Dynamic model based ball trajectory prediction for a robot ping-pong player

Article · December 2010

DOI: 10.1109/ROBIO.2010.5723394

5 authors, including:

Xiaopeng Chen
Beijing Institute of Technology
33 PUBLICATIONS **107** CITATIONS
SEE PROFILE

Zhangguo Yu
Beijing Institute of Technology
76 PUBLICATIONS **285** CITATIONS
SEE PROFILE

Proceedings of the 2010 IEEE
International Conference on Robotics and Biomimetics
December 14-18, 2010, Tianjin, China

Dynamic Model based Ball Trajectory Prediction for a Robot Ping-Pong Player

Xiaopeng Chen, Ye Tian, Qiang Huang, *Member, IEEE,* Weimin Zhang, Zhangguo Yu

***Abstract*—In this paper, a novel ball trajectory prediction method is proposed for a robot ping-pong player. The method is based on the dynamic model in the sky and the bouncing dynamic model between the ball and the table. First, the ambiguity of the dynamic model in the sky is removed. Then, the bouncing model between the ball and the table is proposed to get the relationship of the emergence speed and the incidence speed. The corresponding coefficients of models are trained according to the 3D Cartesian coordinates of the ball calculated from a high speed stereo vision system. With known dynamic models, the trajectory can be predicted if several initial trajectory points of the ball from the vision system are given. Experiments show that the trajectory is able to be predicted accurately in several milliseconds. The result of our method is better than that by LWR based approach.**

I. Introduction

PING pong sport is very popular all over the world. Are we able to construct a robot ping-pong player to play with human players? This idea has existed for a long time. Join Billingsley suggested robot ping pong sport in 1983[1] [2]. Since then, many robot ping-pong players have been built. Such as ping-pong robots in [3] [4] [5], etc. A review of robot ping-pong players is proposed in [6].

Most of the robot ping-pong players used a high speed 3D vision system to measure three dimensional positions of the ball[7, 8]. The vision system is settled to stare at the opposite side of the robot ping pong player. Once the ball is flying from opponent's side, the vision system reconstructs several 3D coordinates of the ball (trajectory I of figure 1). Taken these coordinates as inputs, the prediction process is executed to estimate the future trajectory of the ball (trajectory II, III, IV in fig. 1).

This paper mainly focuses on the ping-pong ball trajectory prediction problem. There are two categories of approaches. One category is non-model based approaches, such as [9-11]. Paper [10] proposed a classical LWR (Locally weighted Regression) based trajectory prediction algorithm and is frequently used. Usually learning algorithms are used in non-model based approaches to build a map between the inputs and outputs. As the trajectory is impacted by many factors, the relationship of the inputs and outputs are nonlinear and complicated. That's why it is difficult to improve the prediction performance for the non-model based approaches. The other is model based approaches such as [12, 13]. In paper [12], the Magnus force was assumed to be proportional to spin rate and to speed squared. However, the Magnus force was proportional to spin rate and to speed in [14] [15] [16]. Paper [17] gave explanations on why there were two different expressions on Magnus force. In [13], the bouncing model is determined by the vertical restitution coefficient and the horizontal coefficient. However, Cross [18] [19] pointed out that the horizontal restitution coefficient was not constant. In paper [12] Andersson pointed out that the result of a bounce is determined by the vertical restitution coefficient, ball's spin and the friction coefficient. Many researchers assumed that when bouncing occurs, the speed of the bottom point of the ball was zero [20]. However, this is not always the case. Obviously, there are ambiguities in model based approaches.

In this paper, a dynamic model based ball trajectory prediction method is proposed. The ambiguities of the aerodynamic model and the bouncing model are removed. In the method, the aerodynamic model is revisited and the bouncing model is analyzed in detail. In the bouncing model, the vertical speed and the horizontal speed are separately treated. The relationship of the emergence speed and the incidence speed is derived under some simple assumptions. And then, the coefficients of the two models are trained according to real data calculated from the stereo vision system. Once the coefficients are derived, the two dynamic models can be used for ball trajectory prediction. With the aid of the vision system, the prediction errors can be measured exactly by experiments. The results show that our method is feasible.

The remainder of this paper is organized as follows. Section II describes the dynamic models of the ping-pong ball. Section III describes the training process for coefficients calculation. Section IV describes the trajectory prediction process. Experiments are presented in Section V. A conclusion is given in Section VI.

Manuscript received July 21, 2010. This work was supported by National High-Technology Research and Development Program (No 2007AA04Z241, No 2008AA042601), the National Natural Science Foundation of China (No.61005081, No. 60874048, No. 60705025) and the 111 Project (No. B08043).

X. Chen is with the School of Mechatronics, Beijing Institute of Technology, Beijing, 100081, China (e-mail: xpchen@bit.edu.cn)

Y Tian is student of the School of Mechatronics, Beijing Institute of Technology, Beijing, 100081, China (e-mail: tianye7248@bit.edu.cn)

Q Huang is with the School of Mechatronics, Beijing Institute of Technology, Beijing, 100081, China (e-mail: qhuang@bit.edu.cn)

W Zhang is with the School of Mechatronics, Beijing Institute of Technology, Beijing, 100081, China(e-mail: zhwm@bit.edu.cn)

Z Yu is with the school of Mechatronics, Beijing Institute of Technology, Beijing, 100081, China (e-mail: yuzg@bit.edu.cn)

978-1-4244-9317-3/10/$26.00 © 2010 IEEE

II. Dynamic Models of The Ball

The ball trajectory is vital for successful ball hitting. In order to predict the trajectory, the aerodynamic model of the ball flying in the sky and the bouncing model are established in this section. The aerodynamic model can be used to estimate the trajectories in the sky. The bouncing model can be used to get the velocity changes after bouncing. Look at Figure 1, curve I, II, IV, V are trajectories in the sky. The bouncing takes place in point III.

There are some parameters of the ball should be mentioned. According to the standard, the diameter D is 0.04m, the radius r is 0.02m, the mass m is 0.0027kg, $\boldsymbol{g}$ is the acceleration of gravity. It is 9.8015m/s^2 in Beijing, China. ρ_a is the density of the air. The standard value is 1.29kg/m^3. The inertia along the diameter is $I = 2mr^2/3$. The world coordinate frame of the prediction system is defined like this: the origin is on the farthest right of the table from the robot's view; the Z axis points to upper side of the table vertically; the Y axis is parallel to the longer edge of the table and points to the robot; the X axis is defined accordingly to comply with the right hand rule. The frame is illustrated in figure 1. $\boldsymbol{p}$, $\boldsymbol{v}$, $\boldsymbol{\omega}$, $\boldsymbol{a}$ are the position, velocity, angular velocity and acceleration in the coordinate frame. When bouncing, the incidence velocity and spin are $\boldsymbol{v}_i$, $\boldsymbol{\omega}_i$, and the emergence velocity and spin are $\boldsymbol{v}_e$ and $\boldsymbol{\omega}_e$.

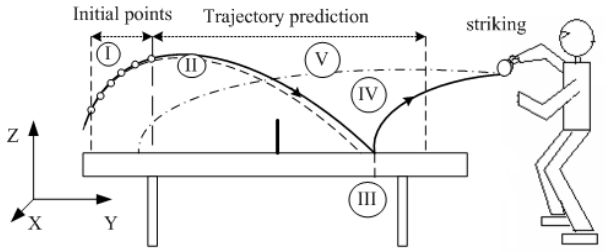

Fig. 1. Demo picture of the ping-pong ball flying in the sky and bouncing off.

A. Aerodynamic Model

A ball flying in the sky is given four forces. They are the gravity, the air resistance, the buoyancy force and the Magnus force. The gravity has the following form:

$$\boldsymbol{F}_g = m\boldsymbol{g} \tag{1}$$

The buoyancy force is equal to the weight of the air occupied by the ball.

$$\boldsymbol{F}_b = -m_b\boldsymbol{g} \tag{2}$$

The air resistance of a ball flying in the sky is proportional to the square of the velocity. The air resistance direction is opposite to the velocity of the ball. It has the following form [14]:

$$\boldsymbol{F}_d = -\frac{1}{8}C_D\rho_a\pi D^2\|\boldsymbol{v}\|\boldsymbol{v} \tag{3}$$

C_D is the drag coefficient. For the table tennis, C_D is about 0.4~0.6.

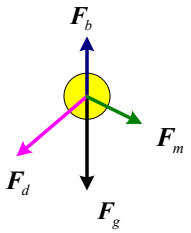

Fig. 2. The four forces given to the ball in the sky.

There are two opinions about the Magnus force. One of them stated that the Magnus force was proportional to spin rate and to speed squared [12]. The other stated that the Magnus force was proportional to spin rate and to speed in [14]. From [17] it can be inferred that the latter is true. It is

$$\boldsymbol{F}_m = \frac{1}{8}C_m\rho_a\pi D^3\boldsymbol{\omega}\times\boldsymbol{v} \tag{4}$$

according to [14]. Where C_m is the Magnus coefficient. For table tennis, it is usually 1.

And so the resultant force is

$$\boldsymbol{F} = m\boldsymbol{g} - m_b\boldsymbol{g} - \frac{1}{8}C_D\rho_a\pi D^2\|\boldsymbol{v}\|v + \frac{1}{8}C_m\rho_a\pi D^3\boldsymbol{\omega}\times\boldsymbol{v} \tag{5}$$

Because $m_b << m$, the buoyancy force is not taken for consideration.

The resultant acceleration is

$$\boldsymbol{a} = \boldsymbol{g} - k_d\|\boldsymbol{v}\|\boldsymbol{v} + k_m\boldsymbol{\omega}\times\boldsymbol{v} \tag{6}$$

Where

$$k_d = \frac{C_D\rho_a\pi D^2}{8m} \tag{7}$$

and

$$k_m = \frac{C_m\rho_a\pi D^3}{8m} \tag{8}$$

According to the theoretic calculation, k_d is about 0.15. It should not be neglected. k_m is 0.012.

When there is no spin, the acceleration would be

$$\boldsymbol{a} = \boldsymbol{g} - k_d\|\boldsymbol{v}\|\boldsymbol{v} \tag{9}$$

Equation (6) can be rewritten as

$$\dot{\boldsymbol{v}}(t) = \begin{bmatrix} -k_d v & -k_m\omega_z & k_m\omega_y \\ k_m\omega_z & -k_d v & -k_m\omega_x \\ -k_m\omega_y & k_m\omega_x & -k_d v \end{bmatrix}\boldsymbol{v}(t) + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \tag{10}$$

And equation (9) can be rewritten as

$$\dot{\boldsymbol{v}}(t) = \begin{bmatrix} -k_d v & 0 & 0 \\ 0 & -k_d v & 0 \\ 0 & 0 & -k_d v \end{bmatrix}\boldsymbol{v}(t) + \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \tag{11}$$

The position can be integrated from the velocities.

$$\dot{\boldsymbol{p}}(t) = \boldsymbol{v}(t) \tag{12}$$

If k_d , k_m , $\boldsymbol{v}_0$, $\boldsymbol{\omega}_0$, $\boldsymbol{p}_0$ are known, and $\boldsymbol{\omega}$ is constant, the trajectory in the sky can be obtained by (10) or (11).

B. Bouncing Model

Existing detailed bouncing models can be found in [13, 21] and [20]. In [13] and [21], the authors assumed that the

horizontal restitution coefficient was constant. In [20], lowest velocity of the ball was always zero when bouncing off, which implied that the horizontal restitution coefficient was zero. However, these assumptions showed that the emergence velocities had no relationship with the friction coefficient. This is not the case. Andersson's assumption [12] was much more reasonable, however, no detailed description of the bouncing model was presented. In this paper, similar to [12], it is assumed that:

1) The emergence velocity in Z direction is proportional to the incidence velocity in Z direction. The proportion coefficient is the vertical restitution coefficient. That is:

$$v_{ez} = -k_v v_{iz} \tag{13}$$

2) The emergence spin in Z direction is equal to the incidence spin in Z direction.

If the normal direction of table is vertical, the action point of the horizontal forces is the lowest point of the ball keeps in touch with the table. So there is no reason to impose a torque in Z direction to the ball. Thus the spin should be constant.

$$\omega_{ez} = \omega_{iz} \tag{14}$$

3) The horizontal velocity direction of the lowest point could not be changed by the friction. In worst case the horizontal velocity can be zero.

If the above assumptions are satisfied, the emergence velocity and spin can be calculated uniquely. There are two cases. In the first case, the emergence horizontal velocity of the lowest point is not zero. In the second case, the emergence horizontal velocity of the lowest point is zero.

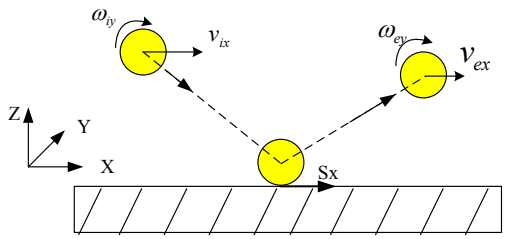

Fig. 3. bouncing model in XOZ plane.[1]

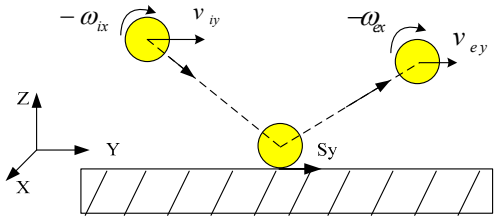

Fig. 4. bouncing model in YOZ plane.[1]

1) the emergence horizontal velocity of the lowest point is not zero

In this case, force of sliding friction is given to the ball during the bouncing process. Its direction is opposite to the incidence horizontal velocity. Its magnitude is proportional to the vertical press. The impulse of the friction force can be calculated according to:

$$F \cdot \Delta t = \mu N \cdot \Delta t \approx \mu m(v_{ez} - v_{iz}) = \mu m(1 + k_v)\left|v_{iz}\right| \tag{15}$$

And so the impulse in the x direction is

$$S_x = -F \cdot \Delta t \cdot \cos\theta = -\mu m(1 + k_v)\left|v_{iz}\right|\cos\theta \tag{16}$$

$$S_y = -\mu m(1 + k_v)\left|v_{iz}\right|\sin\theta \tag{17}$$

Where θ is the angle between the velocity of the lowest point and the x axis. The minus symbol indicates that the sliding friction direction is opposite to the direction of the velocity of the lowest point.

From figure 3 the following could be obtained:

$$\begin{cases} v_{bix} = v_{ix} - \omega_{iy} r \\ S_x = m(v_{ex} - v_{ix}) \\ S_x r = I(\omega_{iy} - \omega_{ey}) \end{cases} \tag{18}$$

Where v_{bix} is the incidence horizontal velocity of the lowest point in x axis. Accordingly, from figure 4 equation (19) is obtained:

$$\begin{cases} v_{biy} = v_{iy} + \omega_{iy} r \\ S_y = m(v_{ey} - v_{iy}) \\ S_y r = I(\omega_{ex} - \omega_{ix}) \end{cases} \tag{19}$$

Thus

$$\cos\theta = \frac{v_{bix}}{\sqrt{v_{bix}^2 + v_{biy}^2}} \tag{20}$$

$$\sin\theta = \frac{v_{biy}}{\sqrt{v_{bix}^2 + v_{biy}^2}} \tag{21}$$

From (16) to (21) the following equation is obtained:

$$\begin{cases} v_{ex} = \dfrac{\mu(1+k_v)\left|v_{iz}\right|(\omega_{iy} r - v_{ix})}{\sqrt{(v_{ix} - \omega_{iy} r)^2 + (v_{iy} + \omega_{ix} r)^2}} + v_{ix} \\ v_{ey} = \dfrac{\mu(1+k_v)\left|v_{iz}\right|(-\omega_{ix} r - v_{iy})}{\sqrt{(v_{ix} - \omega_{iy} r)^2 + (v_{iy} + \omega_{ix} r)^2}} + v_{iy} \\ v_{ez} = -k_v v_{iz} \end{cases} \tag{22}$$

$$\begin{cases} \omega_{ex} = \omega_{ix} + \dfrac{3\mu(1+k_v)\left|v_{iz}\right|(-\omega_{ix} r - v_{iy})}{2r\sqrt{(v_{ix} - \omega_{iy} r)^2 + (v_{iy} + \omega_{ix} r)^2}} \\ \omega_{ey} = \omega_{iy} - \dfrac{3\mu(1+k_v)\left|v_{iz}\right|(\omega_{iy} r - v_{ix})}{2r\sqrt{(v_{ix} - \omega_{iy} r)^2 + (v_{iy} + \omega_{ix} r)^2}} \\ \omega_{ez} = \omega_{iz} \end{cases} \tag{23}$$

The assumption that the horizontal velocity direction of the lowest point could not be changed should be obeyed. The lowest velocity after bouncing is

$$\begin{cases} v_{bex} = v_{bix}\left(1 - \dfrac{2.5\mu(1+k_v)\left|v_{iz}\right|}{\sqrt{(v_{ix} - \omega_{iy} r)^2 + (v_{iy} + \omega_{ix} r)^2}}\right) \\ v_{bey} = v_{biy}\left(1 - \dfrac{2.5\mu(1+k_v)\left|v_{iz}\right|}{\sqrt{(v_{ix} - \omega_{iy} r)^2 + (v_{iy} + \omega_{ix} r)^2}}\right) \end{cases} \tag{24}$$

So the condition

$$\frac{\mu(1+k_v)|v_{iz}|}{\sqrt{(v_{ix}-\omega_{iy}r)^2+(v_{iy}+\omega_{ix}r)^2}} \le 0.4 \tag{25}$$

must be satisfied.

2) the emergence horizontal velocity of the lowest point is zero

if equal condition of (25) is satisfied, the emergence horizontal velocity of the lowest point is zero. Thus

$$\begin{cases} v_{ex} = 0.4\omega_{iy}r + 0.6v_{ix} \\ v_{ey} = -0.4\omega_{ix}r + 0.6v_{iy} \\ v_{ez} = -k_v v_{iz} \end{cases} \tag{26}$$

$$\begin{cases} \omega_{ex} = 0.4\omega_{ix} - 0.6v_{iy}/r \\ \omega_{ey} = 0.4\omega_{iy} + 0.6v_{ix}/r \\ \omega_{ez} = \omega_{iz} \end{cases} \tag{27}$$

Equations (26)(27) are still right if

$$\frac{\mu(1+k_v)|v_{iz}|}{\sqrt{(v_{ix}-\omega_{iy}r)^2+(v_{iy}+\omega_{ix}r)^2}} > 0.4 \tag{28}$$

is true to satisfy the condition that the emergence horizontal velocity should be zero. However, the impulse is generated by both the sliding friction and the rotating friction.

So, if the incidence velocity, incidence spin, the vertical restitution coefficient k_v and the sliding friction coefficient μ are known, the emergence velocity and spin can be calculated according to (26)(27) if (28) is satisfied, or according to (22)(23) if (28) is not satisfied.

III. Training for Initial Values and Coefficients

From last section, it is obvious that if the initial values $\boldsymbol{v}_0$, $\boldsymbol{\omega}_0$, $\boldsymbol{p}_0$ and the coefficients k_d, k_m, k_v, μ are derived, the whole trajectory could be predicted accordingly.

In order to obtain the initial values, sample trajectory data observed from stereo vision system are fitted to be polynomial expressions. The polynomial coefficients are then derived by least mean square method. In this way, the observation errors are suppressed.

The model coefficients k_d, k_m, k_v, μ can also be trained from sample trajectory data. The detail is illustrated below.

A. Initial Values

Given sample data array $\{\boldsymbol{p}_i, t_i\}_{i=1..M}$, an n degree polynomial is defined

$$\boldsymbol{p}_n(t) = \sum_{k=0}^{n} \boldsymbol{\alpha}_k t^k \tag{29}$$

Where $\boldsymbol{\alpha}_k$ satisfies

$$F(\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1, ..., \boldsymbol{\alpha}_n) = \min_{\boldsymbol{\alpha}_0, \boldsymbol{\alpha}_1, \cdots, \boldsymbol{\alpha}_n} \sum_{i=1}^{M} \left\| \boldsymbol{p}_i - \boldsymbol{p}(t_i) \right\|^2 \tag{30}$$

Solving

$$\frac{\partial F}{\partial \boldsymbol{\alpha}_j} = 2\sum_{i=1}^{M}[\sum_{k=0}^{n} \boldsymbol{\alpha}_k t_i^k - \boldsymbol{p}_i]t_i^j = 0 \quad j = 1 \cdots n \tag{31}$$

The optimal coefficients can be obtained by (31). Once the coefficients $\boldsymbol{\alpha}_k$ are obtained, then the initial position $\boldsymbol{p}_0 = \boldsymbol{p}_n(t_0)$ and the initial velocity $\boldsymbol{v}_0 = \dot{\boldsymbol{p}}_n(t_0)$ can be derived. In order to obtain the angular velocity, equation (10) can be rewritten as

$$\begin{bmatrix} 0 & v_z & -v_y \\ -v_z & 0 & v_x \\ v_y & -v_x & 0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \frac{1}{k_m} \begin{bmatrix} a_x + k_d v v_x \\ a_y + k_d v v_y \\ a_z + k_d v v_z + g \end{bmatrix} \tag{32}$$

However, there is no solution for equation (32). Differentiate (32) with time t, another equation is derived.

$$\begin{bmatrix} 0 & a_z & -a_y \\ -a_z & 0 & a_x \\ a_y & -a_x & 0 \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} = \frac{1}{k_m} \begin{bmatrix} \dot{a}_x + k_d \dot{v} v_x + k_d v a_x \\ \dot{a}_y + k_d \dot{v} v_y + k_d v a_y \\ \dot{a}_z + k_d \dot{v} v_z + k_d v a_z \end{bmatrix} \tag{33}$$

From the above two equations the angular velocity could be derived from $\boldsymbol{v}$, $\boldsymbol{a}$, and $\dot{\boldsymbol{a}}$ if k_d and k_m are known.

B. Coefficients

In order to predict the trajectory, the coefficients k_d, k_m, k_v, μ must be obtained. They are trained by minimizing the maximal errors between the vision data and the predicted data.

IV. Trajectory Prediction

Once the coefficients and initial parameters are derived, equation (10)(22)(23) or(10) (26)(27) can be used to predict the whole trajectory. The discrete form of equation (12) is

$$\boldsymbol{p}(t_i) = \boldsymbol{p}(t_{i-1}) + \boldsymbol{v}(t_{i-1})(t_i - t_{i-1}) \tag{34}$$

The discrete form of equation (10) is

TABLE I
DYNAMIC MODEL BASED TRAJECTORY PREDICTION

1)	Capture initial points in the curve I (fig. 1.) by stereo vision system.
2)	The data is fitted according to equation (31) to suppress noise to get the polynomial.
3)	From equation(29) and its first three order differentiations, the initial position, velocity, acceleration and jerk are derived.
4)	The initial spin is calculated from equation (32) and (33).
5)	Using equation (34)(35) to predict the trajectory II of fig 1 until the ball hits the table at point III.
6)	Judge the bounce condition using equation (25), and then the emergence velocity and angular velocity are derived according to (23)(24) or (26)(27).
7)	Trajectory IV is predicted using the same routine as 5)

$$\begin{aligned} \boldsymbol{v}(t_i) = \boldsymbol{v}(t_{i-1}) &+ \begin{bmatrix} -k_d v & -k_m\omega_z & k_m\omega_y \\ k_m\omega_z & -k_d v & -k_m\omega_x \\ -k_m\omega_y & k_m\omega_x & -k_d v \end{bmatrix} \boldsymbol{v}(t_{i-1})(t_i - t_{i-1}) \\ &+ \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} (t_i - t_{i-1}) \end{aligned} \tag{35}$$

Given initial position and velocities, the following velocities are updated according to (35), the following positions are updated according to (34). The overall trajectory

prediction algorithm is list in Table I.

V. EXPERIMENTS

In order to verify the prediction process, the flying trajectory prediction as well as the bouncing trajectory prediction experiment is done. The result is compared with the LWR approach[10] and the original vision data captured by four cameras based vision system. In the LWR based approach, the aerodynamic model is the same as this paper. The bouncing model is simplified by the following equation

$$\begin{cases} v_{bex} = kx \cdot v_{bix} + cx \\ v_{bey} = ky \cdot v_{biy} + cy \\ v_{bez} = kz \cdot v_{biz} + cz \end{cases} \quad (36)$$

The coefficients kx , ky , kz , cx , cy , cz are trained by experimental data. In our experiment it is

$$\begin{cases} kx = 0.648819 \\ cx = 0.010491 \\ ky = 0.643550 \\ cy = 0.024064 \\ kz = -0.707392 \\ bz = 0.468110 \end{cases} \quad (37)$$

In our method, kd , km , kv , μ are trained by minimizing the maximal errors between the vision data and the predicted data. In the experiment, the trained coefficients $kd = 0.1196$, $km = 0.003$, $kv = 0.97$ and $\mu = 0.22$.

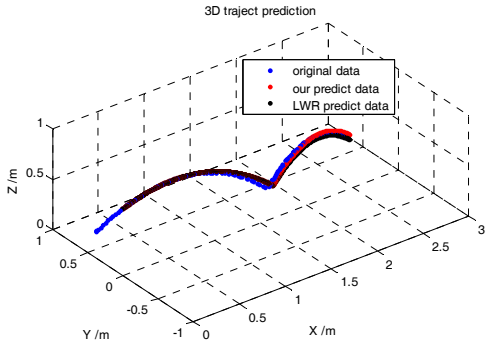

Fig.5 . 3D vision data versus predicted trajectories.

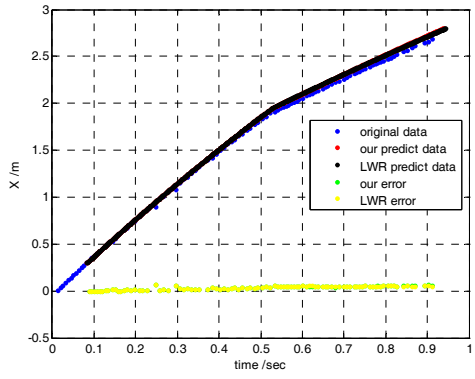

Fig.6 . Vision data versus predicted trajectories in X axis.

In the experiment, the whole trajectory is recorded. The initial 25 points of trajectory I are taken as inputs. From the input data, initial parameters are estimated by (29)~(33). From the estimated initial parameters, the trajectory is then predicted accordingly. When the ping pong ball hits the table, the LWR bouncing model and the nonlinear bouncing model in this paper are separately used for prediction. Look at figure 5, the blue trajectory is the vision data captured. The black trajectory is the LWR based approach. Figure 5 shows that the trajectories have been successfully predicted. The three trajectories are close to each other. Errors in each dimension are shown in figure 6, 7 and 8. The mean absolute errors with respect to the original vision data are listed in Table II.

TABLE II
THE MEAN ABSOLUTE PREDICTION ERRORS

method	X error /mm	Y error /mm	Z error /mm
Our approach	8.1031	5.8838	5.0517
LWR	8.5443	8.3472	23.0306

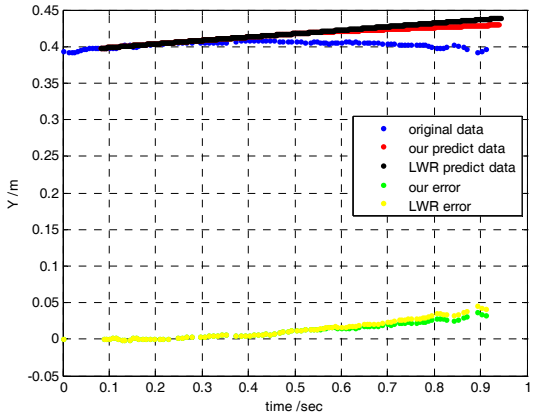

Fig. 7. Vision data versus predicted trajectories in Y axis.

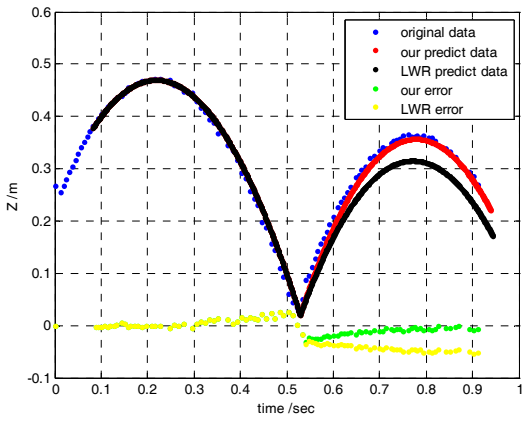

Fig. 8.Vision data versus predicted trajectories in Z axis.

From above pictures we can see that our method is better than LWR's approach. The mean absolute error of our

method in each direction is less than 1 cm. While for the LWR's approach, the trajectory in Z axis is not so good and the mean absolute error is about 2.3 cm. In our prediction algorithm, the time step of each iteration is 0.1ms. The ping pong ball should consume about 1 second to travel the whole trajectory. That is to say, about 10000 iterations are needed to predict the whole trajectory. It consumes only 3-4 ms for a PC Computer.

VI. CONCLUSION

In this paper, a non linear model based ball trajectory prediction method has been proposed. In this method, the aerodynamic model is revisited and the bouncing model is proposed. The corresponding coefficients are trained by experiments. With the two models and the initial inputs, the whole trajectory has been predicted in a few milliseconds. Our method is compared with the LWR based approach and the original vision data. Experiments show that the trajectory has been predicted successfully by our method. Our method is better than LWR based approach.

The main contribution of our method is that the bouncing model is proposed. The model is determined by the vertical restitution coefficient and the friction coefficient. Once the two coefficients are obtained, the emergence velocity and spin can be derived if the incidence velocity and spin are known. This bouncing model is also applicable for collision between the ball and the racket. However, the vertical restitution coefficient and the friction coefficient may be different. The advantage of the nonlinear method is that the spin changes can also be solved analytically, which is vital for predicting the trajectories of rotating balls.

In our method, the initial position and velocities are needed for iteration. The initial parameters are vital for trajectory prediction. In our experiment, they are derived from polynomial curve fitting. In order to improve the robustness, the vision data should be filtered by a Kalman filter in advance. The RANSAC algorithm instead of the least square method is preferred to estimate the initial parameters.

REFERENCES

[1] Z. Yu, "Distributed Control Structure and Stability Control Based on Waist Motion for a Humanoid Robot," in *School of Mechetronics*. vol. Doctor's Degree Beijing: Beijing Institute of Technology, 2009.

[2] J. Billingsley, "Robot ping pong," *Practical Computing,* 1983.

[3] M. Takeuchi, F. Miyazaki, M. Matsushima, et al., "Dynamic dexterity for the performance of" wall-bouncing" tasks," in *IEEE International Conference on Robotics and Automation*, 2002, pp. 1559-1564.

[4] R. L. Andersson, "Understanding and applying a robot ping-pong player's expertcontroller," 1989, pp. 1284-1289.

[5] L. Acosta, J. J. Rodrigo, J. A. Mendez, et al., "Ping-pong player prototype," *Ieee Robotics & Automation Magazine,* vol. 10, pp. 44-52, 2003.

[6] Z. Zhang, J. Y. De Xu, "Research and latest development of Ping-Pong robot player," 2008, pp. 4881-4886.

[7] R. L. Andersson, "A low-latency 60 Hz stereo vision system for real-time visualcontrol," 1990, pp. 165-170.

[8] Z. Zhang, D. Xu, "High-Speed Vision System Based on Smart Camera and Its Target Tracking Algorithm," *Robots,* vol. 31, pp. 229-234, 2009.

[9] M. Matsushima, T. Hashimoto, F. Miyazaki, "Learning to the Robot Table Tennis Task-Ball Control and Rally with a Human," 2003, pp. 2962-2969.

[10] Q. Rui, "The simulation research of the Ping-Pong orbit prediction by LWR," *Robot,* vol. 20, pp. 373–377, 1998.

[11] M. Matsushima, T. Hashimoto, M. Takeuchi, et al., "A Learning Approach to Robotic Table Tennis," *Ieee Transactions on Robotics,* vol. 21, pp. 767-771, 2005.

[12] R. L. Andersson, "Dynamic sensing in a ping-pong playing robot," *Ieee Transactions on Robotics and Automation,* vol. 5, pp. 728-739, 1989.

[13] Bo peng, Yongchao Hong, Sensen Du, et al., "An Approach to Hit Point Prediction for Ping Pong Robot," *Journal of Jiangnan University (Natural Science Edition),* vol. 6, pp. 433-437, 2007.

[14] R. K. Adair, P. J. Brancazio, "The physics of baseball," *American Journal of Physics,* vol. 58, p. 1117, 1990.

[15] S. Rusdorf, G. Brunnett, M. Lorenz, et al., "Real-time interaction with a humanoid avatar in an immersive table tennis simulation," *Ieee Transactions on Visualization and Computer Graphics,* vol. 13, pp. 15-25, 2007.

[16] H. Pan, "Mechanical Model of Magnus Effect," *ZheJiang Sports Science,* vol. 17, pp. 16-19, 1995.

[17] I. Griffiths, C. Evans, N. Griffiths, "Tracking the flight of a spinning football in three dimensions," *Measurement Science and Technology,* vol. 16, p. 2056, 2005.

[18] R. Cross, "Measurements of the horizontal coefficient of restitution for a superball and a tennis ball," *American Journal of Physics,* vol. 70, p. 482, 2002.

[19] R. Cross, "Bounce of a spinning ball near normal incidence," *American Journal of Physics,* vol. 73, p. 914, 2005.

[20] G. Meng, Y. Chen, "Mechanical Analysis on Table Tennis Sport," *Journal of Northeast Heavy Machinery Institute,* vol. 20, pp. 80-83, 1996.

[21] D. Guo, J. Li, S. Pan, "Mechanical Model of Collision between Table Tennis Ball and Table," *ZheJiang Sports Science,* vol. 18, pp. 43-45, 39, 1996.