JOURNAL OF THEORETICAL
AND APPLIED MECHANICS
62, 3, pp. 561-571, Warsaw 2024
https://doi.org/10.15632/jtam-pl/188373

PREDICTING THE TRAJECTORY OF A SPINNING PING PONG BALL DURING FLIGHT USING THREE-DIMENSIONAL COORDINATES

Xinyue Li
Nanjing Forestry University, Department of Physical Education, Nanjing, Jiangsu, China
e-mail: y3z411@yeah.net

Predicting the trajectory of a spinning ping pong ball can improve the effectiveness of a ping pong robot in daily training. In this study, the Vicon system was used to capture three-dimensional coordinates of the spinning ping pong ball during flight. Then, a long short-term memory (LSTM) neural network algorithm was improved by combining an adaptive particle swarm optimization (APSO) algorithm and the attention mechanism, and the APSO-LSTM-attention method was obtained for predicting the trajectory of the spinning ping pong ball. It was found through experiments that the APSO-LSTM-attention method had average displacement errors of 6.01 mm, 11.26 mm, and 8.97 mm in the X, Y and Z axes, respectively, and the final point displacement errors were 15.64 mm, 17.93 mm, and 11.26 mm, respectively, indicating that the method outperformed methods such as recurrent neural networks. The time required to predict the complete trajectory by the APSO-LSTM-attention method was also short, only 0.0186 s. The results demonstrate reliability of the proposed method in predicting the trajectory of the spinning ping pong ball and its potential application in practical scenarios.

Keywords: three-dimensional coordinates, ping pong ball, trajectory prediction

1. Introduction

Ping pong, the national game of China, is popular with the public. China has won many medals in various international competitions. For athletes, burying their heads in training can often lead to a bottleneck in technique. How to breakthrough their skills is quite concerned by athletes. With the development of intelligent algorithms and artificial intelligence, ping pong ball robots have become a new tool for daily training players (Gomez-Gonzalez *et al.*, 2019). In the field of ping pong, predicting the trajectory of a ball can help athletes practice better, which effectively reduces manpower costs and improves training efficiency. In other fields, the prediction of missile trajectories in military operations (Mir, 2018), aircraft trajectories (Huang *et al.*, 2021), trajectories of human body motion during transport driving (Bertugli *et al.*, 2021), driving trajectories (Amirloo *et al.*, 2022), and trajectories of athletes during sports (Hauri *et al.*, 2021) can greatly improve research efficiency. With the advancement of intelligent algorithms, research on trajectory prediction has become increasingly widespread worldwide. Kalatian and Farooq (2022) developed a new multi-input network based on long short-term memory (LSTM) and fully connected dense layers for predicting future pedestrian trajectories. The experimental results showed small prediction errors with this method. Xi *et al.* (2021) designed a prediction model for a target maneuvering trajectory, introducing the Levenberg-Marquardt and improved particle swarm optimization (IPSO) algorithms mixed with k-means for optimizing parameters of the radial basis function. Simulation experiments demonstrated high accuracy of this model. Mirmohammad *et al.* (2021) investigated trajectory prediction of soccer balls on a soccer field, proposed a method based on the K-nearest neighbor regression and autoregressive model, and proved high accuracy of the method through simulation and practical testing. Chen *et al.*

(2021) developed an end-to-end fully convolutional coding and decoding attention model based on convolutional LSTM, which was found to have excellent performance in predicting future trajectories of pedestrians through experiments on five crowded video sequences. Song *et al.* (2022) proposed a bidirectional gated recurrent unit with an attention mechanism for prediction of tropical cyclone trajectories. Experimental results on the best path data of Northwest Pacific tropical cyclones from 1988 to 2017 demonstrated excellent performance of this model in predicting future trajectories. Chen *et al.* (2020) introduced a method that utilized a genetic algorithm to optimize the number of neurons and weights in a backpropagation neural network (BPNN) for ship trajectory prediction. The experimental findings indicated that this approach significantly enhanced the accuracy of predictions. Song *et al.* (2022) proposed a radar track prediction method based on the BPNN, compared its result with the Kalman filter track, and found that this method was highly accurate to forecast tracks. Rajini Selvaraj and Gurusamy (2023) integrated an independent recursive neural network, Harris Hawk optimization algorithm, and one-dimensional convolutional neural network autoencoder to forecast tropical cyclone trajectories. Comparison with the existing methods revealed that the method had higher prediction accuracy and efficiency. The trajectory prediction of spinning ping pong balls is the research focus of this paper. By collecting three-dimensional coordinates, a method based on an LSTM neural network was developed, and its performance was analyzed through experiments. This paper provides some theoretical support for promoting the development of intelligent robots, which is conducive to promoting the performance of table tennis robots and their application in practical sports training.

