)T) = e^{AT}x(kT) + (\textstyle\int_0^T e^{A\sigma}d\sigma)Bu(kT) \tag{12}$$

Use k to represent kT.

$$x(k+1) = G(T)x(k) + H(T)u(k) \tag{13}$$

$$\varphi_{acce}(k) = Cx(k) + \omega_a(k) \tag{14}$$

Where

$$G(T) = e^{AT} = \begin{bmatrix} 1 & -T \\ 0 & 1 \end{bmatrix},$$

$$H(T) = B\textstyle\int_0^T e^{A\sigma}d\sigma = \begin{bmatrix} T & T \\ 0 & 0 \end{bmatrix}$$

Substitute(13),(14)into(1)-(5):

$$X(k|k-1) = GX(k-1|k-1) + HU(k) \tag{15}$$

$$P(k|k-1) = GP(k-1|k-1)G^T + Q \tag{16}$$

$$X(k|k) = X(k|k-1) + K(k)(\psi_{acce}(k) - CX(k|k-1)) \tag{17}$$

$$K(k) = P(k|k-1)C^T(CP(k|k-1)C^T + R))^{-1} \tag{18}$$

$$P(k|k) = (I - K(k)C)P(k|k-1) \tag{19}$$

Substitute $x = \begin{bmatrix} \varphi \\ g_e \end{bmatrix}$, $u = \begin{bmatrix} \omega_{gyro} \\ \omega_g \end{bmatrix}$ into(15),(17):

$$\begin{aligned} \begin{bmatrix} \varphi(k|k-1) \\ g_e(k|k-1) \end{bmatrix} &= \begin{bmatrix} 1 & -T \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \varphi(k-1|k-1) \\ g_e(k-1|k-1) \end{bmatrix} \\ &+ \begin{bmatrix} T & T \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \omega_{gyro}(k) \\ \omega_g(k) \end{bmatrix} \end{aligned} \tag{20}$$

$$\begin{aligned} \begin{bmatrix} \varphi(k|k) \\ g_e(k|k) \end{bmatrix} &= \begin{bmatrix} \varphi(k|k-1) \\ g_e(k|k-1) \end{bmatrix} + \begin{bmatrix} K_1(k) \\ K_2(k) \end{bmatrix} \cdot \\ & \{\psi_{acce}(k) - \begin{bmatrix} 1 & 0 \end{bmatrix} \cdot \\ & \begin{bmatrix} \varphi(k|k-1) \\ g_e(k|k-1) \end{bmatrix}\} \end{aligned} \tag{21}$$

Transform the form of the two equations above into:

$$\begin{aligned} \varphi(k|k-1) &= \varphi(k-1|k-1) - Tg_e(k-1|k-1) \\ &+ T\omega_{gyro}(k) + T\omega_g(k) \end{aligned} \tag{22}$$

$$g_e(k|k-1) = g_e(k-1|k-1) \tag{23}$$

$$\varphi(k|k) = \varphi(k|k-1) + K_1[\psi_{acce}(k) - \varphi(k|k-1)] \tag{24}$$

$$g_e(k|k) = g_e(k|k-1) + K_2[\psi_{acce}(k) - \varphi(k|k-1)] \tag{25}$$

$\varphi(k|k)$is the slant angle outputted by Kalman filter.

One thing to be noted is that (16),(18)and(19) are order to figure out K.While K is a convergent variate according to the experimental observation.Its value tends to zero for the robot system.So we could substitute adjusting the value of K_2 for calculating the ten equations.The value of K_2 is designed to zero through the experiment.Then simplify computing of Kalman filter equations.

The simplified Kalman filter equations:

$$\begin{aligned} \varphi(k|k-1) &= \varphi(k-1|k-1) - Tg_e(k-1|k-1) \\ &+ T\omega_{gyro}(k) + T\omega_g(k) \end{aligned} \tag{26}$$

$$\varphi(k|k) = \varphi(k|k-1) + K_1[\psi_{acce}(k) - \varphi(k|k-1)] \tag{27}$$

C. *Parameter adjustment*

Parameters K and T need to be adjusted after simplifying the Kalman filter.K determines the weight of the accelerometer.T is more than a sampling interval.They are proportional relationship between T and the actual angle because the units output by gyroscope and accelerometer are different.

The bigger the value of T,the faster the speed of integrating.Then the tracking effect would be better.But the error caused by gyroscopic drift increased with the value of T growing.It can decrease the drift of actual output by increasing the value of K and the filtering noise increased in the same time because K reflects the weight of the accelerometer.So the evaluation of T and K should avoid to take extreme for the reason that its contradicts each other.

IV. CONTROL ALGORITHM DESIGN

A. *Mathematical model*

The physical motion model of the robot is shown in *Figure 3*

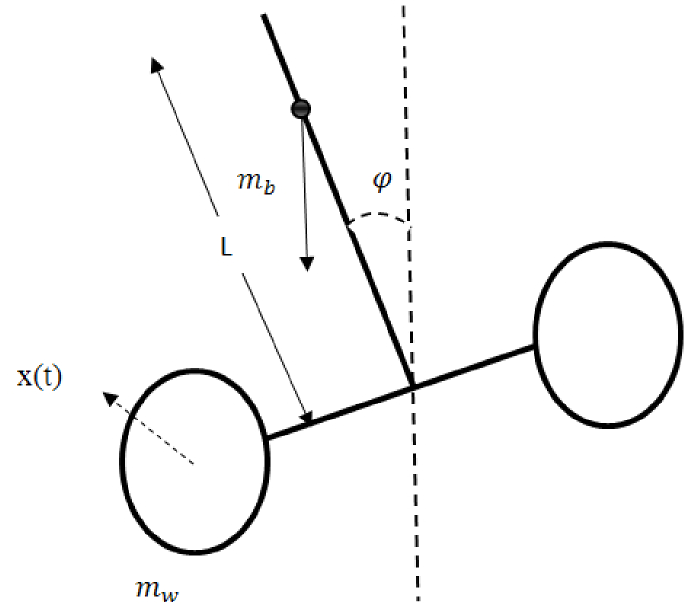

Fig. 3: Physical motion model of the robot

Parameters are defined in *Table II*.

TABLE II.PARAMETERS SETTING

Symbols	Parameter	Value
φ	Pitch angle	rad
$x(t)$	Advance distance	m
m_b	Quality of the chassis	$0.5kg$
m_w	Quality of the wheel	$0.05kg$
L	Height of the robot	$0.28m$

Define I as the moment of inertia of the robot and $u(t)$ is defined as the total power provided by two motors,$I = \frac{1}{3}m_bL^2$.Use Newton's laws of motion to establish the physical motion model [1] [4].

The forces in the horizontal direction:

$$u(t) = (2m_w + m_b)\ddot{x(t)} - m_bL\ddot{\varphi(t)}cos\varphi(t) \tag{28}$$

The forces in the vertical direction:

$$m_bL\ddot{x(t)}cos\varphi(t) = (I + m_bL^2)\ddot{\varphi(t)} - m_bgLsin\varphi(t) \tag{29}$$

When $\varphi(t)$ is very small,then:

$$sin\varphi(t) = \varphi(t)$$

$$cos\varphi(t) = 1$$

Then (28)and(29) are shown as follow:

$$u(t) = (2m_w + m_b)\ddot{x}(t) - m_bL\ddot{\varphi}(t) \tag{30}$$

$$m_bL\ddot{x(t)} = (I + m_bL^2)\ddot{\varphi(t)} - m_bgL\varphi(t) \tag{31}$$

Solve the two equations:

$$\ddot{\varphi}(t) = \frac{(m_w g + m_b) m_b g L}{I(2m_w+m_b)+2m_w m_b L^2}\varphi(t) + \frac{m_b L}{I(2m_w+m_b)+2m_w m_b L^2} u(t)$$

$$\ddot{x}(t) = \frac{m_b^2 g L^2}{I(2m_w+m_b)+2m_w m_b L^2}\varphi(t) + \frac{m_b L^2 + I}{I(2m_w+m_b)+2m_w m_b L^2} u(t)$$

The state space description of the two-wheeled self-balancing robot is written as follow:

$$\begin{bmatrix}\dot{\varphi}(t)\\ \ddot{\varphi}(t)\\ \dot{x}(t)\\ \ddot{x}(t)\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ A_1 & 0 & 0 & 0\\ 0 & 0 & 0 & 1\\ A_2 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\varphi(t)\\ \dot{\varphi}(t)\\ x(t)\\ \dot{x}(t)\end{bmatrix} + \begin{bmatrix}0\\ B_1\\ 0\\ B_2\end{bmatrix} u(t) \quad (32)$$

$$y = \begin{bmatrix}\varphi(t)\\ x(t)\end{bmatrix} = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}\varphi(t)\\ \dot{\varphi}(t)\\ x(t)\\ \dot{x}(t)\end{bmatrix} + \begin{bmatrix}0\\ 0\end{bmatrix} u(t) \quad (33)$$

Where:

$$A_1 = \frac{(m_w g + m_b) m_b g L}{I(2m_w+m_b)+2m_w m_b L^2}$$

$$A_2 = \frac{m_b^2 g L^2}{I(2m_w+m_b)+2m_w m_b L^2}$$

$$B_1 = \frac{m_b L}{I(2m_w+m_b)+2m_w m_b L^2}$$

$$B_2 = \frac{m_b L^2 + I}{I(2m_w+m_b)+2m_w m_b L^2}$$

Substituting the actual parameters:

$$A_1 = 115.5, A_2 = 16.333, B_1 = 11.905, B_2 = 4.447$$

B. LQR

LQR theory has been one of the oldest and most mature state space design method with the development of modern control theory.Especially valuable is that LQR could get the optimal control law of state linear feedback and easy to constitute a closed-loop optimal control.

A given system:

$$\dot{x} = Ax + Bu$$

The quadratic performance index function with respect to x(t) and u(t):

$$J(u(\cdot)) = \frac{1}{2}\int_0^\infty [x^T(t)Qx(t) + u^T(t)Ru(t)]dt$$

Where $Q = Q^T \geq 0$(positive semi-definite) and $R = R^T \geq 0$(positive) are weighting matrix.Exists a unique optimal control u^* making performance index J to obtain the minimal value.

$$u^* = -K^* x^*(t), K^* = R^{-1}B^T P \geq 0$$

Real symmetric matrix P is the only solution matrix of Riccati matrix algebraic equation:

$$PA + A^T P + Q - PBR^{-1}B^T P = 0$$

The optimal closed loop control system is:

$$\dot{x^*} = (A - BR^{-1}B^T P)x^*$$

V. SIMULATION AND EXPERIMENT

Matrix Q and R reflect the weight of state and input.They are usually chosen according to the experience.Choose

$$Q = \begin{bmatrix}1 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}, R = 1$$

and use the MATLAB commend lqr obtain $K = \begin{bmatrix}27.5841 & 2.9604 & -1.0000 & -1.5277\end{bmatrix}$.Set $\varphi_0 = 0.01$.The result of simulation is shown in *Figure 4*.

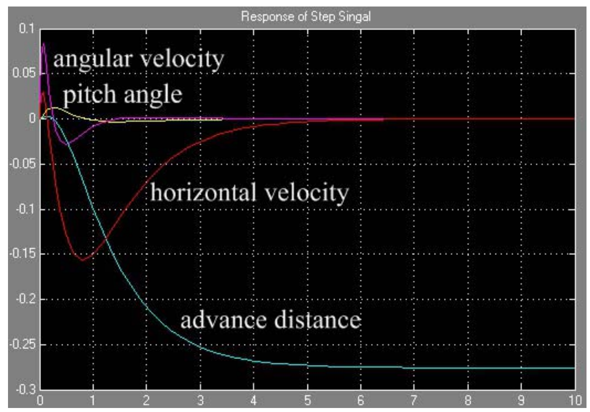

Fig. 4: Attitude control

While Q_{11} and Q_{33} reflect the weight of $\varphi(t)$ and $x(t)$,we try to change its value.Set:

$$Q = \begin{bmatrix}100 & 0 & 0 & 0\\ 0 & 1 & 0 & 0\\ 0 & 0 & 100 & 0\\ 0 & 0 & 0 & 1\end{bmatrix}, R = 1$$

Then $K = \begin{bmatrix}44.6170 & 4.7724 & -10.0000 & -5.2005\end{bmatrix}$.Similarly,set $\varphi_0 = 0.01$.The result of simulation is shown in *Figure 5*.

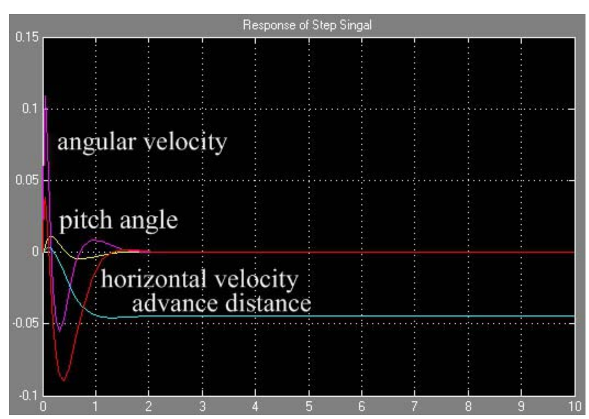

Fig. 5: Attitude control after the adjustment of Q

As is shown in *Figure 5*,regulating time reduced with the values of Q_{11} and Q_{33} increasing.The response time of the system has been improved.

Set $x_0 = 0.5$.Position tracking is shown in *Figure 6*

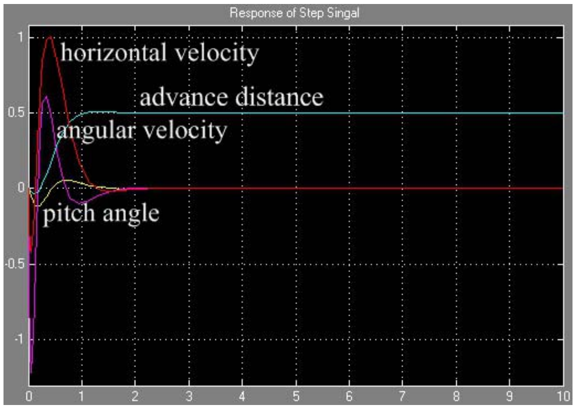

Fig. 6: Position tracking

VI. CONCLUSION

This paper proposed a simplifying method based on Kalman filter and realized the attitude and position controlling of the two-wheeled self-balancing robot combined with LQR controller.The physical experiments and simulations results demonstrate that it makes a good effect in response speed and controlling accuracy.With the development of the theory,a large number of simple and effective methods would spring up and be used in practice.

REFERENCES

[1] M. Baloh and M. Parent, “Modeling and model verification of an intelligent self-balancing two-wheeled vehicle for an autonomous urban transportation system,” in *The conference on computational intelligence, robotics, and autonomous systems*, 2003, pp. 1–7.

[2] C.-N. Huang, “The development of self-balancing controller for one-wheeled vehicles.” *Engineering*, vol. 2, no. 4, 2010.

[3] J. Juan Rincon Pasaye, J. Alberto Bonales Valencia, and F. Jimenez Perez, “Tilt measurement based on an accelerometer, a gyro and a kalman filter to control a self-balancing vehicle,” in *Power, Electronics and Computing (ROPEC), 2013 IEEE International Autumn Meeting on*. IEEE, 2013, pp. 1–5.

[4] B. Mahler and J. Haase, “Mathematical model and control strategy of a two-wheeled self-balancing robot,” in *Industrial Electronics Society, IECON 2013-39th Annual Conference of the IEEE*. IEEE, 2013, pp. 4198–4203.

[5] A. M. Mohtasib and M. H. Shawar, “Self-balancing two-wheel electric vehicle (steve),” in *Mechatronics and its Applications (ISMA), 2013 9th International Symposium on*. IEEE, 2013, pp. 1–8.

[6] C. Sun, T. Lu, and K. Yuan, “Balance control of two-wheeled self-balancing robot based on linear quadratic regulator and neural network,” in *Intelligent Control and Information Processing (ICICIP), 2013 Fourth International Conference on*. IEEE, 2013, pp. 862–867.

[7] L. Sun and J. Gan, “Researching of two-wheeled self-balancing robot base on lqr combined with pid,” in *Intelligent Systems and Applications (ISA), 2010 2nd International Workshop on*. IEEE, 2010, pp. 1–5.

[8] R. Xiaogang, L. Jiang, D. Haijiang, and L. XinYuan, “Design and lq control of a two-wheeled self-balancing robot,” in *Control Conference, 2008. CCC 2008. 27th Chinese*. IEEE, 2008, pp. 275–279.

[9] Q. Yong, L. Yanlong, Z. Xizhe *et al.*, “Balance control of two-wheeled self-balancing mobile robot based on ts fuzzy model,” in *Strategic Technology (IFOST), 2011 6th International Forum on*, vol. 1. IEEE, 2011, pp. 406–409.

[10] N. Yu, Y. Li, X. Ruan, and C. Wang, “Research on attitude estimation of small self-balancing two-wheeled robot,” in *Control Conference (CCC), 2013 32nd Chinese*. IEEE, 2013, pp. 5872–5876.