IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received October 25, 2019, accepted November 9, 2019, date of publication November 13, 2019, date of current version November 25, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2953282

Deep Belief Network–Based Learning Algorithm for Humanoid Robot in a Pitching Game

TZUU-HSENG S. LI[1], (Member, IEEE), PING-HUAN KUO[2], CHIEN-YU CHANG[1], HAO-PING HSU[1], YUAN-CHIH CHEN[1], AND CHIEN-HSIN CHANG[1]
[1]aiRobots Laboratory, Department of Electrical Engineering, National Cheng Kung University, Tainan 701, Taiwan
[2]Computer and Intelligent Robot Program for Bachelor Degree, National Pingtung University, Pingtung 90004, Taiwan

Corresponding author: Ping-Huan Kuo (phkuo@mail.nptu.edu.tw)

This work was supported in part by the Ministry of Science and Technology (MOST), Taiwan, under Grant MOST 106-2218-E-153-001-MY3 and Grant MOST 106-2221-E-006-009-MY3.

ABSTRACT A cognition learning algorithm based on a deep belief network and inertia weight Particle Swarm Optimization (PSO) is presented and examined in a humanoid robot. The psychology concepts were adopted from Thinking, Fast and Slow by Daniel Kahneman. The human brain comprises two systems, System 1 and System 2. Based on their characteristics, System 1 and System 2 handle different tasks during cerebration. In this study, Deep Belief Network (DBN) is trained to construct the function of System 1 for the rapid reaction. On the other hand, PSO is applied to build System 2 for the slow and complicated brain behavior. Through the cooperation of System 1 and System 2, the proposed cognition learning algorithm can apply the psychology theories to allow the humanoid robot for learning the suitable pitching postures autonomously. In the experiments conducted in this study, the robot was trained for only five selected points and was then asked to throw precisely to nine points. The proposed algorithm provided 100% accuracy in the robot pitching game. The feasibility of the proposed algorithm was thus verified.

INDEX TERMS Deep belief network, humanoid robot, machine learning, particle swarm optimization.

I. INTRODUCTION

There are several types of robots, such as humanoid robots [1], [2], mobile robots [3], and swarm robots [4]. They use biological mechanisms, perception, and artificial intelligence to emulate the behaviors, thoughts, feelings, and actions of human beings or other organisms. Because of the rapid development of computer technology and artificial intelligence technology, robots have been considerably improved at the functional and technical level. These improvements have allowed the realization of mobile robots, robot vision, and speech recognition technology. Developments in these technologies have allowed us to use these concepts for robots. These concepts have not only guided the research and application of robotic technology but also provided a great opportunity for robotic technology to develop on a wide scale; thus, many dreams have become a reality.

The associate editor coordinating the review of this manuscript and approving it for publication was Anandakumar Haldorai.

In the previous research, a basic cognition learning algorithm framework for a humanoid robot [5] is also proposed. Advantages of the rapid progression of the processing ability of computers and the reduction in the size and weight of hardware devices allowed us to design a teen-sized humanoid robot that has a more multifunctional body and has a more intelligent brain compared with previous robots.

In the field of computer science, optimization and problem-solving can be achieved using many algorithms. These algorithms include the latest algorithms such as the grey wolf optimizer [6] and the whale optimization algorithm [7] as well as the most widely known algorithms such as ant colony optimization [8], genetic algorithms [9], PSO [10], artificial neural networks [11], and artificial bee colonies [12]. These algorithms are used in the fields of navigation [13] and scheduling [14] among other applications. Interestingly, many of these intelligent learning algorithms have been developed by analyzing biological effects related to the distinctive behaviors of animals. Different algorithms have their own advantages and disadvantages because of their different characteristics. Therefore, in this study, the concept of cognitive psychology used was derived from the concept of the book *Thinking, Fast and Slow* by Daniel Kahneman, the 2002 winner of the Nobel Memorial Prize in Economic Science [15], to develop a novel learning method known as a cognition learning algorithm. The aim of this study was to design and apply a cognition learning algorithm that allows a humanoid robot to simulate human cerebration behavior for learning the postures of pitching a baseball.

Scholars [16] reported a robot that applied a Petri Net-Based Wireless Sensor Node Architecture (PN-WSNA) model; that robot was able to grab a ball from the ground and then throw it to a basket, but that paper did not report much research on optimization. The performance of that PN-WSNA robot was 70%-80% for 50 continuous throws. Researchers [17] applied a visual feedback system to a robot, which enabled it to achieve an accuracy of 99% in a throwing task. The problem was that the basket center was defined by the square on the board, but the relationship between basket and board tends to change when different baskets are used. Our work presents a method that can complete the task and still deliver high accuracy by applying the cognitive styles of System 1 and System 2; therefore, our work outperforms competitors in the field of robot pitching.

In the reinforcement learning (RL) field, a prioritized replay can be an example of System 2. Cases with high reward are highly likely to be sampled. In aspect of the proposed cognition learning algorithm, the highly reasonable pitching motions, which are determined by System 2, will be prioritized for the training basis of System 1. Moreover, several researches that incorporate intrinsic reward or explicit model can also work as System 2. Some recent studies in RL might be adopted to build System 2 agents that guide System 1 agents. For the deep curriculum reinforcement learning (DCRL) paradigm proposed in [18], the researchers controlled the complexity of transitions by manipulating two criteria, namely self-paced priority and coverage penalty. In the same study, the researchers compared DCRL to a Deep Q Network (DQN) and Prioritized Experience Replay (PER) on the Arcade Learning Environment (ALE) platform, and the study results indicated that DCRL outperformed the DQN and PER. At the same time, the application of the proposed DCRL training paradigm in a double DQN and dual networks can result in even higher performance. In [19], an off-policy adaptive Q-learning algorithm was proposed. Specifically, an off-policy learning method was used for simultaneously introducing the Q-function and learning the optimal Q-function. For the algorithm used in [19], an adaptive parameter was used to achieve tradeoff between the current and future Q-functions. The employment of this type of experience replay in the learning process facilitates the implementation of the adaptive Q-Learning algorithm. In [20], the researchers introduced the reinforcement learning algorithm to solve the problem of controlling traffic signals control at multiple intersections. More importantly, the algorithm proposed in [20] can be expanded to more intersections without being restricted by the problem of dimensionality or intersection structures. The reinforcement learning algorithms used in [18]–[20] can all be applied in the proposed System 2 code; they each have their respective advantages in terms of various performance indicators and actual applications. The algorithm proposed in the current study was inspired by the reaction and thinking modes of the human mind. When we decided the algorithm should be used for System 2 thinking, we believed that PSO approximates the logic of the human mind more closely than other algorithms do. As an algorithm that is used extensively, its achievement in various performance indicators is unquestionable. Additionally, the characteristics (i.e., the ability to simultaneously handle gradually converging large-scale searches by particle swarms with high particle counts) of PSO make it similar to the decision-making processes of human beings (i.e., reaching a decision through reflection and comparison of relevant experiences). Therefore, the PSO algorithm presented in the current paper was inspired by the logic of human thinking.