2. Collection of three-dimensional coordinates for a ping pong ball

This paper used a Vicon motion capture system with functions of motion capture and position tracking to measure the trajectory of ping pong balls (Rodrigues *et al.*, 2019; Goldfarb *et al.*, 2021). In a space formed by six cameras, data was captured by attaching reflective markers on ping pong balls, and the Vicon system was utilized to collect and calculate the three-dimensional coordinate data of the reflective markers in real time. The collected data was processed using the accompanying Tracker software to obtain information such as the velocity of the ping pong ball. The Vicon system consisted of the following components.

(1) Vicon cameras: These cameras had a resolution of 2432 × 3048 pixels and a maximum capture frequency of 420 Hz. The capture range was 12 m.
(2) PoE switch: It was used to connect with the host personal computer for data analysis.
(3) Host personal computer: The Tracker software was installed in the host personal computer for data capture, processing, and visualization.
(4) Calibration bar: It has applied to calibrate the Vicon cameras and establish the origin of the coordinate system.
(5) Other accessories: Cables to connect the cameras with the switch, reflective markers for capturing coordinates, and so on.

In the experiment, the environment for collecting the three-dimensional coordinates of the ping pong ball is illustrated in Fig. 1.

During the experiment, the researchers threw the ping pong ball, and the three-dimensional coordinates of the spinning flight of the ping pong ball were collected in real-time using the Vicon system. As the ping pong ball was a sphere with three symmetrical axes, six reflective markers were symmetrically attached to the ping pong ball along its symmetrical axes as reference points. The data was collected at a frequency of 180 Hz. The ping pong ball was thrown 500 times. Each trajectory started from the release and ended when it hit the ground. To further expand the dataset, the collected coordinates of each trajectory were translated and rotated, resulting

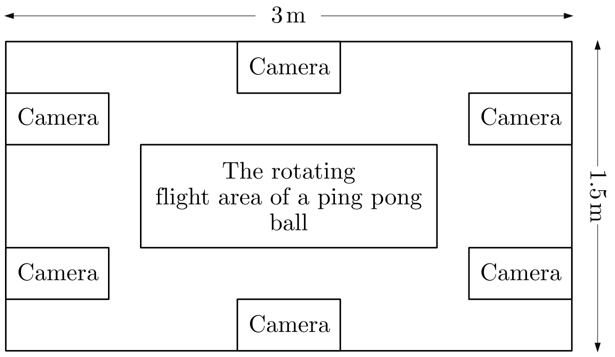

Fig. 1. Three-dimensional coordinate collection environment for a ping pong ball

in a total of 1500 trajectory data. The original three-dimensional coordinates of the ping pong ball are denoted by $L(x, y, z)$. After translating each coordinate by $L(x, y, z)$ units, the new coordinates are obtained as follows

$$L'(x', y', z') = \begin{bmatrix} 1 & 0 & 0 & x_0 \\ 0 & 1 & 0 & y_0 \\ 0 & 0 & 1 & z_0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.1}$$

Assuming that each coordinate rotates by $\theta°$ around the Z axis, the following coordinates are obtained

$$L'(x', y', z') = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2.2}$$

Some of the ping pong ball trajectory data is shown in Fig. 2 and Table 1.

