Robot Collision Avoidance Technology at Manufacturing Sites that Apply AI Prediction Technology for Walking Trajectories

NING Xiaoguang

In order to enable persons and robots to work in the same shared space at manufacturing sites, robot trajectory planning technologies have been developed for robots to avoid collisions with persons during operation. However, by conventional technologies, since the operation must be temporarily stopped before the robot performs collision avoidance, the robot frequently stops when it comes into close contact with a person, and productivity is significantly reduced.

In this research, as one of the technologies necessary to solve the above problem, we are working on an innovative technology for predicting the position of the human body in a few seconds in advance at the manufacturing site. This time, a novel method has been proposed by applying an AI-based human walking trajectory prediction technology to the prediction of the human body position at the manufacturing site, and performance has been evaluated using one case at the manufacturing site. As a result, the proposed method is able to predict the body position 4 seconds in advance in a processing time of 0.4 ms, and a technical issue has also been identified. In addition, comparative evaluation experiments revealed that this method has better prediction accuracy and a processing speed more than 10 times faster than the orthodox time series analysis method.

1. Introduction

In recent years, the securing of skilled human resources has become a serious issue in the manufacturing industry[1]. In order to resolve this issue, a production method is attracting the attention where a part of the manual work at manufacturing sites is replaced by robots, safety fences used to separate persons and robots are abolished, and persons and robots coexist to perform the work.

In general, there are two means of ensuring safety when introducing industrial robots: separating people from robots using safety fences and sensors, and adopting power and force limiting (PFL) cooperative robots that do not cause injury even if they collide with people[2]. The former has issues of degraded space efficiency and difficulty in flexible layout changes because the fences and safety equipment are used to secure the dedicated workspace for robots. The latter is fence-less and provides coexistence with persons, but the speed and power of the robot are always restricted, and the operation of the robot frequently stops because of collisions with persons during operation, which makes it difficult for industrial robots to achieve their original productivity.

In order to solve the above problems, attention has been focused in recent years on technology to detect human motion using sensors and to allow the robot to plan its motion trajectory in accordance with the human motion so that it does not collide with the person[3,4]. This technology is expected to reduce the frequency of robot stops due to collisions when coexisting with persons in a fence-less environment.

Most research in this technology adopts the methodology of detecting human movement in real time and monitoring the positional relationship of persons and robots based on that information. According to this method, when a person and a robot are approaching closer than the predetermined distance necessary to ensure the safety of the person, in other words, in the case of an over-approach, robot operation is stopped, and a new trajectory is generated to secure the necessary distance, and then robot operation is restarted.

On the other hand, in this methodology, when an over-approach between a person and a robot is detected, the robot must stop the action under operation to prevent a collision

Contact : *NING Xiaoguang* xiaoguang.ning@omron.com

because the new trajectory has not yet been determined. Accordingly, when a robot is working closely with a person in a narrow workspace, the problem is the frequent interruption of the robot's work due to the frequent occurrences of over-approach.

As one of the necessary technologies to solve the above problem, we worked on the technology to predict the movement of a person at the manufacturing site. Through this technology, the acquisition of human location information and the prediction of the positional relationship of persons and robots will be possible in the near future. This makes it possible to complete the generation of a new trajectory before an over-approach actually occurs, and the robot will switch to a new trajectory without stopping. This will mitigate the problem of stopping in the conventional collision avoidance operation and will permit the robot to smoothly avoid the possible collision with a person.

For more information, in this study, the movement of a person working near the robot at a manufacturing site is divided into two steps: (1) the movement of the trunk between work locations, and (2) the movement of the hands when staying at a position and working manually. In the current stage, we are working on the prediction of a person's trunk position as described in (1) above.

The changes in the position of the human trunk due to walking, or walking trajectory, is affected by many factors, such as the environment and individual differences, and is very difficult to derive by theoretical calculations; therefore, in recent years, the development of prediction technology using artificial intelligence (AI) has been progressing.

Conventional technologies have been used for such applications as predicting the walking trajectories of multiple people in a crowded space using image data and have been able to predict the positions of people after a few seconds[5,6]. Being a neural network algorithm that uses a large volume of learned data, the processing time for prediction is several seconds, and the prediction distance error is approximately 1 meter.

Furthermore, these technologies have been applied to predict walking trajectories in open spaces, such as automated driving applications for automobiles, but still not much progress has been made in the research on the application to the closed spaces of manufacturing sites. In addition, there is the issue of incapability to process real time according to the movement of the robot due to the excessive time required by the prediction process.

In order to achieve the aforementioned smooth avoidance of a collision, in this study, the methodology and the effects were examined on the application of conventional AI-based walking trajectory prediction technology to the human body trunk

position prediction at a manufacturing site. In the study, the necessary prediction performance index and the target values were first defined, and in order to achieve it, an approach for applying the conventional technology to the manufacturing site was devised, and a specific application method was proposed based on it. In addition, the effectiveness and issues of the applied method were validated using a case example of a manufacturing site. The details are introduced in this paper.