The major contributions of this study are as follows. 1) Psychology theories were applied, 2) DBN model and PSO method were combined to construct the proposed cognitive learning algorithm, 3) a humanoid robot was designed for the experiment, and 4) the feasibility of the proposed algorithm was demonstrated in a robot pitching game.

This paper is organized as follows. In Section II, cognitive psychology is introduced. The proposed cognition learning algorithm is introduced in Section III. In Section IV, the experimental results are presented to verify the feasibility of the proposed algorithm. Some corresponding topics are discussed in Section V. Finally, the conclusions are presented in Section VI.

II. COGNITIVE PSYCHOLOGY

Cognitive psychology is the study of mental processes involved in attention, memory, perception, language processing, problem-solving, learning, and cerebration [21]. The primary focus of cognitive psychology is on mental actions or processes by which knowledge is acquired. In other words, cognitive psychology deals mainly with the processes that mediate between the input of a stimulus and the output of a response. Some theories in cognitive science coupled with evidence from neuropsychology and computational modeling are the main foundations of cognitive neuroscience [22], [23]. Cognitive science addresses many questions of how cognitive functions are produced by neural circuits in the brain. For example, some advanced brain imaging systems are used to obtain deeper knowledge of the division of labor within the brain. The study of cognitive neuroscience can help people to obtain further information on why others have difficulty in demonstrating empathy. Computers have allowed psychologists to try to understand the complexity of human cognition.

From this viewpoint, in psychology, cerebration behavior can be divided into two modes—fast and slow. Psychologists have been intensely interested in these modes for several decades. Daniel Kahneman, the author of *Thinking, Fast and*

Slow, described mental life using a metaphor of two agents, System 1 and System 2, that play distinct roles in the human brain [21]. If we consider these two modes of cerebration as two characters in a psychodrama, the two agents have individual abilities, limitations, and functions. They use their dissimilar properties to deal with miscellaneous problems. Then, they give our minds an intuition that dominates our cerebration behaviors.

The System 1 and the System 2 operate continuously in our brains, and we construct a coherent interpretation of what is going on at any instant. Let us consider students as an example. To be a good student, we must go to class, sit on our seats, and listen to the teacher with undivided attention. In an actual situation, we might concentrate on a lecture from the beginning. System 2 causes us to pay attention in class. However, after a while, we might become distracted from the lesson. We might free our thoughts and emotions unconsciously and just fix our eyes on the blackboard. In this case, System 1 overtakes System 2. A question that arises is how can System 1 be in charge of our mind alternately? The primary reason is that System 2 is slow. System 1 and System 2 are both active when we are conscious. System 1 runs automatically, whereas System 2 operates in a comfortable and effortless manner in which only a fraction of its capacity is engaged. The highly diverse operations of System 2 all require attention and will be ignored only when the attention is disrupted. In contrast to System 1, System 2 is more diligent in its work. System 1 continuously gives System 2 suggestions, including impressions, intuitions, intentions, and feelings. When all operations run smoothly, System 2 adopts the suggestions of System 1 with little or no modification. Thus, if we become inattentive in class, we might be easily led by System 1. Thus, our teachers and parents ask us to pay attention to the teacher in the class as children. If we concentrate and pay undivided attention in class, we can obtain wisdom from our teachers and absorb new knowledge.

In general, System 1 has the characteristic of intuitive cerebration, and System 2 has the characteristic of peculiar logical cerebration. At first, System 2 seems to be much more reliable than System 1. However, human beings almost entirely use System 1 while cerebration in their daily lives. For instance, normally, both System 1 and System 2 operate when a person goes to a famous dessert shop. If the person has a sweet tooth, the person will be attracted by the cake cabinet immediately after entering the store. This action is generated by System 1, which tells the person of the love for those desserts and suggest trying them. While the person eats the dessert, some of the following thoughts usually come to mind. Does it have too many calories? Is the price of the dessert too high? Does the dessert really taste good? Some of these questions were generated by System 2 to review the behaviors generated by System 1. System 2 has some ability to change the ideas produced from System 1 by continuously monitoring our own behaviors. In this example, one of the tasks of System 2 was to overcome the impulses or weaknesses of System 1. Thus, System 2 is in charge of self-control.

III. PROPOSED COGNITION LEARNING ALGORITHM

Deep learning is a type of machine learning that constructs a model of multiple layers with a complex structure or multiple nonlinear transformations. One of the advantages of deep learning is that it recognizes and extracts hierarchical features using efficient algorithms for semisupervised or unsupervised learning. This advantage of deep learning is beneficial because most of the training data are unlabeled. However, deep learning allows computers to learn to represent high-level concepts with low-level data. Deep learning can be regarded as a method of expressing information, and an artificial neural network is the most suitable technology to achieve this type of expression. There are various artificial neural networks of deep learning architectures such as recurrent neural networks (RNNs) [24], DBNs [25], and convolutional neural networks (CNNs) [26]. These deep learning algorithms have been applied to a wide variety of fields and have exhibited high performance in applications. Deep learning architectures involving artificial neural networks have become the subject of a number of recent studies.

A. DBNs

A DBN [25] is a type of a deep neural network that is stacked with a multilayer structure of restricted Boltzmann machines (RBMs). The structure of an RBM includes a visible layer and a hidden layer that have visible neurons and hidden neurons, respectively. Visible neurons fully connect with hidden neurons, and there is no intralayer within the visible or hidden layer. There are two main learning methods in a DBN—unsupervised and supervised learning. An RBM performs unsupervised learning, whereas a back-propagation network (BPN) performs supervised learning. After conducting RBM operations, hidden neurons are conditionally independent when the visible states are given. Thus, we can quickly take an unbiased sample from the posterior distribution when an input vector is given. An RBM is a probabilistic model. Two main functions can be used to represent this model—the energy function and probability distribution function. The weights and biases of an RBM determine the energy of a joint configuration of visible and hidden neurons.

$$E(v, h; \theta) = -\sum_{i=1}^{n}\sum_{j=1}^{m} v_i h_j w_{ij} - \sum_{i=1}^{n} v_i a_i - \sum_{j=1}^{m} h_j b_j \quad (1)$$

where E represents energy with configuration v on the visible neurons and h on the hidden neurons, v_i denotes the binary state of the visible neuron i, h_j denotes the binary state of the hidden neuron j, and w_{ij} represents the weight between neurons i and j. For the model parameter θ of $\{W, b, a\}$ and v_i, $h_j \in \{0, 1\}$. Here, W is the symmetric weight with dimensions of $n \times m$, a represents the biases of visible neurons, and b represents the biases of hidden neurons. The energy of a joint configuration of visible and hidden neurons determines the probability of the configuration.

$$p(v_i, h_j) = \frac{e^{-E(v_i,h_j)}}{\sum_{v_i,h_j} e^{-E(v_i,h_j)}} \quad (2)$$

The denominator in (2) is defined as a partition function and is obtained by summing all possible pairs of visible and hidden vectors. Based on the probability distribution function, the conditional probability distribution functions are derived and represented as follows:

$$h_j = sigmoid(\sum_i v_i W_{ij} + b_j) \tag{3}$$

$$v_i = sigmoid(\sum_j h_j W_{ij}^T + a_i) \tag{4}$$

where h_j represents the probability distribution when v_i is given and v_i represents the probability distribution when h_j is given.