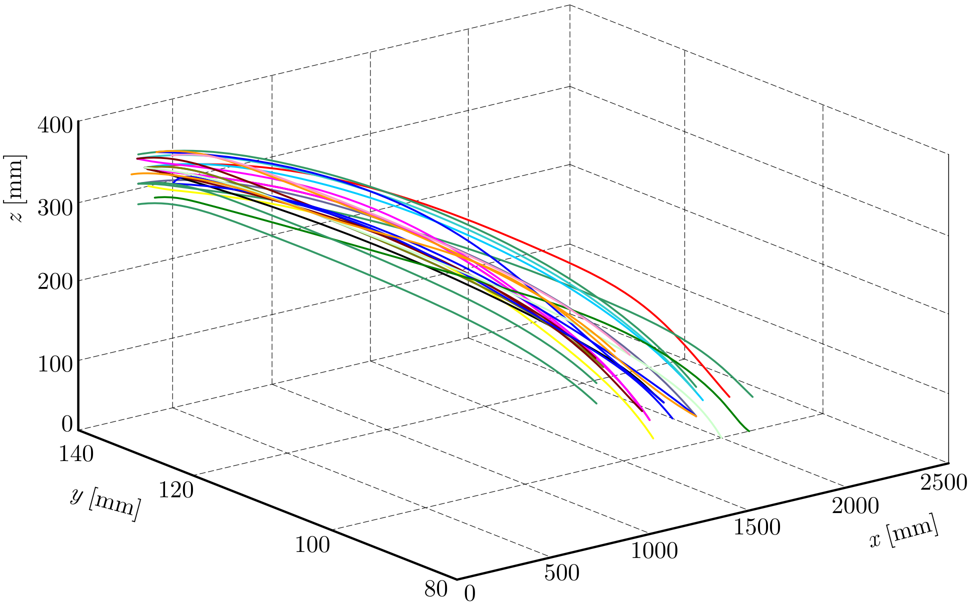

Fig. 2. Ping pong ball rotational flight trajectory data

Table 1. Example of ping pong ball trajectory data

Number	Three-dimensional coordinates [mm]
1	$(233.56, 137.88, 354.02)$
2	$(345.77, 131.25, 341.22)$
3	$(455.26, 127.36, 322.15)$
4	$(556.25, 121.26, 311.25)$
5	$(667.62, 115.33, 296.37)$
6	$(764.25, 111.25, 256.36)$
7	$(864.22, 107.36, 241.85)$
8	$(958.67, 103.22, 222.87)$
9	$(1065.25, 99.52, 195.74)$
10	$(1174.52, 97.28, 161.07)$

3. Long short-term memory-based trajectory prediction method

3.1. LSTM algorithm

The trajectory of a spinning ping pong ball during flight is a sequence with temporal properties. LSTM has demonstrated excellent performance in predicting time series (Kumar and Gomathi, 2022), and it has been extensively used in various domains such as weather forecasting and stock prediction (Gruet *et al.*, 2018). Therefore, in this study, LSTM is chosen to forecast the trajectory of the spinning ping pong ball during flight.

LSTM predicts data through three gates. First, let σ be the sigmoid activation function. Let W and b be the weight and bias of each gate. In the LSTM, the forgetting gate is used to determine how much information in the unit state value c^{t-1} from the previous moment needs to be forgotten. The input includes input information x^t from the current moment and output h^{t-1} from the previous moment. The output is

$$f_t = \sigma[W_f(h^{t-1}, x^t) + b_f] \tag{3.1}$$

The input gate is used to determine how much information can be input to the cell state. It includes two parts. The first part is to calculate how much information needs to be updated. The output is

$$i_t = \sigma[W_i(h^{t-1}, x^t) + b_i] \tag{3.2}$$

The other part is to calculate a new unit state candidate value $\overline{c}_t$. The calculation formula is

$$\overline{c}_t = \tanh[W_c(h^{t-1}, x^t) + b_c] \tag{3.3}$$

where tanh stands for the tangent function. Finally, the unit state value c^t is updated as

$$c^t = f_t c^{t-1} + i_t \overline{c}_t \tag{3.4}$$

The output calculates how much information can be output first. The corresponding formula is

$$o_t = \sigma[W_o(h^{t-1}, x^t) + b_o] \tag{3.5}$$

Then, the tanh function is combined to control the c^t value between -1 and 1. The final output of the LSTM is obtained after multiplication

$$h_t = o_t \tanh c^t \tag{3.6}$$

The learning process of LSTM is as follows:

- By performing forward calculations on the three gates, the output values of each neuron are obtained.
- The error between the output of each LSTM unit and the actual value is calculated, and all errors are summed up to obtain the total error.
- The weights are continuously updated through backward propagation of the error.
- It is checked whether the total error meets the accuracy requirement. If not, it returns to step one and repeats the calculation until the total error satisfies the accuracy requirement.

3.2. Attention mechanism

To further improve the effectiveness of LSTM on trajectory prediction, the attention mechanism (Zheng *et al.*, 2018) is added to better learn the input three-dimension coordinates of the ping-pong ball. The computational procedure of the attention layer is as follows:
— the attention probability distribution value at the t-th time is calculated using h_t, the output of the LSTM

$$e_t = v \tanh(wh_t + b) \tag{3.7}$$

— the normalized weight coefficient a_t is calculated

$$a_t = \frac{\exp(e_t)}{\sum_{j=1}^{t} e_j} \tag{3.8}$$

— the output of attention at the t-th is calculated

$$s_t = \sum_{t=1}^{i} a_t h_t \tag{3.9}$$

After passing the attention layer, the predictive trajectory value of the ping pong ball output at the t-th time is

$$y_t = \sigma(W_o s_t + b_o) \tag{3.10}$$

3.3. Parameter optimization methods

In LSTM, some parameters are usually determined based on empirical knowledge and require extensive experimentation for validation. This can significantly increase the training time of the algorithm. Therefore, this study employs an adaptive particle swarm optimization algorithm (APSO) to optimize the following parameters of LSTM:

- Learning rate: A value that is too small increases the learning time of the network, while a too large value may cause oscillations around the optimal value.
- Number of iterations: A value that is too small may prevent the network from achieving the best performance, while a value that is too large increases the training time.
- Number of hidden layer nodes: A value that is too small may result in underfitting, while a value that is too large may lead to overfitting.

The PSO algorithm is a method based on the foraging behavior of birds (Amiri *et al.*, 2023). It is known for its few parameters and high precision, and finds extensive applications in multi-objective optimization, industrial system control, and other fields (Bidyanath *et al.*, 2023).

Assuming a particle population $X = (x_1, x_2, \ldots, x_n)$ in a D-dimensional space, with initial positions $X = (X_{i1}, X_{i2}, \ldots, X_{iD})$ and initial velocities $V = (V_{i1}, V_{i2}, \ldots, V_{iD})$, individual and global best positions are denoted by P_{iD} and P_{gD}, respectively. The PSO algorithm updates the positions and velocities of particles to find the optimal solution. The formulas are as follows

$$\begin{aligned} V_{id}^{k+1} &= wV_{id}^{k} + c_1 r_1 (P_{id}^{k} - X_{id}^{k}) + c_2 r_2 (P_{gd}^{k} - X_{gd}^{k}) \\ X_{id}^{k+1} &= X_{id}^{k} + V_{id}^{k+1} \end{aligned} \tag{3.11}$$

where w stands for the inertia weight, c_1 and c_2 are learning factors, r_1 and r_2 are random numbers in $[0, 1]$, and k denotes the number of iterations.

The value of w will affect the optimization performance of the PSO algorithm. The APSO algorithm makes adaptive improvement on it

$$w = \begin{cases} w_{min} - \dfrac{(w_{max} - w_{min})(f - f_{min})}{f_{avg} - f_{min}} & f \leqslant f_{avg} \\ w_{max} & f > f_{avg} \end{cases} \tag{3.12}$$

where w_{min} and w_{max} represent the maximum and minimum values of w, f is the current particle fitness value, f_{avg} and f_{min} are the average and minimum values of the current particle fitness.

Finally, the flow of the proposed APSO-LSTM-attention method for the ping pong ball trajectory prediction is depicted in Fig. 3.

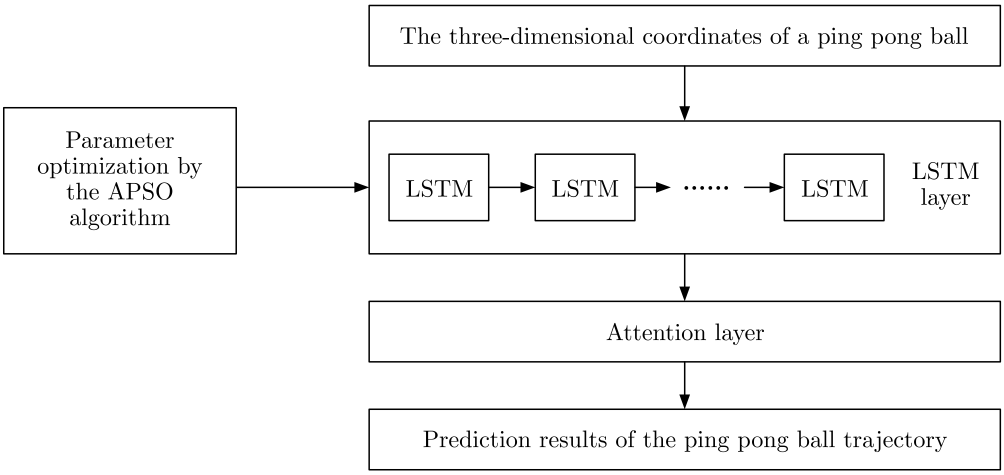
The three-dimensional coordinates of a ping pong ball
Parameter optimization by the APSO algorithm
LSTM
LSTM
······
LSTM
LSTM layer
Attention layer
Prediction results of the ping pong ball trajectory

Fig. 3. The APSO-LSTM-attention trajectory prediction method

As shown in Fig. 3, the parameters of the LSTM are first optimized using the APSO algorithm. The optimized parameters are then input to the LSTM to learn from the three-dimensional coordinates of the ping pong ball. Next, the output of the LSTM serves as the input for the attention layer. By integrating the output of the attention layer, the final prediction results for the ping pong ball trajectory are obtained.

4. Results

4.1. Experimental setup

The experiment was conducted in a Windows 10 operating system with an Intel(R) Core(TM) i7-8550U processor and 8 GB of memory. The Python language was used, and the network model was built on the Kears framework based on TensorFlow. In the APSO-LSTM-attention model, the population size for the APSO algorithm was set to 50, the number of iterations was 500, and $c_1 = c_2 = 1.5$. The optimal values of the LSTM parameters obtained by the APSO algorithm were as follows: learning rate 0.001, number of iterations 200, number of nodes in the hidden layer 64. The three-dimensional coordinates of the i-th ping pong ball trajectory at time t were denoted by $p_t^i(x_t, y_t, z_t)$. A complete trajectory was represented by $(p_1, p_2, \ldots, p_k, \ldots, p_T)$. To predict the three-dimensional coordinates of the ping pong ball at time $k+1$, the trajectory from time 1 to k was used. Then, the trajectory from time 2 to $k+1$ was used to predict the coordinates at time $k+2$. This process was repeated until the entire trajectory was predicted. A total of 1500 trajectories were used in the experiment. The ratio of the training set, the validation set and to and test set was 5:3:2. The data was predicted 100 times for each entry. The final result was obtained by taking the average.

Let $p_t'(x_t', y_t', z_t')$ denote the predicted three-dimensional coordinates of the ping pong ball at time t, and $p_t(x_t, y_t, z_t)$ represents the actual values. Similarly, $p_f(x_f, y_f, z_f)$ represents the predicted three-dimensional coordinates of the endpoint of a trajectory, and $p_f'(x_f', y_f', z_f')$ represents the actual values. The evaluation of trajectory prediction effectiveness was based on the following two indicators:

(1) Average displacement error (ADE), which refers to the error between the predicted result of the three-dimensional coordinates of the ping pong ball and the actual values

$$\mathrm{ADE} = \frac{1}{N} \sum_{t=1}^{k} \sqrt{(x_t' - x_t)^2 + (y_t' - y_t)^2 + (z_t' - z_t)^2} \tag{4.1}$$

(2) Final point displacement error (FDE), which refers to the error between the predicted results of the three-dimensional coordinates of the endpoint of every trajectory and the actual values

$$\mathrm{FDE} = \frac{1}{N} \sum_{t=1}^{k} \sqrt{(x_f' - x_f)^2 + (y_f' - y_f)^2 + (z_f' - z_f)^2} \tag{4.2}$$

4.2. Result analysis

Taking the x-axis coordinate prediction of a trajectory with 20 sample points as an example, the prediction performance of the following methods were compared:

- recurrent neural network (RNN) (Inoue *et al.*, 2018),
- LSTM,
- LSTM-attention,
- APSO-LSTM-attention.

Table 2 presents the percentage error of different methods on the X-axis.

From Table 2, it can be observed that both RNN and LSTM algorithms exhibited relatively large prediction errors on the X-axis, with maximum percentage errors around 3%. In contrast, the LSTM-attention algorithm demonstrated percentage errors below 3% on the X-axis. This confirmed the effectiveness of the attention mechanism. Furthermore, the proposed method achieved a maximum percentage error of only 1.20% and a minimum of 0.03%, showcasing the reliability of optimizing LSTM parameters with the APSO algorithm and its ability to achieve superior results in trajectory prediction.

Taking the 20 sampling points in Table 3 as an example, the results of the proposed method for predicting the three-dimensional coordinates of ping pong ball trajectories are shown in Table 3.

From Table 3, it can be observed that the APSO-LSTM-attention method yielded small errors when compared to the actual values. Among the predictions for the 20 sampling points, the maximum error was found in predicting the X-axis coordinate of sample point 8, with a value of 10.29 mm. The errors for all other sampling points were below 10 mm, which demonstrated the reliability of this method in trajectory prediction.

Table 2. Percentage error of different methods

	Actual value	RNN algorithm	LSTM algorithm	LSTM-attention algorithm	APSO-LSTM--attention method
1	565.38	1.15%	3.08%	1.75%	0.61%
2	662.90	2.85%	3.00%	0.69%	1.13%
3	692.35	3.58%	0.88%	1.84%	0.32%
4	757.02	1.15%	3.54%	1.55%	0.21%
5	810.51	1.11%	−3.18%	0.24%	1.20%
6	826.31	0.57%	1.61%	1.86%	0.86%
7	949.32	−0.16%	1.67%	2.05%	0.43%
8	978.51	3.40%	1.98%	1.98%	1.05%
9	1037.46	3.01%	2.14%	1.97%	0.40%
10	1096.57	2.74%	2.49%	1.25%	0.31%
11	1133.80	1.92%	0.22%	1.38%	0.73%
12	1168.62	−3.38%	1.95%	−0.37%	0.70%
13	1182.20	1.55%	0.81%	0.08%	0.49%
14	1235.44	2.99%	1.42%	0.66%	0.18%
15	1311.63	1.12%	−2.08%	1.35%	0.46%
16	1439.78	−1.24%	1.67%	−1.13%	0.20%
17	1489.86	2.36%	0.90%	0.92%	0.15%
18	1521.97	0.73%	0.76%	1.25%	0.63%
19	1591.20	1.06%	1.55%	0.11%	0.12%
20	1625.60	1.76%	0.48%	0.70%	0.03%

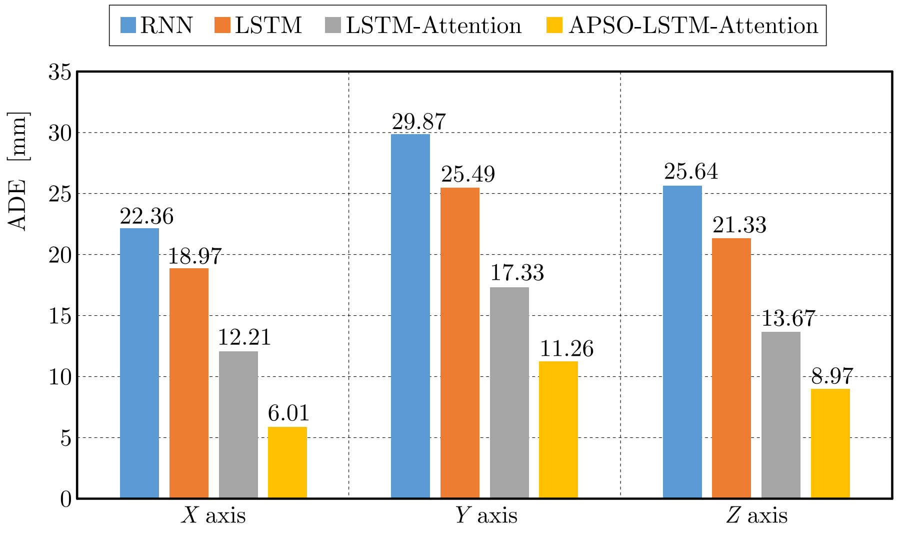

Fig. 4. Comparison of ADE

The comparison results of the ADE among different methods on the test set are presented in Fig. 4.

Firstly, in terms of the prediction in the X-axis, the RNN, LSTM, and LSTM-attention algorithms had ADE values above 10 mm, while the proposed method achieved an ADE of 6.01 mm, reducing the errors by 16.35 mm, 12.96 mm, and 6.2 mm, respectively, compared to the RNN, LSTM, and LSTM-attention algorithms. All methods exhibited high ADE values on the Y-axis. Among them, the RNN algorithm had an ADE of 29.87 mm, while the proposed method showed an ADE of 11.26 mm, significantly lower than the other methods. Finally, in

Table 3. The prediction results of the APSO-LSTM-attention method for three-dimensional coordinates

	Actual value	Prediction result	Error
1	(565.38, 120.33, 321.56)	(568.85, 122.36, 323.55)	(3.47, 2.03, 1.99)
2	(662.90, 116.34, 305.12)	(670.38, 114.25, 303.56)	(7.48, −2.09, −1.56)
3	(692.35, 112.36, 300.12)	(694.58, 114.25, 298.25)	(2.23, 1.89, −1.87)
4	(757.02, 108.26, 297.36)	(758.64, 105.36, 295.62)	(1.62, −2.90, −1.74)
5	(810.51, 106.25, 294.33)	(820.24, 104.22, 291.36)	(9.73, −2.03, −2.94)
6	(826.31, 97.36, 284.26)	(833.39, 100.03, 281.33)	(7.08, 2.67, −2.93)
7	(949.32, 95.36, 281.32)	(953.40, 91.26, 278.65)	(4.07, −4.10, −2.67)
8	(978.51, 93.26, 278.25)	(988.81, 91.26, 275.33)	(10.29, −2.00, −3.79)
9	(1037.46, 89.97, 275.12)	(1041.58, 87.21, 271.33)	(4.12, −2.76, −3.79)
10	(1096.57, 86.25, 271.36)	(1100.00, 84.33, 268.24)	(3.43, −1.92, −3.12)
11	(1133.80, 83.26, 268.45)	(1142.03, 80.26, 264.33)	(8.24, −3.00, −4.12)
12	(1168.62, 81.22, 264.26)	(1176.79, 78.66, 261.36)	(8.17, −2.56, −2.90)
13	(1182.20, 78.64, 261.25)	(1187.98, 75.33, 258.34)	(5.78, −3.31, −5.91)
14	(1235.44, 76.12, 257.26)	(1237.67, 74.21, 255.35)	(2.23, −1.91, −1.91)
15	(1311.63, 74.22, 254.36)	(1317.68, 71.26, 251.33)	(6.05, −2.96, −3.03)
16	(1439.78, 71.15, 251.13)	(1442.68, 68.26, 253.27)	(2.90, −2.89, 2.14)
17	(1489.86, 68.21, 248.61)	(1492.07, 66.33, 245.28)	(2.21, −1.88, −3.33)
18	(1521.97, 65.12, 245.36)	(1531.56, 61.26, 247.36)	(9.59, −3.86, −2.00)
19	(1591.20, 61.42, 241.33)	(1593.18, 59.33, 238.64)	(1.98, −2.09, −2.69)
20	(1625.60, 57.64, 237.52)	(1626.06, 56.97, 235.61)	(0.46, −0.67, 1.91)

the comparison in the Z-axis, the proposed method had an ADE of 8.97 mm, reducing the errors by 16.67 mm, 12.36 mm, and 4.7 mm, respectively, compared to the RNN, LSTM, and LSTM-attention algorithms.

Next, a comparison of the FDE among the different methods is presented in Fig. 5.

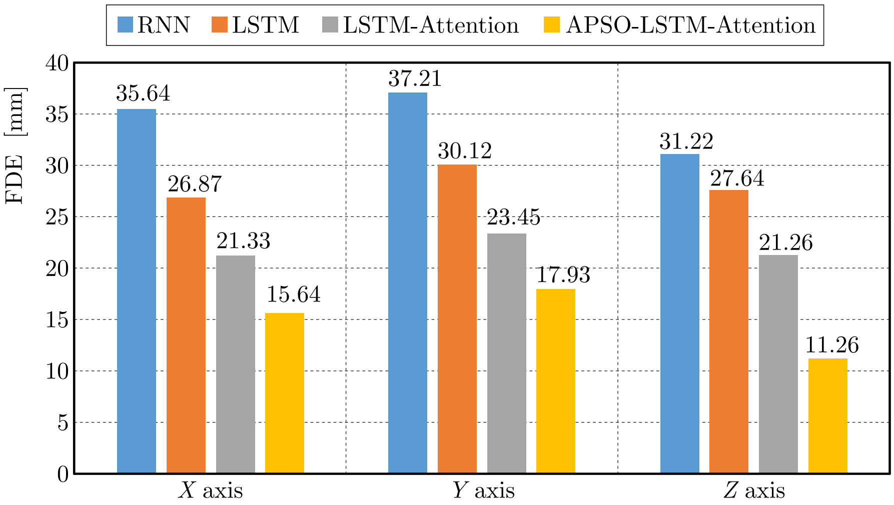

Fig. 5. Comparison of FDE

From Figure 5, it can be seen that the APSO-LSTM-attention method exhibited smaller FDE compared to the RNN, LSTM, and LSTM-attention algorithms. This indicated that the proposed method provided more accurate predictions of the three-dimensional coordinates of the endpoint trajectory of the ping pong ball. This high accuracy is crucial for meeting the precision requirements in practical applications, such as in human-robot ping pong matches, where accurate prediction of the three-dimensional coordinates is essential for higher training efficiency.

Finally, the prediction time between the different methods was compared. The time required for each method to predict a complete trajectory is presented in Fig. 6.

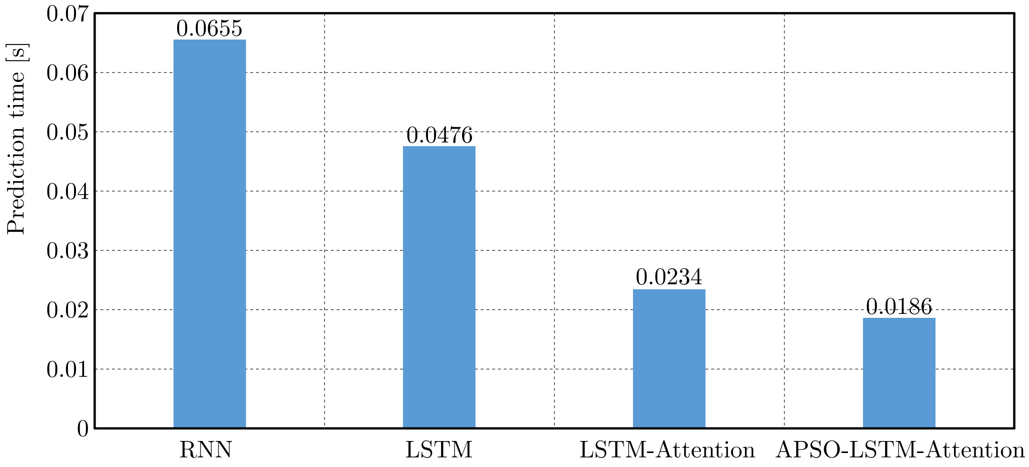

Fig. 6. Comparison of prediction time

From Fig. 6, it can be observed that the prediction time for all the methods was less than 1 s. Comparatively, the RNN algorithm required the longest prediction time, with 0.0655 , while the proposed method had the shortest prediction time of 0.186 s, which was 71.6% less than the RNN algorithm, 60.92% less than the LSTM algorithm, and 20.51% less than the LSTM-attention algorithm. These findings demonstrate that the proposed method does not only exhibit good prediction accuracy but also provides real-time performance.

5. Conclusion

In this study, a trajectory prediction method, the APSO-LSTM-attention algorithm, was designed based on the three-dimensional coordinates of a ping pong ball during rotational flight. The results demonstrated that compared to methods like the RNN and LSTM algorithms, the APSO-LSTM-attention algorithm achieved smaller prediction errors with ADEs of 6.01 mm, 11.26 mm, and 8.97 mm in the X, Y and Z axes, respectively. The FDEs were also smaller, and the time required to predict a complete trajectory was only 0.0186 s, indicating good accuracy and efficiency. These findings support the further application of the proposed method in practical scenarios.

References

1. Amiri A., Salmasnia A., Zarifi M., Maleki M.R., 2023, Adaptive Shewhart control charts under fuzzy parameters with tuned particle swarm optimization algorithm, *Journal of Industrial Integration and Management*, **8**, 2, 241-276

2. Amirloo E., Rasouli A., Lakner P., Rohani M., Luo J., 2022, *Latent Former: Multi-Agent Transformer-Based Interaction Modeling and Trajectory Prediction*, arXiv e-prints

3. Bertugli A., Calderara S., Coscia P., Ballan L., Cucchiara R., 2021, AC-VRNN: Attentive Conditional-VRNN for multi-future trajectory prediction, *Computer Vision and Image Understanding*, **210**, 5, 1-10

4. Bidyanath K., Singh S.D., Adhikari S., 2023, Implementation of genetic and particle swarm optimization algorithm for voltage profile improvement and loss reduction using capacitors in 132 kV Manipur transmission system, *Energy Reports*, **9**, 738-746
5. Chen K., Song X., Ren X., 2021, Modeling social interaction and intention for pedestrian trajectory prediction, *Physica A: Statistical Mechanics and its Applications*, **570**, 9, 1-12.
6. Chen X., Meng X., Zhao Y., 2020, Genetic algorithm to improve Back Propagation Neural Network ship track prediction, *Journal of Physics: Conference Series*, **1650**, 3, 1-9
7. Goldfarb N., Lewis A., Tacescu A., Fischer G.S., 2021, Open source Vicon Toolkit for motion capture and gait analysis, *Computer Methods and Programs in Biomedicine*, **212**, 106414
8. Gomez-Gonzalez S., Nemmour Y., Schölkopf B., Peters J., 2019, Reliable real time ball tracking for robot table tennis, *Robotics*, **8**, 4, 1-13
9. Gruet M.A., Chandorkar M., Sicard A., Camporeale E., 2018, Multiple-hour-ahead forecast of the DST index using a combination of long short-term memory neural network and Gaussian process, *Space Weather*, **16**, 11, 1882-1896
10. Hauri S., Djuric N., Radosavljevic V., Vucetic S., 2021, Multi-modal trajectory prediction of NBA players, *Workshop on Applications of Computer Vision*, 1639-1648
11. Huang M., Ochieng W.Y., Escribano Macias J.J., Ding Y., 2021, Accuracy evaluation of a new generic trajectory prediction model for Unmanned Aerial Vehicles, *Aerospace Science and Technology*, **19**, Dec. Pt. 2, 1-26
12. Inoue M., Inoue S., Nishida T., 2018, Deep recurrent neural network for mobile human activity recognition with high throughput, *Artificial Life and Robotics*, **23**, 2, 173-185
13. Kalatian A., Farooq B., 2022, A context-aware pedestrian trajectory prediction framework for automated vehicles, *Transportation Research Part C: Emerging Technologies*, **134**, 103453
14. Kumar K.S., Gomathi B.R., 2022, Forecasting photovoltaic power output using long short-term memory and neural network models, *International Journal on Engineering Applications*, **10**, 2, 116-125
15. Li S., Wang S., Xie D., 2019, Radar track prediction method based on BP neural network, *The Journal of Engineering*, **21**, 8051-8055
16. Mir M., 2018, *Cruise Missile Target Trajectory Movement Prediction Based on Optimal 3D Kalman Filter with Firefly Algorithm*, arXiv:1807.07006
17. Mirmohammad Y., Khorsandi S., Shahsavari M.N., Yazdankhoo B., Sadeghnejad S., 2021, Ball trajectory prediction for humanoid robots: combination of k-NN regression and autoregression methods, *International RoboCup Symposium*, 1-12
18. Rodrigues T.B., Cathin C.Ó., Devine D., Moran K., O'Connor N.E., Murray N., 2019, An evaluation of a 3D multimodal marker-less motion analysis system, *The 10th ACM Multimedia Systems Conference*, 213-221
19. Rajini Selvaraj A., Gurusamy T., 2023, An optimal model using single-dimensional CAE--IRNN based SPOA for cyclone track prediction, *Expert Systems with Application*, **230**, 1-13
20. Song T., Li Y., Meng F., Xie P., Xu D., 2022, A novel deep learning model by BiGRU with attention mechanism for tropical cyclone track prediction in the Northwest Pacific, *Journal of Applied Meteorology and Climatology*, **61**, 1, 3-12
21. Xi Z., Xu A., Kou Y., Li Z., Yang A., 2021, Target maneuver trajectory prediction based on RBF neural network optimized by hybrid algorithm, *Journal of Systems Engineering and Electronics*, **32**, 2, 498-516
22. Zheng J., Cai F., Shao T., Chen H., 2018, Self-interaction attention mechanism-based text representation for document classification, *Applied Sciences*, **8**, 4, 1-14

Manuscript received August 17, 2023; accepted for print January 3, 2024