Since this technology is for the collaborative work of persons and robots, the safety designs will also be an important point, but this paper focuses on AI-based prediction technology, and an explanation of the safety mechanism is omitted. We would like to only mention here that, in practice, safety is achieved by structuring the safety-related part and the AI-based person prediction algorithm in the separate systems.

2. Prediction Performance Index and Target Values

The concept of smoothly avoiding a collision by predicting the position of persons is embodied in Fig. 1 for the purpose of solving the aforementioned problem of frequent interruptions of the robot's work when in close contact with a person. In this flow, the position of a person after a few seconds is predicted (a), and the information is used to calculate the future positional relationship of the person and the robot and to determine whether an over-approach is likely to occur or not (b). The robot is controlled to avoid the collision if YES (c and d).

Since an over-approach must always be watched to ensure the safety of people during operation, (a) and (b) must be continuously executed with short cycles. In addition, to realize the smooth avoidance of a collision, an important point is to complete all processes (a) through (d) before an over-approach actually occurs, in other words, within the prediction time for human body trunk position prediction (a).

Consequently, the performance of (a) human body trunk position prediction in the processing flow in Fig. 1 is important in realizing this flow. As an index for this performance, we specified prediction processing time, predictable time, and prediction distance error and defined the target values for each as follows:

– Prediction Processing Time
The time necessary for one (1) prediction process: It is necessary to reduce this index as short as possible for continuous execution with the aforementioned short cycle. In this study, an OMRON laser scanner (OS32C) was used to acquire the data necessary for prediction with 40 ms cycles (details are explained in paragraph 2.3). In order to perform

the prediction process continuously in conjunction with it, the target processing time for the prediction process was set to 40 ms or less.

– Predictable Time

The time in the immediate future that can be predicted by the prediction process: As mentioned above, the target value of this index should be set up according to the time necessary for processing (a) through (d) in Fig. 1. In this study, the processing times for (a) through (d) are estimated as listed in Table 1, depending on the system configuration. These values may vary depending on the performance of the implemented hardware; however, as a research phase, a strict evaluation was omitted, but stricter setups were chosen. By adding the buffer time for data transmission to the total of these necessary times, we set up the predictable time target as 1.2 s.

– Prediction Distance Error

The distance between the predicted and true values of the human trunk position (Fig. 2): The trunk position is indicated with the position of the center of the trunk. This will be explained in chapter 3. This index indicates the correctness of the prediction results and should be considered as an error by the prediction algorithm when determining the over-approach using the prediction results in (b) of the above flow. In this study, preliminary desk calculations determined what should be considered "close," and the acquired result was that the prediction distance error of 100 mm or less was effective in achieving smooth collision avoidance when a person was working in a cylindrical area with a radius of 2 m centered on the robot's first joint rotation axis. For this reason, the target value was set to 100 mm or less.
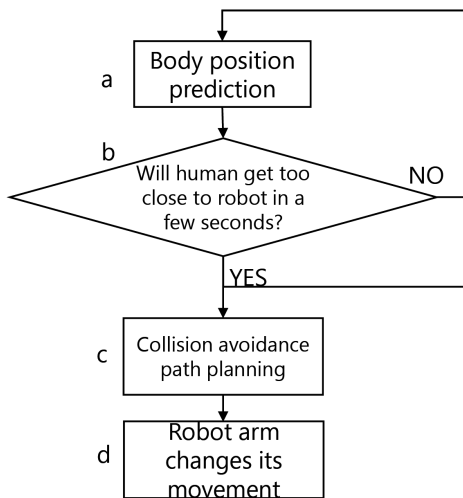
Table 1 Experiment Walking Trajectory Setup

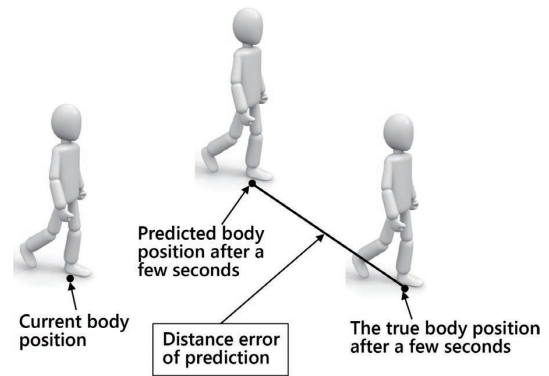| No. in Fig. 1	Processing Time [ms]	How to Determine Processing Time
a	40	Set up with the target value (chapter 2) of the predicted processing time.
b	10	Set up in reference to the actual values when the over-approach detection program was executed on a general-purpose PC.
c	100	Set up in reference to the actual value of the time necessary for the trajectory planning process[3] for a six-joint robot using the probabilistic roadmap method on dedicated processing hardware.
d	800	Set up according to the acceleration/deceleration performance of an OMRON collaborative robot (TM5 series).

Fig. 2 Definition of Prediction Distance Error

Current body position

Predicted body position after a few seconds

Distance error of prediction

The true body position after a few seconds

3. Approach for Manufacturing Site Application

When using AI as the technology for the prediction of walking trajectory, the general method of use is to use personal/environmental factors that affect the change in walking trajectory as a feature quantity and use the learning data that contains it to make the learner learn the laws regarding the change in walking trajectory with regard to the feature quantity. Using an already trained learner, calculate the change in walking trajectory through the analysis of the input data for a prediction containing a similar feature quantity.

The conventional AI-based walking trajectory prediction technologies focused on multiple walking trajectories in a complex environment, and in order to obtain accurate prediction results, considered many feature quantities, such as the location of obstacles and the number of people accompanying the person. The prediction requires several seconds of prediction processing time to analyze a large volume of input data containing the feature quantities. Because of this, there was a large gap compared to the target value of 40 ms for the prediction processing time defined in the previous chapter, and the algorithm could not be directly applied.

To address this problem, we tried to shorten the prediction processing time in this study by minimizing the number of feature quantities.

a Body position prediction

b Will human get too close to robot in a few seconds? NO

YES

c Collision avoidance path planning

d Robot arm changes its movement

Fig. 1 Smooth Collision Avoidance Process Flow

Specifically, since in most cases, the people at the manufacturing site, the target of the prediction in this research, walk on flat ground, and the change in position in the height direction is minimal, the position of the trunk can be expressed by the projection of the trunk center point, that is, the coordinate values (x, y) of the trunk center point (Fig. 3). Therefore, we decided to use the trunk center coordinates (x, y) as the minimum necessary feature quantities for predicting the walking trajectory. Adding the factors, such as the variation in manual work time, to the feature quantities may further improve the prediction accuracy of the walking trajectory, but since it is a trade-off with prediction processing time, it was excluded from consideration this time because the prediction processing time was prioritized. If this feature quantity is needed for further improvement, we will consider it in the next step.
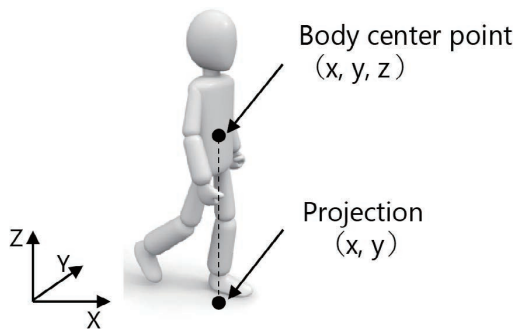

Fig. 3 Body Center Coordinates

4. Prediction Methodology Details

4.1 Overview

Based on the aforementioned approach of minimizing the number of learning feature quantities, we proposed a method to apply AI-based walking trajectory prediction technology to the prediction of human trunk positions at the manufacturing sites. An overview of the method is explained according to the following viewpoints:

– Feature Quantities

As described in chapter 3, x and y of the person's trunk center coordinates are used as the feature quantities.

– Learning Data

The time series data of the trunk center coordinates are used as the learning data. Make the learning machine learn the law for the change in the center of the trunk coordinates with these.

– Learning Machine

A deep binary tree (DBT) learning machine (machine learning package, AISing Ltd.) based on the dynamic system learning tree method[7] was adopted as the learning processor. DBT is good at high speed learning processing with a small

volume of data and featured the ease of use in a built-in system due to its low computational effort. This research as well, since there was not much learning data, finally adopted the DBT, which was easy to integrate into a robot system and provided high-speed processing.

– Prediction Algorithm

By acquiring time series data of the trunk center coordinates of a person that repeats a manufacturing process in real time, and inputting it into a learning machine that has been previously trained with the above learning data, the trunk center coordinate values of the immediate future are acquired as output data. Furthermore, as a characteristic of supervised AI technology, the more valid the input data for prediction, the more accurate the prediction will be; accordingly, in the stage of advanced learner training, tuning of the parameters related to the input data format is performed to improve the prediction distance error.