A DBN employs pretraining, fine tuning, and prediction. Pretraining and fine tuning allow this model to predict an accurate output given an input. In the pretraining process, a set of initial parameters are obtained from the input vectors using an RBM.

B. PSO

In this section, a brief description of PSO is provided. PSO is a population-based intelligent optimization algorithm first proposed by Kennedy and Eberhart [10] by simulating social behavior models such as bird flocking and fish schooling. In PSO, the potential solutions are known as particles, and these particles move in a *D*-dimensional search space. Each particle's movement is influenced by its local best known position and best known position in a search space. This action is expected to move the swarm toward the best solutions. The position and velocity of particles for trying to improve the solutions are presented in the following equations.

$$\begin{aligned} v_{i+1}^k &= v_i^k + C_1 \times rand(0,1) \times (P_{best}^k - x_i^k) \\ &\quad + C_2 \times rand(0,1) \times (G_{best}^k - x_i^k) \end{aligned} \tag{5}$$

$$x_{i+1}^k = x_i^k + v_{i+1}^k \tag{6}$$

where v_i^k and x_i^k represent the velocity and position of the present particle, v_{i+1}^k and x_{i+1}^k represent the velocity and position of the next generation particle, and $rand(0, 1)$ denotes a random number in the range of 0 to 1. Here, P_{best}^k represents a particle's best known position, and G_{best}^k represents a swarm's best known position; C_1 and C_2 are constants.

A variant of PSO is introduced in this section. The inertia weight ω [27], [28] is considered while updating the velocity presented in (7), and the following velocity equation is obtained:

$$\begin{aligned} v_{i+1}^k = \omega_i v_i^k &+ C_1 \times rand(0,1) \times (P_{best}^k - x_i^k) \\ &+ C_2 \times rand(0,1) \times (G_{best}^k - x_i^k) \end{aligned} \tag{7}$$

Shi and Eberhart in [27] indicated that inertia weight is crucial for balancing local and global search. Local exploitation is facilitated by a smaller inertia weight, whereas global exploration is facilitated by a larger inertia weight [28]. Therefore, a parameter for ω is implemented that contributes to improve global exploration in the initial phases and local exploitation when the swarm is near the optimal solution. $C_1 = C_2 = 2$ is be set, as suggested in [29]. To set ω with a parameter, a coefficient $\lambda_\omega = 0.95$ [30] is introduced for making the parameter more efficient. Therefore, the pitching parameters are implemented in every iteration.

$$\omega_{i+1} = \lambda_\omega \omega_i, \quad \text{in which } \lambda_\omega = 0.95 \tag{8}$$

C. COGNITION LEARNING ALGORITHM

A cognition learning algorithm is designed for our robot to learn the postures of pitching baseballs at a 3×3 square target. Understanding that the architecture of the cognition learning algorithm is divided into two systems, the cognition learning algorithm framework is established using DBN and PSO algorithms that represent System 1 and System 2, respectively. The relationship of System 1 and System 2 is shown in Fig. 1. The solid arrows belong to the information flow of System 1, and the dotted line belongs to System 2. In the whole process, System 2 can be treated as the supervisor of System 1, and System 1 is also initially set up by System 2. Either System 1 or System 2 will make the decision to handle the problems. Basically, the action is made by System 1. However, if the action made by System 1 is unreasonable, System 2 will take over the process and try to make the correct decision. The procedures of the proposed cognition learning method are illustrated in the Fig. 2. Algorithmic details of each block are provided in this paper.

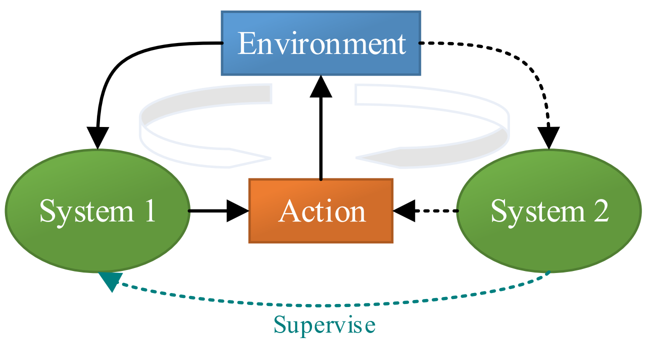

FIGURE 1. The relationship of system 1 and system 2.

First, a square area as the pitching target is selected. System 1 generates a set of outputs corresponding to the target point. The robot pitches using these outputs for the parameters of the motors in the robot. Then, the trajectory and location of the ball on the square are obtained.

The turning angle of the waist of the robot is a parameter that can influence the direction of the ball. If the waist of the robot turns to the right, then the ball moves to the right. If the turning angle of the waist and the direction of the target point are the same, then the pitching data are regarded as reasonable and accepted for training. The data are then trained using the DBN. The more data trained by the DBN, the more robust it becomes.

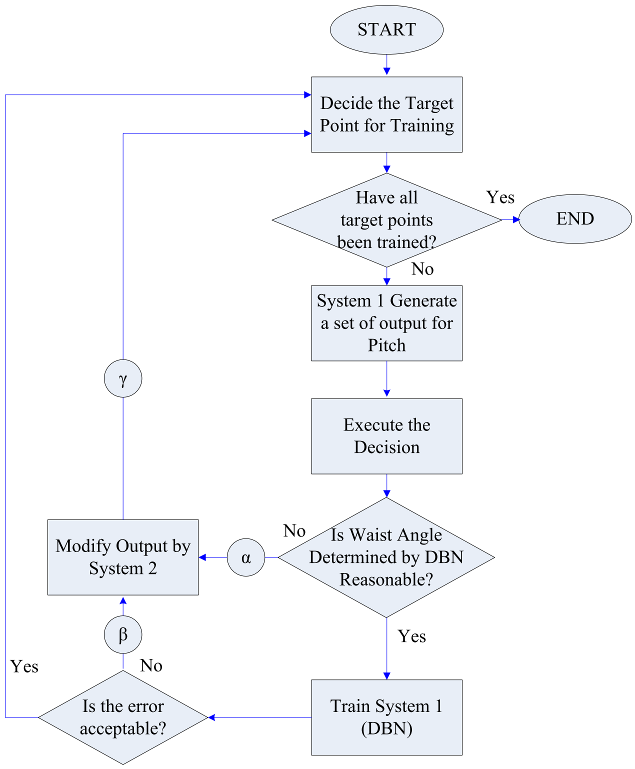

FIGURE 2. Flowchart of the proposed cognition learning method.

Then, based on this procedure, the accuracy error was checked. Accuracy error was defined as the distance between the location of a ball and the target point. If the error is smaller than a threshold, which is obtained from the experiment, then high accuracy is obtained. Once accuracy was achieved, the training of this target point was regarded as completed. Then, the next target points were trained.

Conversely, if the waist of the robot turned away from the target point or if the accuracy error was larger than 0.015, the procedure of System 2 would be conducted to modify the parameters for the same target point. System 2 supersedes System 1 and can perform more detailed and specific processing to solve the aforementioned problem. The procedure of System 2 is described in the following section.

D. DBN IN COGNITION LEARNING ALGORITHM

A DBN is developed to build an intuitive and reflexive cerebration model for the robot. The trajectory and location of the ball were influenced by four parameters of the motors on the robot during every pitch. Fig. 3 illustrates the information of the pitching process. After pitching, the location of the ball was obtained by a camera and transformed into a coordinate system. Four motors influence the pitching posture of the robot. The parameters of these motors are velocity, angle 1, angle 2, and angle 3, as shown in Fig. 3.

The width and height of a 3 × 3 square grid were normalized to 1, and the location of ball was transformed into the

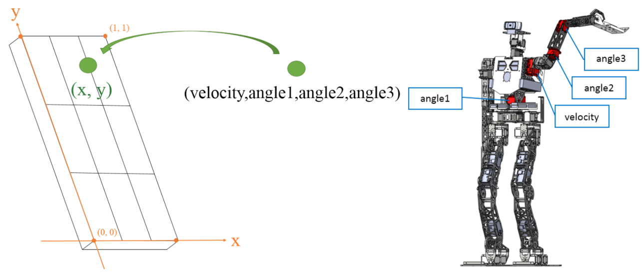

FIGURE 3. Location of the ball and corresponding parameters of the motors during the pitching process.

coordinate system. In the proposed method, we set five target points for training the DBN model.

Images of the experiment are shown in Fig. 4. In the images, the 3 × 3 square is marked by a purple rectangle; blue lines represent the trajectories of the ball, and green balls represent the locations of the ball on the square. Fig. 4 (a) presents the width of the square, and the location of the ball on the x axis can be determined using the width value. Fig. 4 (b) displays the height of the square, and the location of the ball on the y axis can be determined using the height value. When the width and height of the square are normalized to 1, the x coordinate and y coordinate can be obtained proportionally.

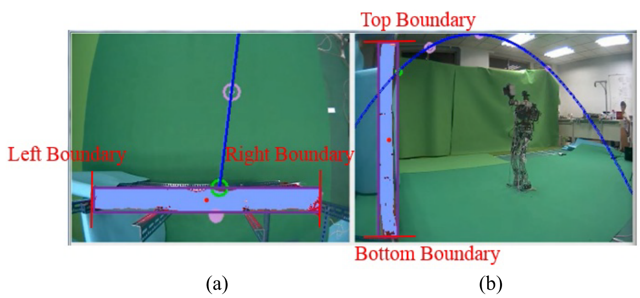

FIGURE 4. Images from the (a) top camera and (b) side camera in the experiment.

The position information was represented by the pixels in the image. The width of the image, which was captured using a camera, was 320 pixels, and the height was 240 pixels. The boundary conditions also had a corresponding pixel position, and the width and height of the square could be obtained using the boundary conditions. Then, the width and height of the grid could be normalized to one. The full range of the square could be transformed into the coordinate system, and the location of ball was the intersection of the square and trajectory. Consequently, the location of the ball on the square could be accurately determined.

As previously mentioned, there are three stages in a DBN: pretraining, fine tuning, and prediction. The trained weights and biases were obtained by conducting the pretraining process. First, the structure of an RBM in a DBN is defined. Fig. 5 presents the RBMs stacked to create a structure with three layers.

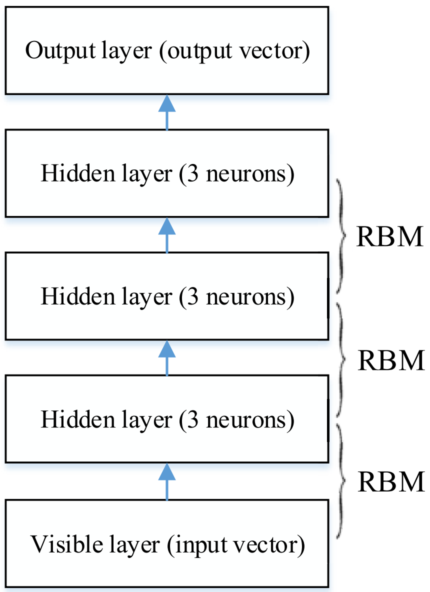

FIGURE 5. Stacking RBMs to create a DBN.

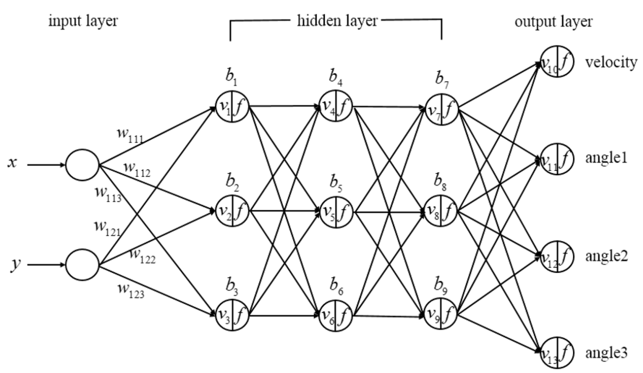

FIGURE 6. Topology of the DBN in the proposed method.

The configuration of neurons is illustrated in Fig. 6. We can clearly observe that there are two inputs, four outputs, and 3×3 hidden neurons. The input data include an x coordinate and a y coordinate. The four outputs correspond to the four parameters of the motors that decide the pitching posture.

In the pretraining process, the locations of balls from accurate pitches were used as the training data. Each training datum contains two numbers that represent the x and y coordinates in the range between 0 and 1. The iterations of pretraining were set to 100, the learning rate of pretraining was set to 0.5, and the amount of training data was based on the number of accurate pitches. The pretraining process is executed using these parameters to obtain the initial value of v_i from the input data. Note that h^s, $\tilde{v}^s$, and $\tilde{h}$ were obtained from v_i using the RBMs in sequence. Moreover, W, a, and b were updated in the contrastive divergence process. As the number of iteration increases, W, a, and b become more useful to the prediction model. In this study, the number of Gibbs sampling iterations k was set to 1.

In the fine-tuning process, the initial weights were used for the BPN. The input and output information of the BPN was obtained after pitching; the locations of the ball on the coordinate system represented the input, and the corresponding parameters of the motors were the outputs. The BPN performed a forward pass using this information and generated a set of outputs corresponding to the location of the ball. Based on the output from the forward pass, the cost function could be obtained. Then, the BPN performed a backward pass to tune the weights and biases. As the number of training data increases, the weights and biases become more beneficial to the set of data that approximates the desired output.

The aim of the cognition learning algorithm is that the DBN generates a set of outputs to predict the pitching parameters of the robot. In the prediction process, a forward pass of the BPN is performed to obtain outputs corresponding to the inputs.

The DBN performed unsupervised learning in the pretraining stage and supervised learning at the fine-tuning and prediction stages with the data of accurate pitches. The training was considered completed when all target points were trained. If no accurate pitches were thrown, System 2 would take over to help System 1 to become robust.

E. INERTIA WEIGHT PSO IN COGNITION LEARNING ALGORITHM

In this subsection, the procedure of System 2 is introduced in detail. The flowchart of System 2 is shown in Fig. 7. When System 1 could not give an output that enables the robot to pitch accurately, System 2 took over System 1 to modify the output data generated by System 1. The main algorithm of System 2 was PSO with inertia weight. First, three sets of data are generated in the vicinity of the output from System 1 as the first-generation particles. A set of data represented a particle in System 2. A swarm was formed using three particles to conduct optimization. Then, the robot pitched using the data of the three particles in the first iteration. The three locations of the ball corresponded to the three times when pitching was conducted. The fitness value of three pitching times could be estimated. However, the turning angle of the waist of a robot must be considered.

If the turning angles of the waist and direction of the target point were the same, then the pitching data were considered to reflect an accurate pitch and were used as training data. Subsequently, the data is trained using a DBN. If the pitch was not accurate, the pitch parameters will not be recorded for the DBN training. To estimate the modification performance, the distance between the target point and the location of a ball must be considered. Fig. 8 illustrates the distance between the target point and the location of a ball.

A fitness function f is designed that is represented by the following equation:

$$f = \frac{k_1}{0.1 + d} + \frac{k_2}{1 + w} \tag{9}$$

where d denotes the distance between the target point and the location of the ball, w denotes the turning angle of the

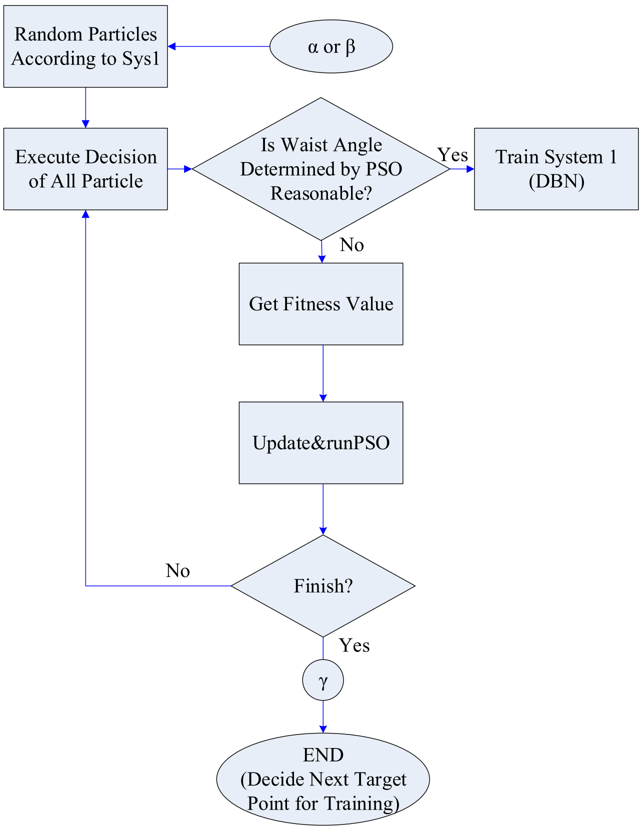

FIGURE 7. Flowchart of system 2.

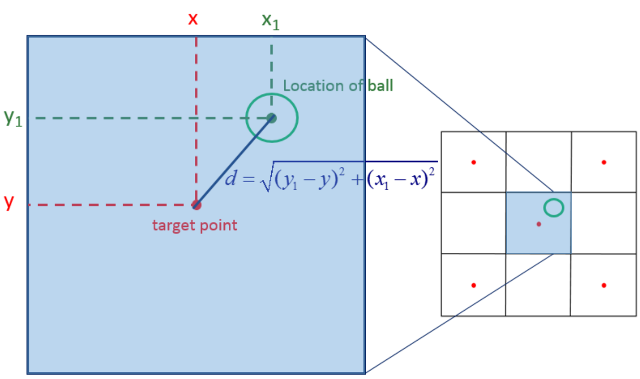

FIGURE 8. Distance between the target point and the location of the ball.

waist of the robot, and k_1 and k_2 are constants. The influence of d and w must be considered. To balance the influence of d and w, k_1 and k_2 were set to 1 and 200, respectively. The values of k_1 and k_2 are determined by heuristic method. Based on the definition of the fitness function, we inferred that the robot could hit the target point accurately with a low turning angle of its waist. After PSO with inertia weight, the velocity and position of the particle were updated. The iteration of PSO continued until the fitness value was sufficiently large or until the value did not change anymore. Then, the modification conducted by System 2 was considered complete.

IV. EXPERIMENTAL RESULTS

In this section, the experimental environment setting is introduced and the results of the cognition learning method are demonstrated.

A. EXPERIMENTAL ENVIRONMENT SETTING

The designed humanoid robot that possesses a human-like appearance with a head, a torso, and limbs. Furthermore, the strategy system of the robot behaves like the brain of a human being, and the control system of the robot behaves like a spinal cord. In this section, the mechanism, hardware, and the control system are explained.

The teen-sized humanoid robot has 26 degrees of freedom, weighs 9.6 kg, and is 95 cm tall. The material is composed of Al–Mg alloys, such as A7075 and A6061, and plastics, such as acrylonitrile butadiene styrene and polyoxy methylene resin. Photographs are presented in Fig. 9.

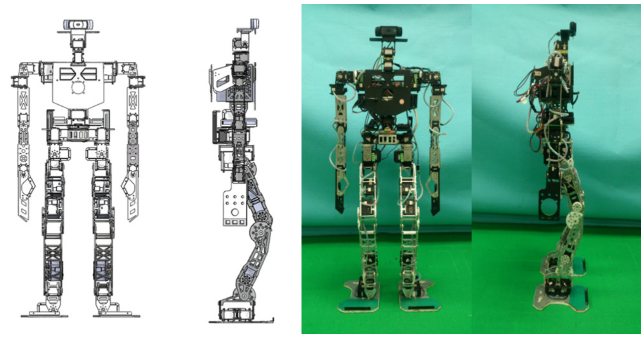

FIGURE 9. Structure of the humanoid robot.

A 3×3 square is constructed using steel angles. A red cloth covered the frame of the square to promptly extract color features using the top and side IP cameras. The arrangements for capturing images in this experiment are presented in Fig. 10.

FIGURE 10. Environment of the experiment.

The plastic boards were undecorated during the training process for expediency. The size and weight of different balls were determined. The results revealed that a tennis ball was the most suitable for the humanoid robot in the pitching game. To expedite the training process, the humanoid robot will get a ball after every pitch. Another web camera recorded the experimental procedure. The distance between the humanoid robot and the grid was set to 140 cm because of the hardware constraints of the humanoid robot. The entire environment of the experiment is displayed in Fig. 10.

B. RESULTS OF COGNITION LEARNING ALGORITHM

To verify the proposed cognition learning algorithm used for learning the posture in the pitching game, the training processes and the final test are described in this section.

As previously mentioned, the proposed cognition learning algorithm was trained during a pitching game. The data of accurate pitches were the training data for System 1 of the proposed method. Simultaneously, System 2 helped System 1 become more robust. The target points were the numbers 1, 3, 5, 7, and 9 on the 3 × 3 square board.

The comparisons of ANN and DBN are shown in Fig. 11. It shows average learning curves and error bars of ANN and DBN, which are the average of ten runs in 1500 epochs to ensure the results are objective. In Fig. 11(b), there are some outliers on the box plot when the epoch is 200, but this problem does not appear in Fig. 11(c). In conclusion, the performance and the convergence speed of DBN is faster than ANN based on the presented learning curve in terms of Root-Mean-Square Error (RMSE) and boxplots, so DBN can usefully be applied to the on-line training experiment. In this experiment, the iteration of RBM is set to 100, the learning rate of RBM is set to 0.5, and the learning rate of the back propagation network is set to 0.5.

On the other hand, once System 2 supersedes System 1, the output motion, which is generated by System 1, will be modified by the inertia weight PSO. In the process of System 2, the fitness value denotes the quality of the pitching result. According to the definition of the fitness function, the larger fitness value represents the higher accuracy of the corresponding pitching motion. Fig. 12 demonstrates the trends of the PSO algorithm's fitness values in five training targets, numbers 1, 3, 5, 7, and 9, and it also shows that the fitness values are all increased in six iterations in five training targets. In conclusion, based on the PSO algorithm, the fitness values can be obviously increased in all the tests, thus proving that the PSO algorithm can improve the performance of pitching.

We should confirm whether the five target points were sufficient for appropriately training the proposed algorithm. To train the proposed algorithm efficiently, we did not train every target point on the square. By considering all points that a ball may hit, the target points were set on the corners and center of the square. After training was completed, the final test was performed. The aim of the final test was to determine whether the humanoid robot could pitch the ball and hit the board accurately. The results of the final test are presented in Fig. 13, where nine numbered plastic boards were arranged in a grid, and the humanoid robot pitched a ball at each of the nine boards in order. Thus, the trained DBN model generated nine outputs that the humanoid robot could use as data to hit the corresponding targets. Fig. 13 depicts a series of photographs in which the humanoid robot hits the corresponding targets in sequence to demonstrate the result of applying the algorithm to the robot.

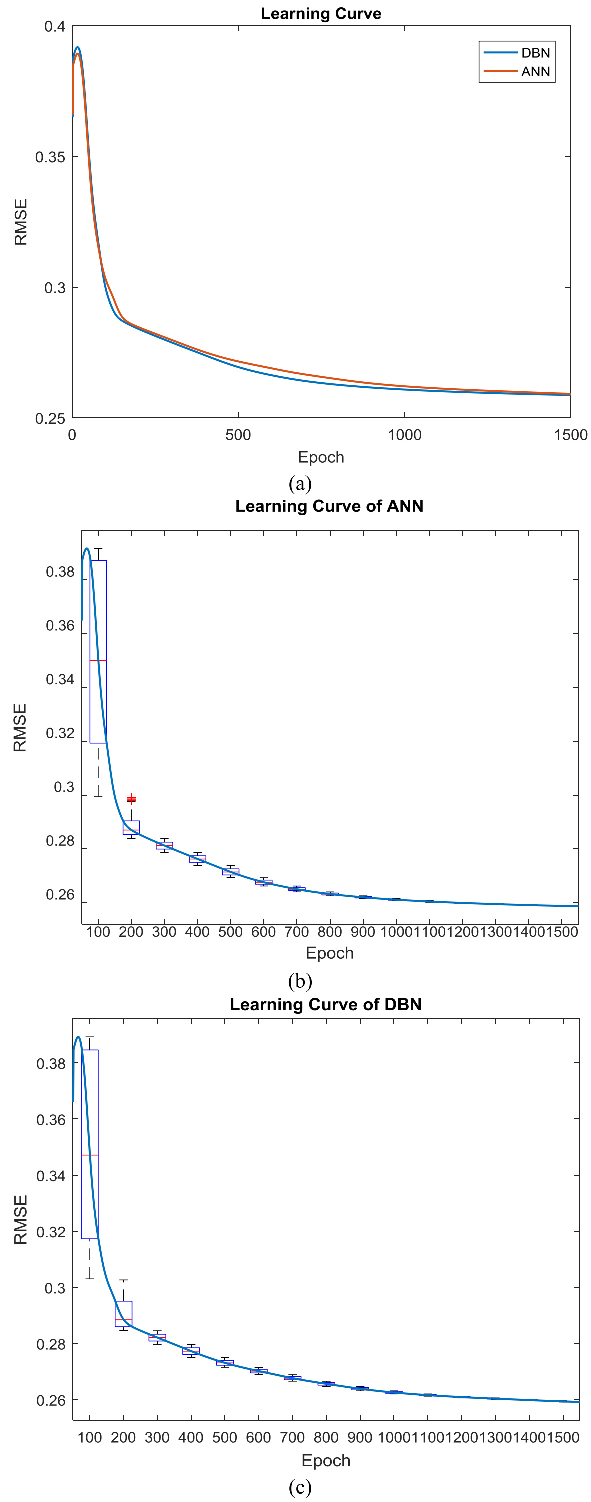

FIGURE 11. Comparisons of ANN and DBN. (a) The average learning curves of ANN and DBN. (b) The learning curve of ANN with error bar. (c) The learning curve of DBN with error bar.

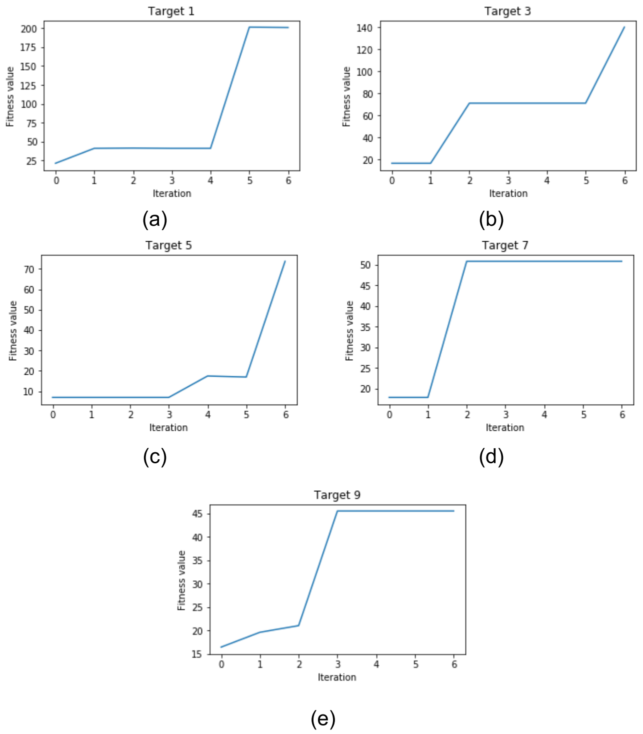

FIGURE 12. **The fitness values in five training targets.**

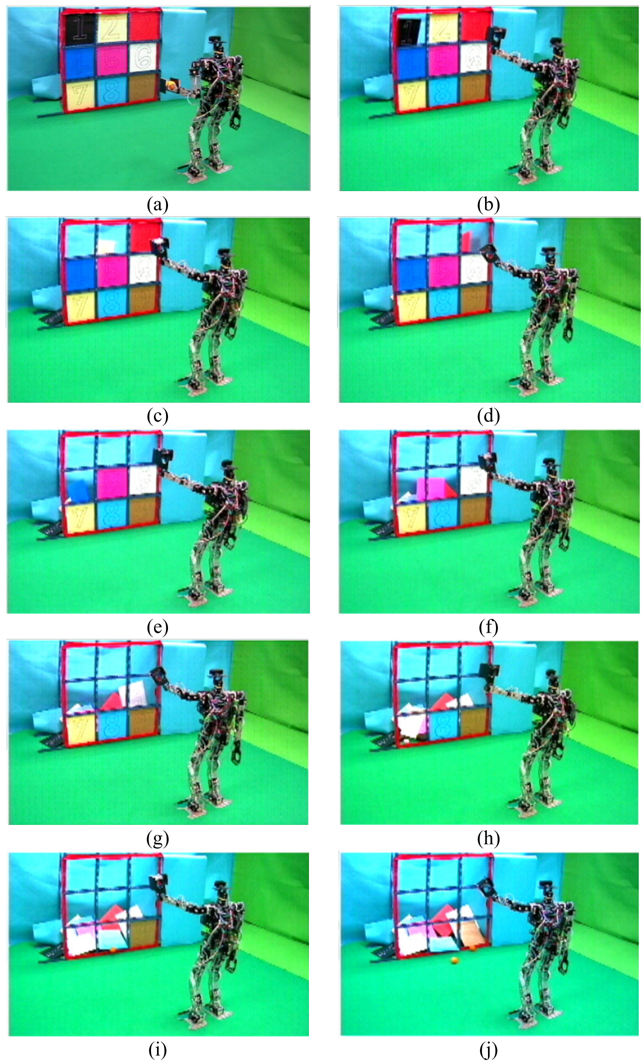

FIGURE 13. **Final test of the cognition learning algorithm.**

The aim of the experiment is for the robot to accurately pitch the ball to a selected target point on a 3×3 grid. Therefore, the location of the target point must be set as the reference input. In other words, the coordinates of the center of the boards must be keyed into the DBN. Through DBN computations, the arm parameters (i.e., the four parameters of the four motors that influence the posture of pitching) needed for the robot to accurately hit the target are obtained. Using these outputs, the humanoid robot could hit all nine boards without any missed shots. In this work, many intelligent algorithms might be applied for System 1 and System 2. However, the performances of DBN and PSO have been validated in many literatures [25], [27]. The experiments show that the combination of DBN and PSO gives good results in the robot pitching game. It also indicates that these two algorithms are suitable for the implementation of the proposed cognition learning algorithm. The whole experimental video can also be accessed in [31].

V. DISCUSSIONS

In the experiments conducted in [5], basketball shooting was performed using a fixed location as the target. Thus, learning was only required for a single target point; this can be achieved in System 1 using curve fitting. What makes the current study different is that the targets in the current study are points within a plane. Therefore, learning in the current study must be performed using multiple points across the plane; this can only be achieved by using a more complex algorithm. Therefore, for System 1, we replaced the curve fitting computational method with DBN. As for System 2, we also used a relatively complex inertial-weight PSO method to replace the conventional arm parameter adjustment model, eliminating the need for manual adjustment of the arm parameter. System 2 could automatically recalibrate the arm parameters when System 1 could not make a reasonable judgement. Accordingly, the algorithm framework proposed in the current study is relatively comprehensive and refined.

In the past, we were obliged to adjust each robot manually. Now, we can achieve fully automated training of any robot and ultimately achieve the goal of intelligent learning through the application of appropriate algorithms. Additionally, we incorporated certain psychological concepts into the training paradigm to make the algorithm a closer approximation of the actual thinking framework of a human mind; the System 1 coded in the algorithm can also be utilized to save computational time. These characteristics of the algorithm offer a huge range of future possibilities for robot learning frameworks. The theory proposed in the current study could be applied in various fields in the future. A possible area of application would be the automated training of mechanical arms in factories. During working hours, System 2 can be used in conjunction with System 1 to perform the parameter training of mechanical arms. When System 1 is not performing as intended, System 2 can be used to complement the computational process. Then, System 2 can propagate the relevant experience obtained back to System 1 to aid the establishment of a more comprehensive rapid reaction model.

The emphasis of this study is the framework constructed on the basis of psychological concepts: System 1 represents the rapid intuitive reaction mechanism, whereas System 2 represents the slower analytical thinking mechanism. Therefore, System 1 and System 2 complement each other, and together they can enhance the computational ability of an algorithm. To verify the effectiveness of the proposed algorithm, we conducted a robot pitching experiment on a 3×3 target grid. However, this is not the only field of application for this algorithm and its underlying concepts; they could be applied in various fields in real life. For example, to achieve the goal of rapid training, we could use System 1 to handle routine computations and establish a database of frequently used knowledge, and we could use System 2 in complicated computational situations where interventions are needed. In terms of the application of System 1 and System 2 in industrial mechanical arms, the rapid computations of System 1 enable it to handle most routine computational tasks. However, when the industrial mechanical arms encounter tricky training situations, System 2 could then intervene and perform complementary computations accordingly. Therefore, utilizing both systems to conduct training would be more beneficial than using a single system.

In this study, we performed trajectory tracking of the ball pitched by the robot (illustrated in Fig. 4); this was achieved by first capturing the trajectories of the target ball, followed by performing curve fitting using the captured trajectory coordinates. In terms of vibration suppression, the motors of the robot were operating at a high speed when the robot was pitching the ball. The effects of slight vibrations on motors that were operating at a high speed were negligible. Accordingly, slight vibrations would also not have a great influence on the ball pitching action of the robot. In terms of the robot's mechanical design, because the torque at the moment of pitching is quite large, the supporting motors of the motor need to sustain a higher level of loading at the instance of pitching. If the robot's body is not supported by a good mechanical design, excessive vibrations would be generated at the instance of pitching, and this would in turn lead to unstable pitching or even damage to motors. Therefore, a refined mechanical design would be helpful for maintaining good stabilization and suppressing vibrations during the pitching action.

In the current study, the tennis ball pitched by the robot in the pitching experiment was approximately 58 g in weight; the maximum weight that the robot can grab is approximately 200 g, which is dependent on the mechanical design and construction materials of the robot. Given an appropriate design, the maximum loading that the robot's hand could sustain could be effectively enhanced, allowing the robot to grab heavier objects and enabling its application in various fields. Therefore, there remains a large space for improvement in this aspect.

From the psychology perspective, System 1 represents rapid intuitive reaction, whereas System 2 represents rational and logical thinking; therefore, System 2 is more time consuming than System 1. After the training for System 1 is completed, the user can quickly obtain relevant feedback by entering the required information into the DBN. Therefore, System 1 represents the more rapid intuitive reaction. As for System 2, it can only produce a solution after a lengthy period of trial-and-error learning. Therefore, the training of System 2 could be achieved using an optimization algorithm such as PSO. Therefore, at the learning stage, performing pitching training at the same grid constitutes the training process for System 2 (PSO). Of course, such training processes have been assumed to constitute the past experience of the robot, and therefore the information about these training processes would be propagated back to System 1. Hence, DBN would definitely perform the adjustment of weights using backpropagation. However, once the training of System 1 (DBN) is complete, the trained System 1 can straightaway be used to estimate the pitching posture appropriate for the selected target location. At this stage, System 1 could readily infer the appropriate pitching posture on the basis of the information regarding the target location. Hence, the reaction at this stage is rapid and intuitive (as illustrated in Fig. 11). Such operations are reasonable and feasible, and they are close to the actual operational modes of the human mind. Therefore, we can assume that human thinking can be divided into two parts—rapid intuitive reaction and deliberation of past experience—and that humans change their instantaneous reactions using their deliberative process. The proposed DBN is suitable for expressing intuitive reactions not only because a trained DBN could produce timely outputs based on given inputs but also because it constantly adjusts and improves the next behavioral reaction using the back propagation of its fine-tuning process. Because PSO is a rapid and highly accurate algorithm, using it to express the human thinking process is highly appropriate. Although the thinking operations of PSO are not instantaneous, the required execution time is reasonable given typical human thinking speed. The researchers combined PSO and DBN in the learning algorithm; specifically, the researchers consistently used the thinking processes of PSO to adjust and improve the next reaction until the reactions exhibited by DBN reached a certain level of accuracy.

In the cognition learning algorithm proposed in the current study, System 1 and System 2 are trained alternatively during the training stage until System 1 (DBN) attained the predetermined accuracy rate. Such a design of the system can be clearly explained using human lived experience. When we are facing with the occurrence of an event, it is natural for

TABLE 1. Nominal definition list.

Jargon nouns	Nominal Definition
ANNs	Mathematical models that imitate biological neural structure
CNNs	Neural networks with convolution, pooling layers, and fully connected layers
DBNs	Neural networks with hidden RBM layers that can model data
PSO	An algorithm that uses features of social models to reach optimization
RBM	A pretraining process for DBNs with a visible layer and a hidden layer that can extract the features of data
RNNs	Recurrent ANNs that can access internal states to solve time-series problems
System 1	A basic cognitive system that does intuitive tasks
System 2	A cognitive system that can think rationally and overcome the impulses or weaknesses of System 1

us to exhibit an intuitive reaction or for our brain to make an immediate decision; this is represented by System 1 in the learning algorithm. After the occurrence of an event, people reflect on the outcomes of the event and deliberate whether they could have performed better. Such a thinking process enables us to improve our intuitive reactions in the case of a similar event in the future, and this part of human thinking is represented by System 2 in the learning algorithm. Therefore, we tend to continually repeat the aforementioned alternate thinking processes until we feel that our intuitive reaction is good enough. In other words, we tend to repeat the alternate thinking process until our intuitive reaction (System 1) is sufficient for handling the demand of the event and the after-the-fact reflection of System 2 is not required. Therefore, in the experiment illustrated in Fig. 13, the intervention of System 2 was not available; only System 1 was in operation during the experiment. At the training stage, System 2 only intervened in 5 training target points. However, in the experiments, we observed that even when only DBN (System 1) was used, the robot could infer the reasonable pitching posture using System 1 and subsequently achieve 100% accuracy in throwing balls at all nine target points.

VI. CONCLUSION

In this study, a cognition learning algorithm is designed and implemented to enable a humanoid robot to learn the posture for pitching in baseball style to a 3×3 target grid. The inspiration for the proposed cognition algorithm was obtained from the book *Thinking, Fast and Slow* by Daniel Kahneman. The book mainly explains the human cerebration model from the perspective of cognitive psychology.

In this paper, the proposed cognition learning algorithm is introduced in detail. The algorithms used in the proposed method were a DBN and PSO. The procedure of the proposed method is explained in detail and illustrated using flowcharts. The cooperation between System 1 and System 2 of the proposed method are presented.

Additionally, experimental results are demonstrated. The training process and final verification are recounted. The humanoid robot exhibited 100% accuracy of pitching a tennis ball to the 3×3 grid. The experiment results demonstrated that the proposed cognition learning algorithm implemented in the robot has powerful and robust capability for autonomous learning. The proposed method also verifies that the cerebration model of a human being can be realized in a robot.

APPENDIX

In this paper, the nominal definition list is given in Table 1.

REFERENCES

[1] L.-H. Juang and J.-S. Zhang, "Visual tracking control of humanoid robot," *IEEE Access*, vol. 7, pp. 29213–29222, 2019.

[2] J.-L. Lin and K.-S. Hwang, "Balancing and reconstruction of segmented postures for humanoid robots in imitation of motion," *IEEE Access*, vol. 5, pp. 17534–17542, 2017.

[3] J. G. N. De Carvalho Filho, E. A. N. Carvalho, L. Molina, and E. O. Freire, "The impact of parametric uncertainties on mobile robots velocities and pose estimation," *IEEE Access*, vol. 7, pp. 69070–69086, 2019.

[4] S. S.-D. Xu, H.-C. Huang, Y.-C. Kung, and S.-K. Lin, "Collision-free fuzzy formation control of swarm robotic cyber-physical systems using a robust orthogonal firefly algorithm," *IEEE Access*, vol. 7, pp. 9205–9214, 2019.

[5] T.-H. S. Li, P.-H. Kuo, Y.-F. Ho, C.-Y. Liu, T.-C. Yu, Y.-T. Ye, C.-Y. Chang, G.-Y. Chen, C.-W. Chien, W.-C. Chen, L.-F. Wu, and N.-C. Fang, "Robots that think fast and slow: An example of throwing the ball into the basket," *IEEE Access*, vol. 4, pp. 5052–5064, 2016.

[6] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Adv. Eng. Softw.*, vol. 69, pp. 46–61, Mar. 2014.

[7] S. Mirjalili and A. Lewis, "The whale optimization algorithm," *Adv. Eng. Softw.*, vol. 95, pp. 51–67, May 2016.

[8] M. Dorigo and K. Socha, "Ant colony optimization," in *Handbook of Approximation Algorithms and Metaheuristics*. London, U.K.: Chapman & Hall, 2007.

[9] M. Mitchell, "Genetic algorithms: An overview," *Complexity*, vol. 1, no. 1, pp. 31–39, 1995.

[10] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proc. IEEE Int. Conf. Neural Netw. (ICNN)*, vol. 4, Nov./Dec. 1995, pp. 1942–1948.

[11] S. Ding, H. Li, C. Su, J. Yu, and F. Jin, "Evolutionary artificial neural networks: A review," *Artif. Intell. Rev.*, vol. 39, no. 3, pp. 251–260, Mar. 2013.

[12] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial bee colony (ABC) algorithm," *J. Global Optim.*, vol. 39, no. 3, pp. 459–471, 2007.

[13] V. Roberge, M. Tarbouchi, and G. Labonté, "Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning," *IEEE Trans. Ind. Informat.*, vol. 9, no. 1, pp. 132–141, Feb. 2013.

[14] N. Todorovic and S. Petrovic, "Bee colony optimization algorithm for nurse rostering," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 43, no. 2, pp. 467–473, Mar. 2013.

[15] L. A. Gardner, "Thinking, fast and slow," *J. Risk Insur.*, vol. 79, no. 4, pp. 1143–1145, 2012.

[16] C.-H. Kuo, T.-S. Chen, H.-C. Chou, and G.-Z. