Optimal State Estimation of Spinning Ping-Pong Ball Using Continuous Motion Model

Yongsheng Zhao, Yifeng Zhang, Rong Xiong, and Jianguo Wang

***Abstract*— Precise trajectory prediction is a fundamental issue for ping-pong robot systems. Due to the difficulty of spin estimation and the complexity of the motion model, most existing algorithms ignore the effect of spin, which will result in a significant deviation in trajectory prediction of spinning ping-pong ball. Some literatures proposed to estimate spin state based on trajectory bias, but due to the limitations of the discrete motion model they derived, only polynomial fitting method can be used for spin estimation, rather than model-based method, which will cause inaccurate spin estimation and trajectory prediction. In this paper, we derive a continuous motion model (CMM) of spinning ping-pong ball based on forces analysis. During the derivation, the Fourier series is used to fit the velocity changing over time, which transforms the model from unsolvable coupled variable-coefficient differential equations to solvable uncoupled equations. On the strength of the CMM, a model-based optimal algorithm for ball's motion state[1] estimation is proposed. Using the initial trajectory acquired by a stereo vision system, the proposed method first estimates the motion state approximately with polynomial fitting, and then uses gradient descent method to achieve a model-based optimal estimation by minimizing a cost function corresponding to the differences between trajectory predictions and observations. We also prove that this optimization problem can be plotted as a convex optimization problem; thus, the globally optimal solution can be obtained. The experimental results confirm the effectiveness and accuracy of the proposed method.**

***Index Terms*— Continuous motion model (CMM), gradient descent method (GDM), initial motion status, optimal estimation, spinning ball, trajectory prediction.**

I. Introduction

PING-PONG robot is an excellent research platform for real-time sensing, intelligent decision making, and servo motion planning. Since Billingsley [1] first proposed a robot table tennis competition in 1983, many research studies have been reported and various ping-pong robot systems that are employing industrial robotic arm [2]–[5], or adopting horizontal mobile track [6]–[11], or using multidegree of freedom dexterous manipulators [12]–[15] with humanoid form have been developed. All the systems mentioned above can play table tennis with some limitations. Some of them, such as the humanoid robots Wu and Kong [14], [15], achieved a high performance of continuous rally either with human players or with each other. However, these systems can only deal with situations where no spin or little spin is involved.

Spin is a common phenomenon in table tennis, and playing spinning ball is an extremely important technique for human players to win the competitions, because spin results in a large bias of trajectory and makes the trajectory difficult to predict accurately. Furthermore, the bouncing of a spinning ball is also more complex and harder to predict. It is an interesting but challenging issue to predict the trajectory of a spinning ping-pong ball. The technique involved can also be extended and applied to the fields of military, sport, or motion control industry. The challenges are mainly from the rigorous requirements on real-time trajectory prediction and high prediction accuracy. The prediction accuracy depends on the precision of the motion model and the estimated state, but the motion model is characterized with high nonlinearity as well as high dimension, and the spin state is hard to be observed precisely in real time.

Due to these difficulties, most existing algorithms of trajectory prediction adopt none of the spinning hypothesis [3], [6], [7], [14]–[18]. Anderson [3] and Matsushima *et al.* [6], [7] took no account of the force analysis and motion dynamics and simply built a mapping between current trajectory observations and future trajectory predictions using polynomial fitting or local linear regression. However, due to the high-order nonlinearity in motion dynamics, these algorithms introduce systematic errors and their prediction accuracy decreases quickly with time. Zhang and Xiong [14], Zhang *et al.* [15]–[17], and Mülling *et al.* built a simplified motion model considering the force effects of gravity, buoyancy, and air resistance. Zhang *et al.* [15]–[17] used the Kalman filter (KF)/the extended KF to estimate the motion state of a ping-pong ball. Zhang and Xiong [14] and Zhang *et al.* [15] proposed a self-adaptive trajectory prediction algorithm that trains corresponding model parameters according to different types of trajectories with artificial neural networks. This algorithm is robust and can derive better trajectory predictions under low spin velocity. Nevertheless, due to inaccurate modeling and observation, the prediction accuracy would decrease as the spin velocity and trajectory bias increase.

Recently, some researchers have made promising progress in the field of spinning ball trajectory prediction. Algorithms using the observation of spin state through

Manuscript received October 9, 2014; revised November 20, 2014; accepted December 3, 2014. Date of publication January 15, 2015; date of current version July 10, 2015. This work was supported in part by the National Natural Science Foundation of China under Grant 61473258 and in part by the Joint Center for Robotics Research between Zhejiang University and the University of Technology, Sydney, NSW, Australia. The Associate Editor coordinating the review process was Dr. Amitava Chatterjee.

Y. Zhao, Y. Zhang, and R. Xiong are with the State Key Laboratory of Industrial Control and Technology, Zhejiang University, Hangzhou 310027, China (e-mail: rxiong@iipc.zju.edu.cn).

J. Wang is with the Center for Autonomous Systems, Faculty of Engineering and Information Technologies, University of Technology, Sydney, NSW 2007, Australia.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIM.2014.2386951

[1] Ball's motion state includes position, flying velocity and spin velocity. For describing convenience, this paper uses the state of the first frame as its state in each trajectory, since once the initial state is given, with the motion model we can derive its state at any given time.

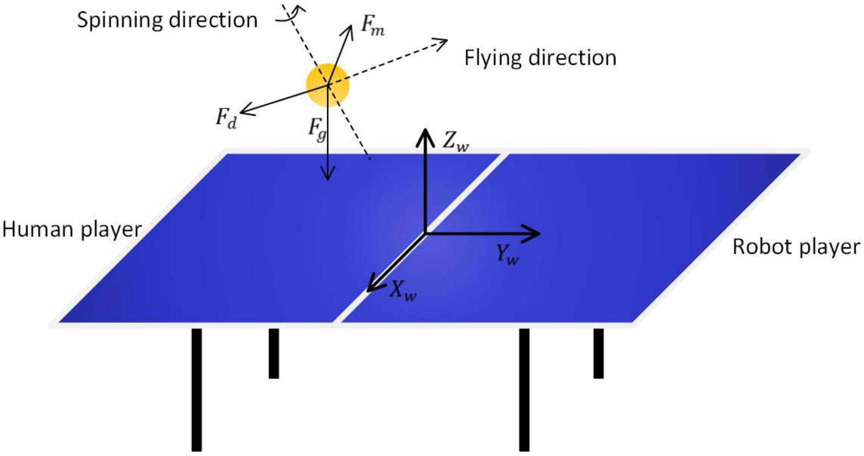


Fig. 1. Global coordinate frame and force analysis of a spinning ping-pong ball.

tracking marks on the ball [19] or using the estimation of spin state through trajectory bias [12], [13], [20]–[24] have been proposed. The former one can provide very precise estimation of spin state and can greatly help in improving trajectory predicting accuracy. However, it needs extra-high-speed pan-tilt tracking vision systems that increase the system cost and complexity. The latter methods estimate the state with polynomial fitting and predict the trajectory by iterating the discrete motion model (DMM), which results in system errors and inaccurate trajectory predictions. Sun *et al.* [23] and Yingshi *et al.* [24] proposed a simplified motion model assuming that the air resistance and the Magnus force are constant during the flight and designed a nonlinear feedback observer to rectify the errors caused by the simplified model. Su *et al.* [22] preprocessed the trajectory observations with a fuzzy filter that decreases the impact of observation errors on the state estimation. However, all of them have not considered the model information during spin-state estimation and cannot achieve optimal results.

Although the DMM is built based on the force analysis and motion dynamics, many problems still exist in the state estimation and trajectory prediction. First, the DMM is not befitting for the optimal state estimation of spinning ping-pong ball, and polynomial fitting will introduce systematic error since the trajectory is of high-order nonlinearity. Second, the spin state is derived from second derivative of the position, and thus, the polynomial fitting error will be magnified. Third, as the DMM assumes that the velocity keeps constant during adjacent frames, the approximation error will accumulate consequently when predicting the trajectory by iterating the DMM.

To conquer the disadvantages of DMM, in this paper, we derive a continuous motion model (CMM) of spinning ping-pong ball, and then propose an optimal state estimation method using gradient descend method (GDM) based on the derived CMM. The major contributions of this paper include the following.

1) A CMM based on force analysis and motion dynamics is derived. To solve the problem of variable coupling with the coefficient matrix in motion equations, we introduce the Fourier series to approximate the velocity changes over time. The CMM makes it possible to integrate the state estimation and the trajectory prediction under a unified motion model, and thus, it can obtain a more accurate trajectory prediction of a spinning ping-pong ball.
2) A model-based optimal state estimation method is proposed using the derived CMM and GDM. We proved that the optimization problem is a convex optimization problem, so the globally optimal state estimation can be searched quickly based on the polynomial fitting results. Compared with the method of polynomial fitting, the state estimated by the proposed method is a model-based optimal estimation, which decreases the impact of observation errors effectively. The experimental results demonstrate the effectiveness and accuracy of the proposed CMM and model-based optimal state estimation method.

The rest of the paper is organized as follows. Section II analyzes the forces applied to a flying spinning ping-pong ball and derives the DMM and CMM, respectively. The optimal state estimation based on the CMM and GDM is proposed in Section III. Then a series of experiments is conducted in Section IV to provide a contrastive analysis of the proposed method in trajectory prediction. Section V summarizes the advantages and disadvantages of the proposed method and gives a prospect on future work.

II. Motion Model of Spinning Ping-Pong Ball

A precise motion model is essential to the accurate trajectory prediction. According to experimental analysis, high spin velocity (up to 360 rad/s) has a great impact on the trajectory bias and must be considered when modeling. The derivation of the CMM contributes to the possibility of designing an optimal state estimation method, which is described in the latter section.

A. Force Analysis

First, a global coordinate frame is defined in Fig. 1. The origin of the global coordinate frame is the center of the table. The axes X_w and Y_w are parallel to the edges of the table,

X_w starts from the origin and points to the right side of the human player, and Y_w points to the robot player.

The forces acting on the ball are also shown in Fig. 1. The air resistance is denoted as F_d, whose magnitude is proportional to the square of the flying velocity and direction is opposite to the flight direction. The gravity is denoted as F_g and the Magnus force is denoted as F_m, which is proportional to the cross product of the flying velocity and spin velocity. In accordance with the research results in [12], [21], and [22], we can conclude

$$\dot{V} = \frac{F}{m} = -\frac{1}{2m} C_D \rho A \, \|V\| \, V + \frac{1}{2m} C_m \rho A r \omega \times V + g \qquad (1)$$

where F is the resultant force, m is the mass of the ping-pong ball, g is the local gravity acceleration, r is the radius and A is the cross sectional area of ping-pong ball, ρ is the air density, C_D is the coefficient of air resistance, C_M is the coefficient of Magnus force, $V = [\, v_x \; v_y \; v_z \,]^T$ is the flying velocity, and $\Omega = [\, \omega_x \; \omega_y \; \omega_z \,]^T$ is the spin velocity in global coordinate frame. The three items on the right side of (1) represent the air resistance, Magnus force and gravity, respectively.

B. Discrete Motion Model

Since the sample period T_s is very short (<8.33 ms), $\dot{V}$ and V can be treated as constant during each sample period. Thus, the DMM can be derived as (2), shown at the bottom of this page, where $k_d = (-1/2m) C_D \rho A$ and $k_m = 1/2m C_m \rho A r$.

C. Continuous Motion Model

The motion model (1) can be described using first-order variable-coefficient differential equations

$$\dot{V}(t) = \begin{bmatrix} -k_d \|V(t)\| & -k_m \omega_z & k_m \omega_y \\ k_m \omega_z & -k_d \|V(t)\| & -k_m \omega_x \\ -k_m \omega_y & k_m \omega_x & -k_d \|V(t)\| \end{bmatrix} V(t) + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}. \qquad (3)$$

The coefficient matrix of (3) contains the independent variable $V(t)$; thus, the motion model is a nonlinear space state model and is not analytically solvable theoretically [25]. That means we cannot obtain the mathematical expression of $V(t)$ by solving (3). Here, we propose a method to deal with this problem, by assuming that the value changing of $\|V(t)\|$ is a function related only to its initial value and time, and there exists a function that can fit the value changing. Consequently, the coefficient matrix can be decoupled from the independent variable $V(t)$. In the Fourier analysis, any function can be represented by the infinite series composed of sine and cosine functions. Therefore, the Fourier series is perfect to fit the velocity changing over time $\|V(t)\|$, which is shown as

$$\|\hat{V}(t)\| = \|V(t)\| = \|V(0)\| \sum_{n=0}^{\infty} a_n \cos(n\omega t) + b_n \sin(n\omega t). \qquad (4)$$

Then the motion model formulated by the coupled variable-coefficient differential equations (3) can be equally transformed to uncoupled equations (5), which can be solved analytically. One thing needs to be declared is that the nonlinearity of the motion model does not change during the derivation, since the coefficient matrix $A(t)$ contains the fitted Fourier series that is time variant

$$\dot{V}(t) = A(t)V(t) + B(t)U(t) \qquad (5)$$

where

$$A\,(t) = \begin{bmatrix} -k_d \|\hat{V}(t)\| & -k_m \omega_z & k_m \omega_y \\ k_m \omega_z & -k_d \|\hat{V}(t)\| & -k_m \omega_x \\ -k_m \omega_y & k_m \omega_x & -k_d \|\hat{V}(t)\| \end{bmatrix}$$

$$B(t) = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}, \quad \text{and } U(t) = 1.$$

The analytic solution of (5) is

$$V(t) = \Phi(t,0)V(0) + \int_0^t \Phi(t,\tau)B(\tau)U(\tau)d\tau \qquad (6)$$

where $\Phi(t,0)$ and $\Phi(t,\tau)$ are the matrix exponential functions and they are equal to the Taylor series of the exponential matrix. Denoting $C(t) = \int A(\tau)d\tau$, we get

$$\Phi(t,0) = e^{\int_0^t A(\tau)d\tau} = \sum_{n=1}^{\infty} \frac{[C(t) - C(0)]^n}{n!}$$
$$\Phi(t,\tau) = e^{\int_\tau^t A(\tau)d\tau} = \sum_{n=1}^{\infty} \frac{[C(t) - C(\tau)]^n}{n!}. \qquad (7)$$

We denote $C(t) - C(0)$ by C_{t-0} and $R(t) - R(0)$ by R_{t-0}, and thus

$$C_{t-0} = \begin{bmatrix} -k_d \|V(0)\| R_{t-0} & -k_m \omega_z t & k_m \omega_y t \\ k_m \omega_z t & -k_d \|V(0)\| R_{t-0} & -k_m \omega_x t \\ -k_m \omega_y t & k_m \omega_x t & -k_d \|V(0)\| R_{t-0} \end{bmatrix} \qquad (8)$$

$$\begin{bmatrix} x(k+1) \\ y(k+1) \\ z(k+1) \\ v_x(k+1) \\ v_y(k+1) \\ v_z(k+1) \end{bmatrix} = \begin{bmatrix} x(k) \\ y(k) \\ z(k) \\ v_x(k) \\ v_y(k) \\ v_z(k) \end{bmatrix} + \begin{bmatrix} v_x(k) \\ v_y(k) \\ v_z(k) \\ -k_d \|V(k)\| v_x(k) + k_m(\omega_y v_z(k) - \omega_z v_y(k)) \\ -k_d \|V(k)\| v_y(k) + k_m(\omega_z v_x(k) - \omega_x v_z(k)) \\ -k_d \|V(k)\| v_z(k) + k_m(\omega_x v_y(k) - \omega_y v_x(k)) - g \end{bmatrix} T_S \qquad (2)$$

where

$$R(t) = \sum_{n=0}^{\infty} \left\{ \frac{a_n}{n\omega} \sin(n\omega t) - \frac{b_n}{n\omega} \cos(n\omega t) \right\}.$$

As a tradeoff between the calculation efficiency and model accuracy, the infinite series in (4) and (7) are counted up to a maximum of second order.

For further convenience, we denote

$$\begin{aligned} J_1(t) = &-\frac{1}{6}a_0 t^3 - \frac{4a_1+a_2}{4\omega^2}t \\ &+\sum_{n=1}^{2} \frac{1}{(n\omega)^3}[a_n \sin(n\omega t) - b_n \cos(n\omega t) + b_n] \end{aligned} \quad (9)$$

$$J_2(t) = -\frac{1}{2}a_0 t^2 + \sum_{n=1}^{2} \frac{1}{(n\omega)^2}[a_n \sin(n\omega t) - b_n \cos(n\omega t) - a_n] \quad (10)$$

$$\begin{aligned} J_3(t) = &-\sum_{n=1}^{2} \frac{2a_0 t}{(n\omega)^2}[a_n \cos(n\omega t) + b_n \sin(n\omega t)] \\ &+\sum_{n=1}^{2} \frac{b_n^2 - a_n^2}{2(n\omega)^3} \sin(n\omega t) \cos(n\omega t) \\ &+\sum_{n=1}^{2} \frac{2a_0}{(n\omega)^3}\{a_n \sin(n\omega t) - b_n[\cos(n\omega t) - 1]\} \\ &+\sum_{n=1}^{2} \left[\frac{a_n^2 + b_n^2}{2(n\omega)^2}t - \frac{a_n b_n}{(n\omega)^3} \sin(n\omega t)^2 \right] \quad (11) \\ &+\sum_{n=1,3} \frac{b_1}{2n\omega^3}[a_2 \cos(n\omega t) + b_2 \sin(n\omega t)] \\ &-\sum_{n=1,3} \frac{(n-2)a_1}{2n\omega^3}[a_2 \sin(n\omega t) - b_2 \cos(n\omega t)] \\ &+\frac{a_1 b_2 - 2a_2 b_1}{3\omega^3} + \frac{1}{3}a_0^2 t^3 \end{aligned}$$

$$\begin{aligned} d_1(t) = &\frac{1}{2}k_m \omega_y t^2 \\ &-k_d k_m \omega_y \|V(0)\| \left[\frac{1}{2}R(t)t^2 + J_1(t) \right] + \frac{1}{6}k_m^2 \omega_x \omega_z t^3 \end{aligned} \quad (12)$$

$$\begin{aligned} d_2(t) = &-\frac{1}{2}k_m \omega_x t^2 \\ &+k_d k_m \omega_x \|V(0)\| \left[\frac{1}{2}R(t)t^2 + J_1(t) \right] + \frac{1}{6}k_m^2 \omega_y \omega_z t^3 \end{aligned} \quad (13)$$

$$\begin{aligned} d_3(t) = &\, t - \frac{1}{6}k_m^2 (\omega_x^2 + \omega_y^2) t^3 - k_d \|V(0)\| [R(t)t + J_2(t)] \\ &+\frac{1}{2}k_d^2 \|V(0)\|^2 \big[R(t)^2 t + 2R(t)J_2(t) + J_3(t) \big]. \end{aligned} \quad (14)$$

Thus, the analytic solution of (5) can be formulated as

$$V(t) = s(t)V(0) + d(t) \quad (15)$$

where

$$s(t) = \left[I + C_{t-0} + \frac{C_{t-0}^2}{2} \right]$$

$$d(t) = [\, d_1(t) \quad d_2(t) \quad d_3(t) \,]^T.$$

In addition, the position change $P(t) - P(0)$ is equal to the integration of $V(t)$ over time

$$P(t) = S(t)V(0) + D(t) + P(0) \quad (16)$$

where

$$S(t) = \int_0^t s(\tau)d\tau, \quad D(t) = \int_0^t d(\tau)d\tau.$$

Equations (15) and (16) constitute the CMM.

III. Model Optimal State Estimation

State estimation is the crucial procedure for trajectory prediction. In addition, it can be formulated as an optimization problem using CMM and be solved with GDM. We also mathematically proved that the optimization problem is a convex optimization problem, which indicates that the obtained state estimation is globally optimal.

A. Spin Velocity Calculation

The relationship between flying velocity and spin velocity can be derived from DMM (2) as

$$\begin{bmatrix} v_x(k+1) + k_v v_x(k) \\ v_y(k+1) + k_v v_y(k) \\ v_z(k+1) + k_v v_z(k) + gT_s \end{bmatrix} = \begin{bmatrix} 0 & k_m v_z(k)T_s & -k_m v_y(k)T_s \\ -k_m v_z(k)T_s & 0 & k_m v_x(k)T_s \\ k_m v_y(k)T_s & -k_m v_x(k)T & 0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (17)$$

where $k_v = (k_d\|V(k)\|T_s - 1)$.

According to the definition of the Magnus force, the spin velocity is perpendicular to the Magnus force, which is shown as follows:

$$\begin{bmatrix} v_x(k+1) + k_v v_x(k) \\ v_y(k+1) + k_v v_y(k) \\ v_z(k+1) + k_v v_z(k) + gT_s \end{bmatrix} \times \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = 0. \quad (18)$$

Given two adjacent flying velocity $V(k)$ and $V(k+1)$, the spin velocity can be calculated by (17) and (18). Consequently, estimating spin velocity is equivalent to estimating the flying velocities of adjacent frames, which can be optimally estimated with the method covered in the next section.

B. Mathematic Description of Optimization Problem

The optimization objective is to find the optimal state by minimizing the cost function under the constraints of the motion model, while the observation error exists in the observed position sequence. The cost function is the quantization of the curvature tolerance between the predicted and observed trajectories. As shown in (19), the optimization goal is minimizing the cost function and the optimization target is the motion state $[P(0), V(0), \omega]$

$$\begin{cases} \min\limits_{V(0)\in\Phi_v} \left\{ F_{\text{eva}}(n) = \frac{1}{2} \sum\limits_{k=T_s}^{nT_s} [\hat{P}(k) - P(k)]^T [\hat{P}(k) - P(k)] \right\} \\ \text{s.t. } P(k) = S(k)V(0) + D(k) + P(0) \end{cases} \quad (19)$$

where $\hat{P}(k)$ and $P(k)$ are the trajectory observation and prediction at the kth frame and Φ_v is the state space of flying velocity.

C. Optimal State Estimation

As discussed above, the position and the spin velocity can be formulated by the functions of flying velocity, and thus, the key point of the state estimation is getting the accurate estimation of flying velocity $V(0)$ and $V(1)$ (Since $V(0)$ and $V(1)$ can be estimated with the same method, for convenience, we will only describe how to get $V(0)$ with the optimal estimation method). The high nonlinearity and high-dimensional characters of the CMM make it difficult to solve this optimization problem analytically with least mean square (LMS) method. However, the gradient of the cost function $F_{\text{eva}}(n)$ with respect to the flying velocity $V(0)$ can be easily derived taking the advantages of CMM. Thus, GDM is used to search the optimal flying velocity over the whole state space.

The gradient of the cost function $F_{\text{eva}}(n)$ with respect to initial flying velocity $\hat{V}(0)$ can be derived from (19) as

$$\nabla_{\hat{V}(0)} F_{\text{eva}}(n) = \sum_{k=T_s}^{nT_s} \left\{ S(k)^T + \frac{\partial S(k)^T}{\partial \hat{V}(0)} \hat{V}(0) + \frac{\partial D(k)^T}{\partial \hat{V}(0)} \right\} [\hat{P}(k) - P(k)]. \tag{20}$$

When the position sequence of a spinning ball is acquired by the high-speed vision system, the motion state of a spinning ball can be obtained approximately using polynomial fitting method, as in [20]. Starting at the approximate flying velocity $\hat{V}(0)$, the optimal flying velocity can be searched with GDM, which is shown as

$$\hat{V}(0)_{k+1} = \hat{V}(0)_k - \lambda \nabla_{\hat{V}(0)} F_{\text{eva}}(n) \tag{21}$$

where λ is the searching step.

For general optimization problems, the solution provided by GDM is locally optimal. In addition, the solution is globally optimal if and only if the optimization problem is a convex optimization. Fortunately, the optimization problem here can be proved to be a convex optimization. In particular, the cost function $F_{\text{eva}}(n)$ is a convex function and the state space Φ_v is a convex set.

We prove the following.

1) The state space Φ_v is a real vector set, so Φ_v is a convex set.
2) According to (15) and (16), $F_{\text{eva}}(n)$ is continuous and differentiable in the state space Φ_v and the second-order partial derivative exists. The Hessian matrix $H(n)$ of $F_{\text{eva}}(n)$ about $\hat{V}(0)$ proves to be positive definite. Thus, $F_{\text{eva}}(n)$ is a convex function, which is shown as

$$H(n) = \sum_{k=T_s}^{nT_s} \left\{ [\hat{P}(k) - P(k)]^T \frac{\partial W(k)}{\partial \hat{V}(0)} + W(k)W(k)^T \right\} > 0 \tag{22}$$

where $W(k) = S(k)^T + (\partial S(k)^T / \partial \hat{V}(0))\hat{V}(0) + (\partial D(k)^T / \partial \hat{V}(0))$.

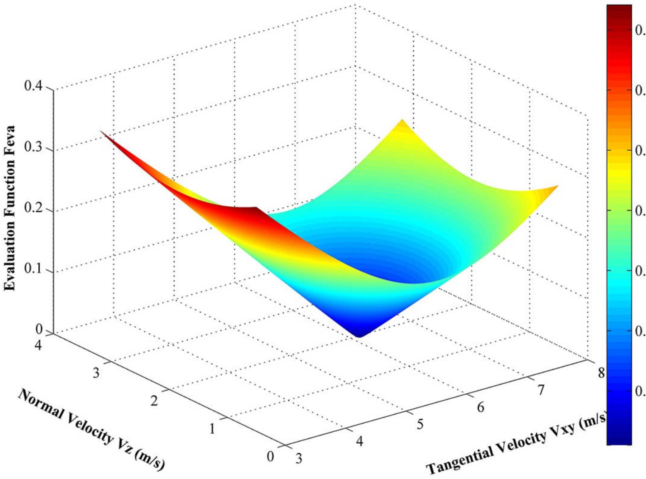


Fig. 2. 3-D graph of the cost function over the whole state space.

Algorithm 1 Model Based Optimal State Estimation

```
The approximate state estimating process
    Fit the observed position sequence with
        two order polynomial proposed in [20]
    Obtain the approximate flying velocity $\hat{V}(0)$ with
        the derivation of the fitted two order polynomial
    Obtain the approximate spin velocity $\hat{\omega}$ with (17) (18)
The optimal state estimating process
    Predict the trajectory with the approximate state
        estimation and CMM
    Compute the cost function $F_{eva}(n)$ with (19)
    While $F_{eva}(n) > threshold$
        Compute the gradient $\nabla_{\hat{V}(0)} F_{eva}(n)$ with (20)
        Search the flying velocity along the gradient descent
            direction $\hat{V}(0)_{k+1} = \hat{V}(0)_k - \lambda \nabla_{\hat{V}(0)} F_{eva}(n)$
        Update spin velocity using the searched flying
            velocity
        Update trajectory prediction with the searched motion
            state
        Update the cost function $F_{eva}(n)$
    end
    Obtain the optimal state estimation
```

The simulation result also supports this fact, as shown in Fig. 2. The initial flying velocity is $V(0) = [0.2\ 5.8\ 1.9]^T$, which is a common case. The state space ranges from $[0.1\ 3.0\ 0.5]^T$ to $[0.3\ 8.0\ 3.2]^T$. The red indicates that cost function is large and the blue is small. As is shown in the figure, there is only one global minimum point in the state space, which is the initial flying velocity $V(0)$. The cost function $F_{\text{eva}}(n)$ is a convex function and it will converge on the global minimum value.

Algorithm 1 gives the complete approach of the model-based optimal state estimation using the CMM and GDM.

After getting the optimal state estimation of a spinning ping-pong ball, the following trajectory can be conveniently calculated using the CMM.

TABLE I

VALUES OF THE CONSTANT PARAMETERS

Symbol	Quantity	Value
m	mass	$0.0027kg$
g	acceleration of gravity	$9.827\ m/s^2$
r	radius	$0.02\ m$
ρ	air density	$1.29kg/m^3$
A	cross sectional area	$0.001256m^2$
C_D	coefficient of air resistance	0.456
C_m	coefficient of Magnus	1.0

TABLE II

FITTED VALUES OF THE TWO-ORDER FOURIER SERIES

Symbol	Value	95% confidence intervals
a_0	2415270.15	-1.101E+10 ~ 1.1017E+10
a_1	-3219498.7	-1.469E+10 ~ 1.4682E+10
a_2	804229.585	-3.669E+09 ~ 3670526206
b_0	0	0
b_1	72174.7826	-246798145 ~ 246942495
b_2	-36075.311	-123457460 ~ 123385309
ω	-0.091638	-104.58344 ~ 104.400163

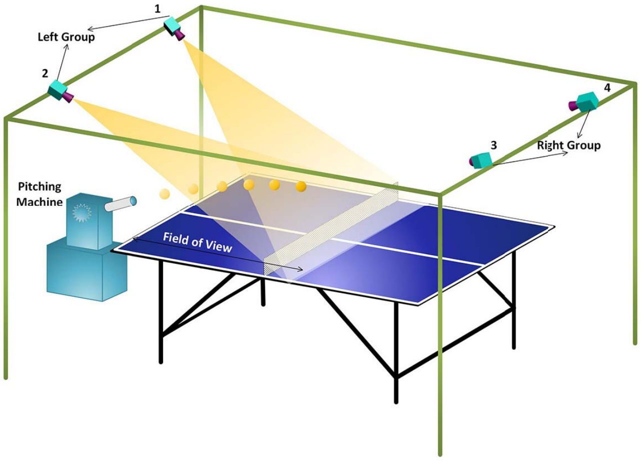


Fig. 3. Structure of the experiment platform.

IV. Experiments and Results

With the purpose to analyze the performance of the state estimation and trajectory prediction method proposed in this paper, we constructed several experiments on a platform shown in Fig. 3. The platform can collect the trajectory data of the spinning ping-pong ball. A series of contrast experiments is designed between the proposed method and the state-of-the-art method [21] to verify the performances from different aspects, including the curvature tolerance error, the collision point error, and the time efficiency.

A. Experiment Platform

The experiment platform is composed of a standard-sized table, a V-989E table tennis serving machine, and a high-speed vision system. The standard-sized table is 2.74 m in length, 1.525 in width, and 0.76 m in height. The height of the cameras to the ground is about 2.25 m, the distance between the cameras 1 and 2 is about 1.2 m. The distance between the cameras 2 and 3 is about 4.6 m. The serving machine can volley a ping-pong ball with a speed up to 10 m/s and spinning up to 360 rad/s. The direction and magnitude of the flying velocity and spin velocity can be changed by adjusting the speed and the angle of two motors inside the serving machine. The high-speed vision system work under 120 frames/s is composed of four cameras and two computers. The four cameras are divided into two groups, each group has two cameras and is in charge of each half court. The trajectory data observed by each group can be merged into a whole trajectory with position error up to 0.005 m.

B. Model Parameters

1) Constant Parameters in Motion Model: According to the forces and motion dynamic analysis, the parameters m, g, r, ρ, C_D, A, and C_m in the motion model can be treated as constants. The values of parameters are estimated by fitting the collected trajectory data with LMS, which is shown in Table I.

2) Variable Parameters in Fourier Series: A two-order Fourier series is used to fit the decay curve of flying velocity as a tradeoff between fitting accuracy and computational consumption. Table II shows the values of the parameters in Fourier series. As shown in Fig. 4, the flying velocity first decreases because of the existence of air resistance. As the effect of gravity exceeds the effect of air resistance over time, the flying velocity increases. The residual is defined as the Euclid distance between the observed data and the fitted data, which is shown in Fig. 5. The residuals of the fitted data set are small and are equally distributed, which demonstrates a perfect two-order Fourier series fitting.

C. Error Analysis of Trajectory Prediction

Since the motion model and the state estimation both serve for the trajectory prediction, the performance of trajectory prediction is a good criterion to evaluate the effectiveness

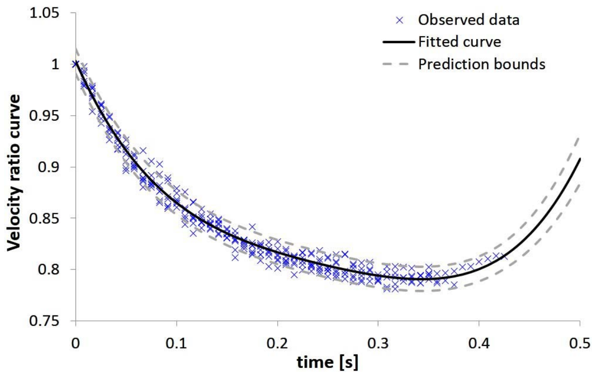


Fig. 4. Fourier series fit to the value changing of flying velocity. The blue points are the observed data of decay curve, the black curve is the fitted result of the two-order Fourier series and the green one is the prediction bounds with 95% confidence. The flying state of the fitted trajectories varies from 4.5 to 7.5 m/s, and the spin state varies from 40 to 70 rad/s.

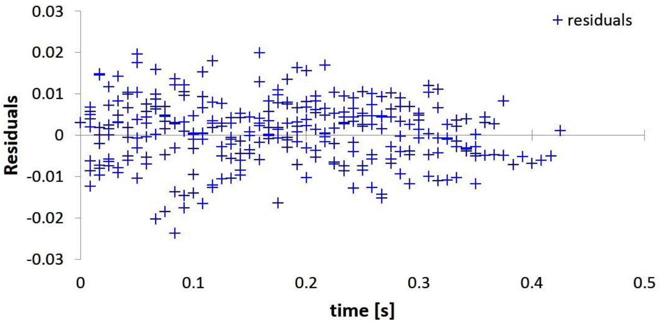


Fig. 5. Residuals of the fitted data.

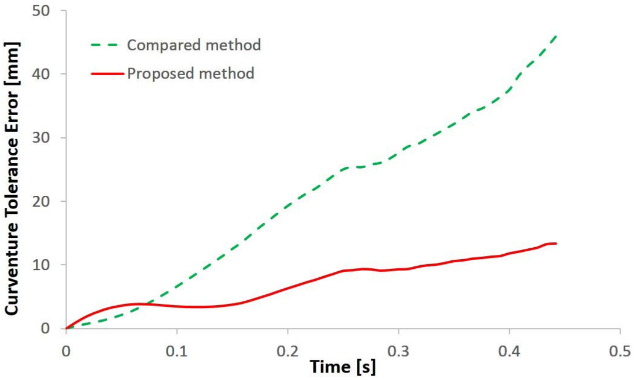


Fig. 6. Curvature tolerance of the two algorithms.

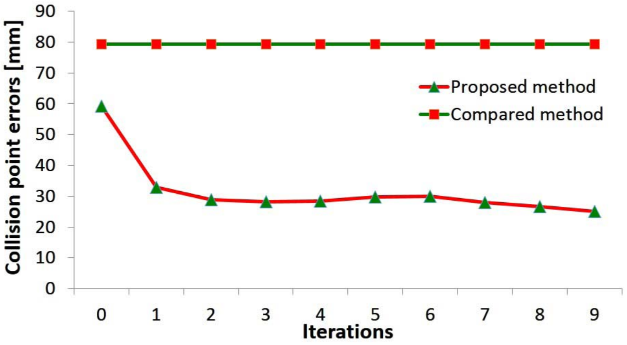


Fig. 7. Trend of collision point prediction errors.

and accuracy of the motion model and the state estimation. In our experiments, when the ball crosses the plain $y = -0.6$ m, the trajectory prediction algorithm starts. The data used in this paper contains 60 spinning ball trajectories.

1) Curvature Tolerance Analysis: The curvature tolerance is defined as the average Euclidean distance between the predicted trajectory and the observed trajectory, as shown in

$$\mathrm{Ct}(n) = \frac{1}{n} \sum_{k=T_s}^{nT_s} \| \hat{P}(k) - P(k) \| \tag{23}$$

where n is the nth frame of the observed trajectory data.

The curvature tolerance is the overall metric of the proposed trajectory prediction method and it is affected by the both performance of the CMM and the state estimation. The smaller the curvature tolerance is, the more accurate the trajectory prediction will be.

Fig. 6 shows the comparison of curvature tolerance between the two methods. The curvature tolerance of the proposed method is much smaller than the compared method, which indicates the advantages of the proposed motion model and state estimation method.

2) Collision Point Prediction Accuracy Analysis: The flying trajectory is separated into two parts at the collision point, and consequently, the collision point is the end of the trajectory before collision and the beginning of the trajectory after the collision. Therefore, the collision point accuracy is a remarkable metric for the performance of the trajectory prediction method. The value of z axis at the collision point is constant and equal to the radius of ping-pong ball, so the measurement of the collision point accuracy can be processed on the x–y plane.

Fig. 7 shows the trend of the collision point error as the iteration of state estimation increases. The collision point error of the compared method is constant since there is no iteration process during the process of the motion state estimation. At the beginning, the collision point prediction errors of the two methods are both large, since they both estimate the motion state using polynomial fitting methods at the first step. However, the error of the proposed method decreases as the iteration increases, especially at the first iteration. This demonstrates that the proposed optimal state estimation method improves the state estimation accuracy effectively and rapidly.

The distribution of the collision point prediction error of the two methods is shown in Fig. 8. The proposed method has a better performance than the compared method in both x- and y-axis with a more equal and smaller distribution in value. Specifically, the compared method shows a noticeable collision point prediction error along y-axis, and some even exceed 0.1 m and cannot meet the needs of a playing servo ping-pong robot.

Table III shows the values of the mean collision point prediction error and the mean squared collision point prediction error for 60 trajectories with different speeds and spin velocities. The mean error and mean squared error of the

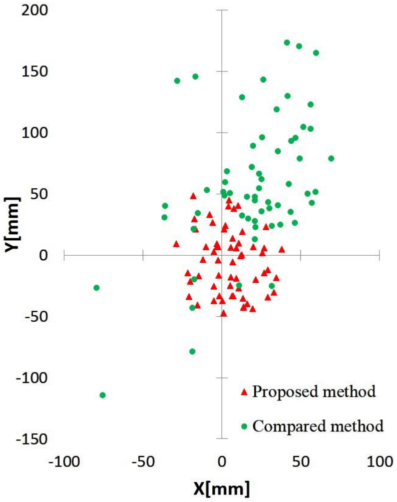
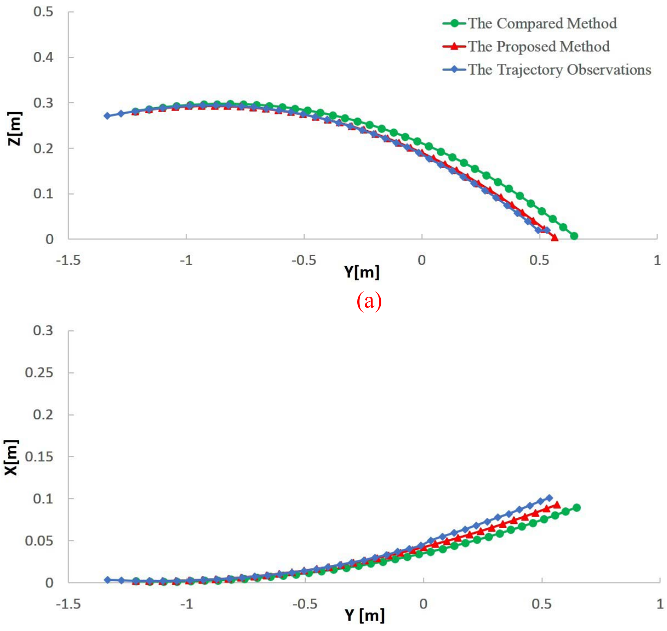


Fig. 8. Distribution of the collision point prediction errors.

TABLE III
STATISTICAL ANALYSIS OF COLLISION POINT ERROR

	Mean Error		Mean Squared Error	
	X(m)	Y(m)	X(m)	Y(m)
The Proposed Method	0.0137	0.0291	0.0174	0.0347
The Compared Method	0.0330	0.0685	0.0392	0.0811



(a)

(b)

Fig. 9. Trajectory prediction demo. (a) z–y plane. (b) x–y plane. The collision point prediction error of the proposed method is (8.2, 30.1) mm and the compared method is (10.1, 121.5) mm. The curvature tolerance of the proposed method is 10.7 mm and the compared method is 30.3 mm.

proposed method are much smaller than the compared method in both x- and y-axis, which indicates that the proposed method is more robust and more accurate.

A trajectory prediction demo is shown in Fig. 9(a) and (b). The observed trajectory is plotted in blue, the trajectory predicted by the proposed method is in red, and the trajectory predicted by the compared method in green. The ball is with a speed of 7.06 m/s and spinning at 70 rad/s.

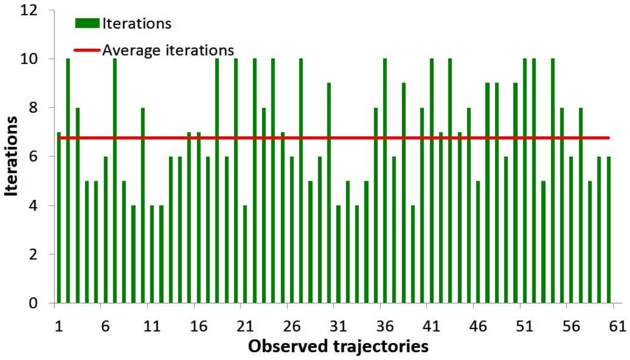


Fig. 10. Iterations of the proposed initial motion status estimation method.

3) Computation Consumption Analysis: Fig. 10 shows the iterations processed in the proposed state method. The iterations vary from four to 10 times according to different types of trajectories, and the average iteration times is 6.75. The hardware platform is an 8-G memory and i7, 2.0-G CPU ×86 pc running a 64-bit operation system. The average time consumption of the compared method is 2.29 ms and the proposed method 4.67 ms. Because of the utilization of GDM, the time consumption of the proposed method is larger than the compared method, while it can still be accomplished within one iterative period (8.33 ms for a frame rate of 120 frames/s). The proposed method improves the estimation accuracy of trajectory prediction with a reasonable increase in time consumption.

V. Conclusion

In this paper, we built a DMM for spinning ping-pong ball based on the force analysis and derived the CMM using the Fourier series to fit the value changing of the flying velocity over time. Based on the CMM, an optimal state estimation method was proposed, which can get the model-based optimal state estimation using a short piece of position observation sequence with GDM. The CMM and model-based optimal state estimation method constitute the trajectory prediction algorithm for spinning ping-pong ball. The experimental results demonstrated the effectiveness and accuracy of the proposed trajectory prediction method.

As discussed above, trajectory bias is used to estimate the spin velocity. The larger the included angle between the flying velocity and spin velocity is, the bigger the impact of spin velocity on trajectory bias is, and the easier the spin velocity is to estimate. It is impossible to estimate the spin velocity using the proposed method when the flying velocity and spin velocity are parallel. However, it is rare in a real ping-pong sport scenario, therefore being neglected. The value changing of flying velocity is fitted by the Fourier series with the assumption that it is a function only related to its initial value and time. In fact, it is also affected by the initial spin state. A better way to deal with this situation is classifying the trajectories with initial motion state and then using the Fourier series to fit them correspondingly.

In the future, new filtering algorithm and state estimation method will be investigated to decrease the impact of observation error on state estimation and improve the accuracy of the trajectory prediction. The bouncing of a spinning ball on the table and racket will also be investigated.

References

[1] J. Billingsley, "Robot ping pong," *Pract. Comput.*, vol. 6, no. 5, 1983.

[2] R. L. Andersson, *A Robot Ping-Pong Player: Experiments in Real-Time Intelligent Control*. Cambridge, MA, USA: MIT Press, 1988.

[3] R. L. Andersson, "Aggressive trajectory generator for a robot ping-pong player," *IEEE Control Syst. Mag.*, vol. 9, no. 2, pp. 15–21, Feb. 1989.

[4] R. L. Andersson, "Understanding and applying a robot ping-pong player's expert controller," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 1989, pp. 1284–1289.

[5] K. P. Modi, F. Sahin, and E. Saber, "An application of human robot interaction: Development of a ping-pong playing robotic arm," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2005, pp. 1831–1836.

[6] M. Matsushima, T. Hashimoto, and F. Miyazaki, "Learning to the robot table tennis task-ball control & rally with a human," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2003, pp. 2962–2969.

[7] M. Matsushima, T. Hashimoto, M. Takeuchi, and F. Miyazaki, "A learning approach to robotic table tennis," *IEEE Trans. Robot.*, vol. 21, no. 4, pp. 767–771, Aug. 2005.

[8] F. Miyazaki, M. Matsushima, and M. Takeuchi, "Learning to dynamically manipulate: A table tennis robot controls a ball and rallies with a human being," in *Advances in Robot Control*. New York, NY, USA: Springer, 2006, pp. 317–341.

[9] F. Miyazaki, M. Takeuchi, M. Matsushima, T. Kusano, and T. Hashimoto, "Realization of the table tennis task based on virtual targets," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2002, pp. 3844–3849.

[10] M. Takeuchi, F. Miyazaki, M. Matsushima, M. Kawatani, and T. Hashimoto, "Dynamic dexterity for the performance of 'wall-bouncing' tasks," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2002, pp. 1559–1564.

[11] L. Acosta, J. J. Rodrigo, J. A. Mendez, G. N. Marichal, and M. Sigut, "Ping-pong player prototype," *IEEE Robot. Autom. Mag.*, vol. 10, no. 4, pp. 44–52, Dec. 2003.

[12] X. Chen, Y. Tian, Q. Huang, W. Zhang, and Z. Yu, "Dynamic model based ball trajectory prediction for a robot ping-pong player," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2010, pp. 603–608.

[13] X. Chen, Q. Huang, W. Zhang, Z. Yu, R. Li, and P. Lv, "Ping-pong trajectory perception and prediction by a PC based high speed four-camera vision system," in *Proc. 9th World Congr. Intell. Control Autom (WCICA)*, 2011, pp. 1087–1092.

[14] Y. Zhang and R. Xiong, "Real-time vision system for a ping-pong robot," *Sci. Sinica Inf.*, vol. 42, no. 9, pp. 1115–1129, 2012.

[15] Y. Zhang, R. Xiong, Y. Zhao, and J. Chu, "An adaptive trajectory prediction method for ping-pong robots," in *Proc. Int. Conf. Intell. Robot. Appl. (ICIRA)*, Oct. 2012, pp. 448–459.

[16] Y.-H. Zhang, W. Wei, and D. Yu, "Precisely tracking trajectory of ping-pong robot based on air drag factor estimation," *Opto-Electron. Eng.*, vol. 36, no. 6, pp. 15–20, 2009.

[17] Y.-H. Zhang, W. Wei, D. Yu, and C.-W. Zhong, "A tracking and predicting scheme for ping pong robot," *J. Zhejiang Univ. Sci. C*, vol. 12, no. 2, pp. 110–115, 2011.

[18] K. Mülling, J. Kober, and J. Peters, "A biomimetic approach to robot table tennis," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2010, pp. 1921–1926.

[19] Y. Zhang, Y. Zhao, R. Xiong, Y. Wang, J. Wang, and J. Chu, "Spin observation and trajectory prediction of a ping-pong ball," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May/Jun. 2014, pp. 4108–4114.

[20] Z. Zhang, D. Xu, and M. Tan, "Visual measurement and prediction of ball trajectory for table tennis robot," *IEEE Trans. Instrum. Meas.*, vol. 59, no. 12, pp. 3195–3205, Dec. 2010.

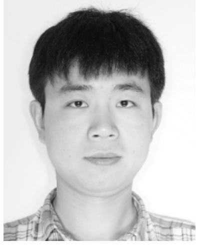
[21] Y. Huang, D. Xu, M. Tan, and H. Su, "Trajectory prediction of spinning ball for ping-pong player robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Sep. 2011, pp. 3434–3439.

[22] H. Su, Z. Fang, D. Xu, and M. Tan, "Trajectory prediction of spinning ball based on fuzzy filtering and local modeling for robotic ping–pong player," *IEEE Trans. Instrum. Meas.*, vol. 62, no. 11, pp. 2890–2900, Nov. 2013.

[23] L. Sun, J. Liu, Y. Wang, L. Zhou, Q. Yang, and S. He, "Ball's flight trajectory prediction for table-tennis game by humanoid robot," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2009, pp. 2379–2384.

[24] W. Yingshi, S. Lei, L. Jingtai, Y. Qi, Z. Lu, and H. Shan, "A novel trajectory prediction approach for table-tennis robot based on nonlinear output feedback observer," in *Proc. IEEE Int. Conf. Robot. Biomimetics (ROBIO)*, Dec. 2010, pp. 1136–1141.

[25] G. Nagy, *Ordinary Differential Equations, Teaching Note* (Mathematics Department). East Lansing, MI, USA: Michigan State Univ., USA, Oct. 2014. [Online]. Available: http://math.msu.edu/ gnagy/teaching/ode.pdf



Yongsheng Zhao received the B.Sc. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 2008, where he is currently pursuing the Ph.D. degree with the Institute of Cyber System and Control.

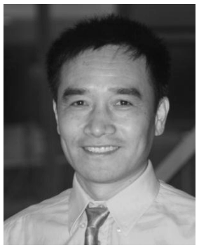
His current research interests include machine learning and robotics.

Yifeng Zhang received the B.Sc. and M.Sc. degrees in control science and engineering from Zhejiang University, Hangzhou, China, in 2006 and 2008, respectively, where he is currently pursuing the Ph.D. degree with the Institute of Cyber System and Control.

His current research interests include machine vision, motion modeling, and robotics.

Rong Xiong received the B.Sc. and M.Sc. degrees in computer science and engineering and the Ph.D. degree in control science and engineering from Zhejiang University, Hangzhou, China, in 1994, 1997, and 2009, respectively.

She has been with the State Key Laboratory of Industrial Control Technology, Zhejiang University, since 1997, where she is currently a Professor with the State Key Laboratory of Industrial Control Technology and directs the Robotics Laboratory. Her current research interests include machine vision, simultaneous localization and mapping, motion planning, and control for humanoid robots.



Jianguo Wang (M'06) received the B.Sc. degree in physics from Nanjing University, Nanjing, China, in 1983, the master's degree in image processing from Sydney University, Sydney, NSW, Australia, in 2001, and the Ph.D. degree in sensor fusion for navigation from the University of New South Wales, Sydney, in 2007.

He has been with the University of Technology, Sydney, since 2009, as a Lecturer, and a Research Fellow with the Faculty of Engineering and Information Technology. His current research interests include robotics, navigation, mapping and automation, and in particular, machine vision and sensor fusion.