– System Configuration

As shown in Fig. 4, in this study, three laser scanners (OS32C) are allocated at the same height as the human waist to continuously detect the coordinate values of the body contour point group around the waist with a 40 ms cycle. The detected data is sent every cycle to a prediction processing PC implemented with a prediction algorithm, and the coordinate values x and y of the center point of the point group are calculated and are used as the body center coordinate values at the time when the point group is detected. The time-series data of trunk center coordinate values acquired through this method are stored in the learning/prediction data storage section and analyzed in the processing section.
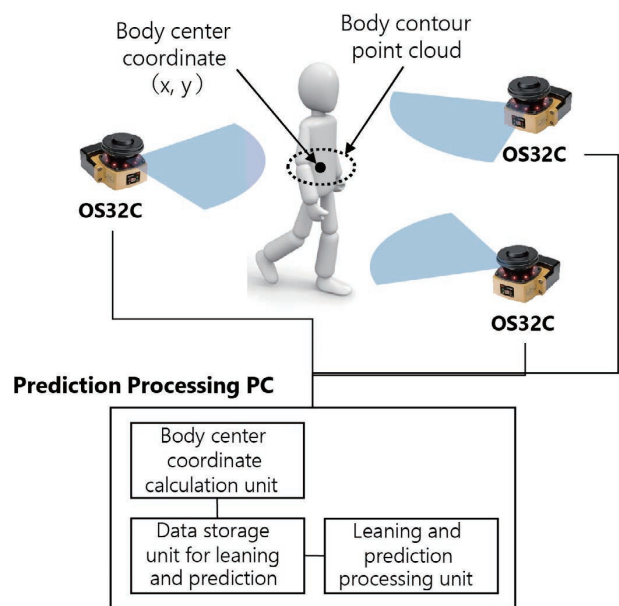

Fig. 4 Trunk Center Position Prediction System Configuration

4.2 Detailed Algorithm

The detailed algorithm of the application method proposed in this study is explained using a processing flow. As shown in Fig. 5, the processing flow of this method consists of three sections, A through C.

Section A (Human Body Trunk Center Coordinate Detection)

This section acquires and stores the trunk center coordinate values (x, y) as described in the system configuration above (Section 4.1).

Section B (Training of Learning Machine)

This chapter makes the DBT learning machine learn the law of trunk center coordinate fluctuations, in other words, the relationship between past trunk positions and future trunk positions, based on the human trunk center coordinate data accumulated in section A. It also tunes the parameters related to the input data format for acquiring more accurate prediction results.

The processing performed in this section is explained using Fig. 6. The symbols in the figure for the trunk center coordinates (x, y) indicate the time at which the coordinate values were obtained. For example, (x_n, y_n) represents the trunk center coordinate values obtained at time n.

First, let the learning machine learn the relationship between past trunk center positions and future trunk center positions. Specifically, as shown in B-1 in Fig. 6, a part of the data accumulated in section A is acquired as the data for training. Every time data is acquired (time p in the example in the figure), use the previously acquired data as the training input data and extract the later acquired data as the training output data to create a data pair. And, let the learner learn these training data pairs. Here, the input data format is determined by the volume of data (k) and the time interval between data (a) and tuned as described later. The volume of output data (l) and the time interval (b) are not tuned but are set up according to the prediction requirements.

Next, as shown in part B-2 in Fig. 6, a part of the data accumulated in section A is acquired as the data for error evaluation, and the prediction distance error for the trunk position prediction by the learning machine already trained in B-1 is evaluated. The evaluation method will be explained in paragraph 5.2.

Furthermore, performing a grid search within a certain range for the input data format parameters "k" and "a", and repeating the learning (B-1) and the evaluation (B-2) in Fig. 6, we find the values of "k" and "a" that give the minimum prediction distance errors.

Section C (Human Trunk Center Coordinate Prediction)

This section predicts the human trunk center coordinates in the immediate future using a trained learning machine obtained in section B and the values "k" and "a".

The details of processing are explained using Fig. 7 as follows:

Acquire the time series data of the trunk center coordinates from section A until the current time (time "t" in the example in the figure) and create the input data for prediction using the values of "k" and "a" determined in section B.

Input this into the already trained learning machine and acquire the trunk center coordinates in the immediate future (of $l \times b$ seconds in the example in the figure).

If the robot position information in the immediate future is added using this prediction result, the positional relationship between a person and the robot in the immediate future is estimated in block "b" of the aforementioned avoidance process flow (Fig. 1), and it becomes possible to determine whether an over-approach will occur or not.
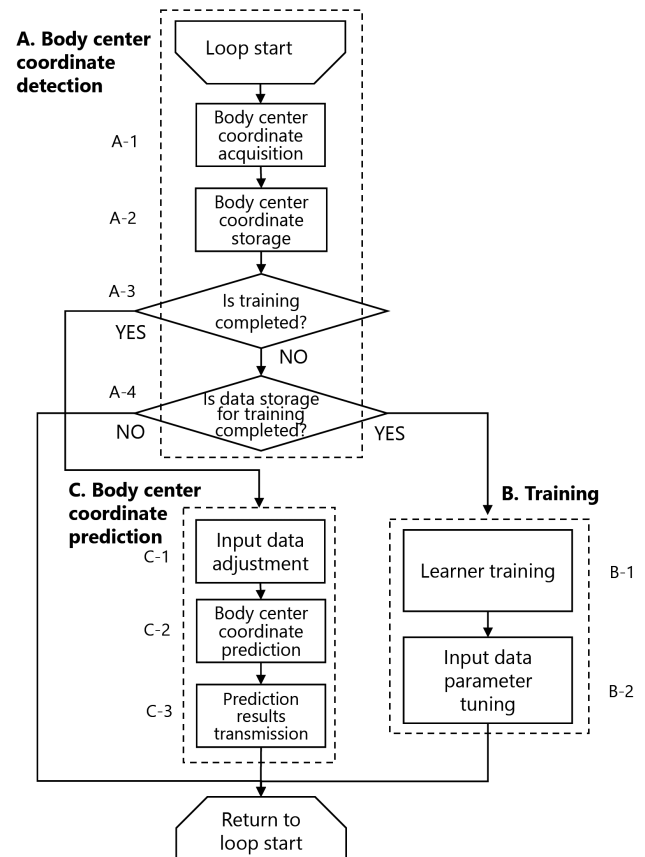

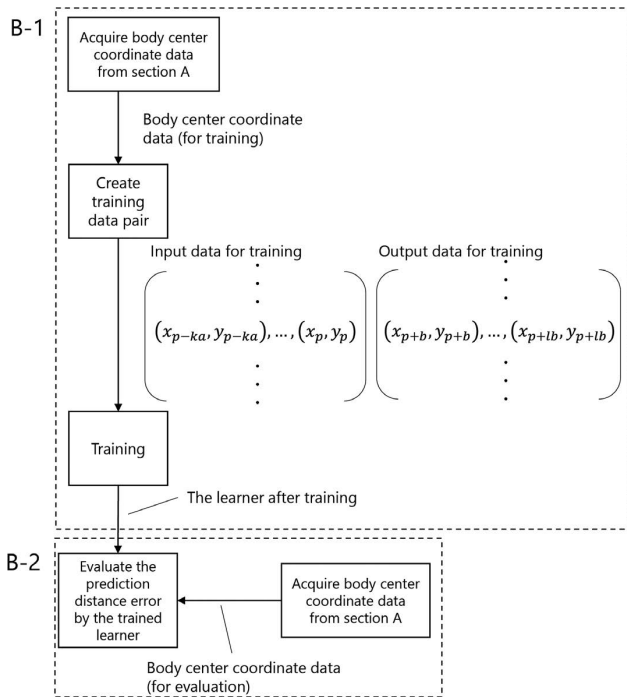**Fig. 5** Prediction Technology Application Method Processing Flow

B-1

Acquire body center coordinate data from section A

Body center coordinate data (for training)

Create training data pair

Input data for training　　Output data for training

$$(x_{p-ka}, y_{p-ka}), \ldots, (x_p, y_p) \quad (x_{p+b}, y_{p+b}), \ldots, (x_{p+lb}, y_{p+lb})$$

Training

The learner after training

B-2

Evaluate the prediction distance error by the trained learner

Acquire body center coordinate data from section A

Body center coordinate data (for evaluation)

Fig. 6 Section B Processing Details

C-1

Acquire body center coordinate data from section A

Body center coordinate data (for prediction)

Create input data for prediction using k and a

Input data
$(x_{t-ka}, y_{t-ka}), \ldots, (x_t, y_t)$

C-2　Input the input data into learner and get prediction results

Prediction results of body center coordinate
$(x_{t+b}, y_{t+b}), \ldots, (x_{t+lb}, y_{t+lb})$
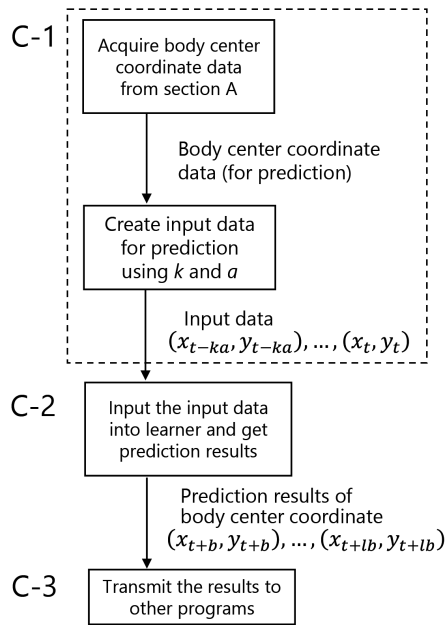
C-3　Transmit the results to other programs

Fig. 7 Section C Prediction Processing Details

5. Performance Evaluation

In this chapter, the results of evaluating the application methodology for the aforementioned AI-based walking trajectory prediction technology in terms of predictable time, prediction processing time, and prediction distance error in an experimental environment that simulates a U-shaped manufacturing line as one of the case examples of a manufacturing site. Paragraph 5.1 describes the conditions of

the experiment, paragraph 5.2 describes the evaluation method, and paragraph 5.3 describes the results and discussions on the experiment.

5.1 Experimental Conditions
Experimental Environment Setup

The layout of an experimental environment was designed in reference to the U-shaped manufacturing line for assembling small electronic components in our factory. Fig. 8 shows a bird's-eye view of this layout. As shown in Fig. 8, a person walks and moves for working in the dot pattern area, which is surrounded by the workbenches installed in U-shape. For covering the entire walking area, three laser scanners for trunk contour detection are installed. For the coordinate systems of the human trunk center, the layout direction of the workbenches (1) through (3) was set as the X-axis and its vertical direction as the Y-axis.
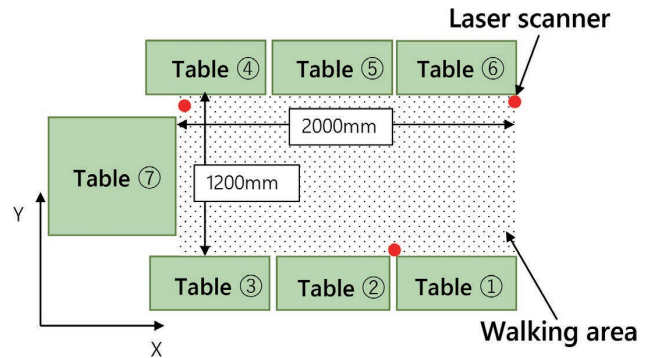

Fig. 8 Experimental Environment Layout Bird's Eye View

Experimental Walking Trajectory

Based on the walking trajectories of a worker engaged in the assembly work for small electronic components in a U-shaped manufacturing line of our plant, the following four experimental walking trajectory configuration indices were extracted. In addition, based on the results of a series of surveys on multiple working processes, the respective typical values were determined.

– Trajectory Shape

Shape of the walking trajectory: In this experiment, we defined three shapes: a straight line, Z-shape, and aisle intersection (Figs. 9 to 11). In the figure, the arrow lines indicate the movement of a person; a round point is the start point, and the arrow is the end point. Each number of the lines indicates the order of the movement.

– Speed

Walking speed when a person moves: In this experiment, the

setups were defined in two stages: low speed (about 1,000 mm/s) and high speed (1,600 mm/s). The value specified in the person-robot cooperative safety standard (ISO 10218−1/ TS 15066) was adopted as the speed for the high speed.

− Moving Distance

Distance of the movement: The distance is the length of the arrow lines in Fig. 9 through Fig. 11. In this experiment, two different setups, 600 mm and 1,200 mm, were adopted.

− Stop Time

The period when a person stops at a workbench for work. In other words, the movement stop time between the arrow lines, which indicate movement, in Fig. 9 through Fig.11. In this experiment, the setup is in two values of 2 s and 10 s.

With the combination of the above indices, six different walking trajectories were set up (Table 2). The pattern most frequently seen in the field, #1, was defined as the standard pattern, and #2 through #6 were set up with only one of the indices changed based on #1. It was considered that this setting would provide the evaluation of the relationship of individual trajectory indices and the prediction results.
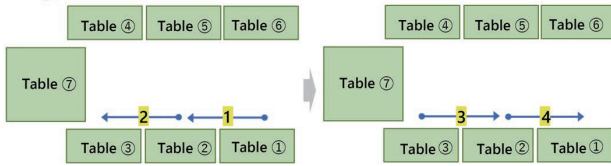
Straight Line

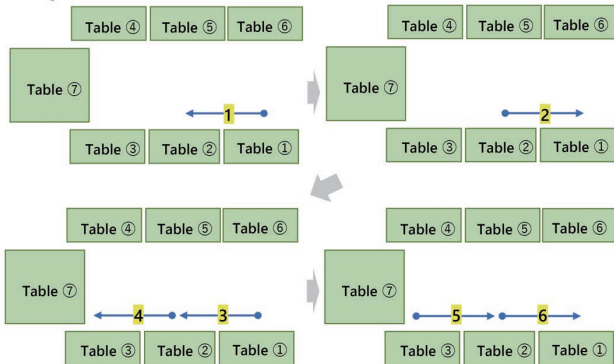**Fig. 9** Straight Line Trajectory Shape

Z shape

Fig. 10 Z-Shape Trajectory
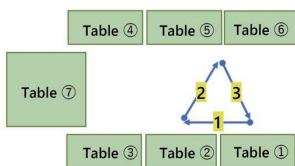
Aisle Intersection

Fig. 11 Aisle Intersection Trajectory Shape

Table 2 Experimental Walking Trajectory Setups

#	Trajectory Shape	Speed [mm/s]	Moving Distance [mm]	Stop Time [s]	Volume of Learning Data	Volume of Evaluation Data
1	Straight Line	about 1000	600	2	9102	4551
2	Z-Shape	about 1000	600	2	8663	4332
3	Aisle Intersection	about 1000	600	2	9207	4603
4	Straight Line	about 1600	600	2	6332	3166
5	Straight Line	about 1000	1200	2	4477	2239
6	Straight Line	about 1000	600	10	9637	4818

Parameter Setups related with Prediction Processing

− Tuning range of parameters (k and a) for the input data format
In this experiment, "k" was tuned in the range of [1, 100] with the increment of 1, and "a" was tuned in the range of [0.04 s, 4 s] with the increment of 0.04 s.

− Setup of the output data format (l and b)
In this experiment, the predicted output data (l) was set to three patterns of 30, 50, and 100, and the between data time interval (b) was set to 0.04 s. According to the combination of l and b, the predictable time ($l \times b$) was set to 1.2 s, 2 s, and 4 s

5.2 Evaluation Method

In the experimental environment described above, a single worker repeats the movement along each aforementioned experimental walking trajectory 30 times respectively. During that time, the trunk center coordinate value data of the worker is acquired by the aforementioned prediction system (Fig. 4), and the learning machine is trained using the data of 20 times out of 30 times as the data for learning.

Using the remaining 10 datasets as the data for evaluation, the trunk center coordinate values of 1.2 s, 2 s, and 4 s ahead for each trajectory are predicted, and the prediction performance indices were evaluated using the following method. The data for training and evaluation for each trajectory is shown in Table 2.

− Prediction Processing Time Evaluation Method
When conducting an experiment, the time required for the prediction processing section of this technology (section C in Fig. 5) is measured per loop. The maximum value of this time is taken as the prediction processing time.

− Prediction Distance Error Evaluation Method
For each trajectory, the relative distance between the measured and predicted coordinate values at each time is calculated every time the trunk center coordinates are acquired, and the maximum value is taken as the prediction distance error. In this experiment, the mean $+3\sigma$ of the distance distribution is taken as the maximum value. Since

Table 3 DBT Method Evaluation Result

Prediction Time [s]	Trajectory #1		Trajectory #2		Trajectory #3		Trajectory #4		Trajectory #5		Trajectory #6	
	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]
1.2	132	0.1	304	0.2	131	0.2	150	0.2	110	0.1	279	0.2
2	130	0.2	416	0.3	113	0.3	153	0.2	112	0.1	303	0.1
4	151	0.3	544	0.2	123	0.2	184	0.2	109	0.2	345	0.4

the influence of the variation in laser scanner measuring distance is large, and the sensor measurement variation is generally evaluated using 3σ, the predicted distance error is evaluated using 3σ.

5.3 Experimental Results

The results of the evaluation experiments conducted in accordance with the methods described in Paragraphs 3.1 and 3.2 are summarized in Table 3.

As for the predicted distance error, the longer the prediction time, the greater the error becomes in many cases, and the maximum error in the case of a prediction of 1.2 s ahead is 304 mm, which deviates from the target value of this study. The maximum error for predicting 4 s ahead was 544 mm, and its occurrence in trajectory #2 was confirmed.

For the predicted processing time, it was confirmed that the processing time for all experimental cases was 1 ms or less with a maximum value of 0.4 ms. The target of 40 ms was cleared, and it was found that the prediction system configuration of this study could perform continuous prediction processing with a 40 ms cycle.

For future improvement, the influence factors on the errors were investigated using the experimental data. Table 4 shows the number of stops per cycle and the total stop time per trajectory. As a trajectory that shows relatively large errors, trajectory #2 was found to have a large number of stops per cycle, and trajectory #6 had a long stop time per cycle. This indicated that the number of stops and the stop time per cycle of

the trajectory to be predicted may influence the prediction distance error.

Table 4 Number of Stops and Stop Time per Cycle of Waking Trajectory

Trajectory #	Number of Stops	Total Stop Time [s]
1	4	8
2	6	12
3	3	6
4	4	8
5	2	4
6	4	40

5.4 Comparison with Conventional Methods

For a comparison with the prediction results by the prediction method in this study using DBT, the prediction and evaluation methods using an orthodox time series analysis method in accordance with the evaluation method in 5.2 were conducted, using the time series data of the trunk center coordinates of six trajectories acquired under the experimental conditions in 5.1.

As an orthodox method, the ARIMA model and LSTM were employed.

The ARIMA model is an abbreviation for auto-regressive integrated moving average model, which is a typical method used in statistics for time series analyses[8]. In this study, we used Python's Darts library. Because a univariate analysis was possible, we prepared separate learning machines for trunk center coordinates X and Y to carry out the prediction.

LSTM is the abbreviation for long short-term memory, which is a kind of RNN and is characterized by its ability to deal with

Table 5 Evaluation Result of Three Prediction Methods

Prediction Time [s]	Learning Machine	Trajectory #1		Trajectory #2		Trajectory #3		Trajectory #4		Trajectory #5		Trajectory #6	
		Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]	Distance Error [mm]	Processing Time [ms]
1.2	ARIMA	620	54.1	612	41.3	296	42.7	232	30	178	51.2	432	40.3
	LSTM	135	9.3	309	8.9	131	8.9	253	7.9	126	7.9	278	7.9
	DBT	132	0.1	304	0.2	131	0.2	150	0.2	110	0.1	279	0.2
2	ARIMA	781	62	787	42.8	355	100.2	219	25	218	51.9	595	52.0
	LSTM	137	13	436	12.8	99	12.8	278	11.3	128	11.3	312	11.2
	DBT	130	0.2	416	0.3	113	0.3	153	0.2	112	0.1	303	0.1
4	ARIMA	901	48.5	900	42.7	366	43.7	224	40.2	148	42.9	770	32.8
	LSTM	145	22.9	589	22.8	104	22.9	350	20.3	118	20.2	369	20.2
	DBT	151	0.3	544	0.2	123	0.2	184	0.2	109	0.2	345	0.4

longer-term series[9]. This time, we used Python's Darts library. Since multivariate prediction is possible, the trunk center coordinates X and Y were predicted simultaneously by a single learning machine similarly as DBT.

The evaluation results for the prediction distance error and processing time of the six trajectories using the three methods (ARIMA model, LSTM, and DBT) are shown in Table 5.

It was confirmed that the distance error was almost DBT < LSTM < ARIMA. In trajectory #4, the distance error of DBT was prominently smaller than LSTM and similar or slightly smaller than LSTM in other trajectories.

In terms of processing time, DBT < LSTM < ARIMA was confirmed for all trajectories, and the speed of prediction processing of DBT was found to be more than 10 times faster than LSTM.

6. Conclusions

In this study, in order to achieve smooth person-robot collision avoidance, we defined the performance indices and target values of the human trunk position prediction technology and proposed a method to apply the AI-based walking trajectory prediction technology to human trunk position prediction in manufacturing sites based on the use of trunk center coordinates as the feature quantity.

The effectiveness of the method was evaluated in a simulated environment of a U-shaped manufacturing line. As a result, it was possible to predict the position of the human trunk 4 s ahead with a processing time of 0.4 ms, and the maximum prediction distance error was 544 mm. In addition, comparative evaluation experiments have shown that our method has better prediction accuracy than orthodox time series analysis methods, and the processing speed was more than 10 times faster.

Regarding the prediction distance error by this method, it was found that it could not achieve the goal and needed to be improved. This was due to the frequent stops and long stop time of the trajectory to be predicted and made it difficult to predict the next action. As a countermeasure, we are considering the possibility of improving the AI judgment precision and the prediction distance error by adding the information that allows predicting the next action to the prediction process, such as tracking the visual line of workers or monitoring the state of manual work when stopped.

In the future, we will improve the practicality and versatility of this method by improving prediction distance error and evaluating the increased number of case examples of manufacturing sites. In addition, we will continue our research aiming at the acquisition of technologies for handling multiple people and predicting hand motions, and eventually realize smooth person-robot collision avoidance using human motion prediction technology, thereby contributing to the practical application of a person-robot collaborative production environment where safety and productivity are highly compatible.

References

1) Ministry of Economy, Trade and Industry, *Monozukuri White Paper 2019*. Research Institute of Economy, Trade and Industry, 2019, pp. 195–216.

2) Japan Industrial Safety and Health Association. *Functional Safety Utilization Practical Manual / Industrial Robot Systems Edition*, Japan Industrial Safety and Health Association, 2017, pp. 8–42.

3) S. Murray, W. Floyd-Jones, and Y. Qi, "The Microarchitecture of a Real-Time Robot Motion Planning Accelerator," in *2016 49th Ann. IEEE/ACM Int. Symp. Microarchitecture (MICRO)*, 2016, pp. 1–12.

4) H. Schumann-Olsen, M. Bakken, and Ø. H. Holhjem, "Parallel Dynamic Roadmaps for Real-Time Motion Planning in Complex Dynamic Scenes," in *3rd Workshop on Robots in Clutter*, IEEE, 2014

5) S. Pellegrini, A. Ess, K. Schindler, and L. Van Gool, "You'll never walk alone: Modeling social behavior for multi-target tracking," in *2009 IEEE 12th Int. Conf. Computer Vision*, 2009, pp. 261–268.

6) A. Alahi, K. Goel, and V. Ramanathan, "Social LSTM: Human Trajectory Prediction in Crowded Spaces," in *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016, pp. 961–971.

7) C. H. Kim, "Machine Property Learning using a Forging Model," in *23rd Robotics Symposia Preliminary Drafts*, 2018, 4C4.

8) G. Box and G. Jenkins, *Time Series Analysis: Forecasting and Control*. San Francisco, CA, USA: Holden-Day, 1970, pp. 88–97.

9) S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.

About the Authors

NING Xiaoguang Ph.D. *(Engineering)*

Technology Dept. 1

Technology Development Division H.Q.

Industrial Automation Company

Specialty: Mechanical Engineering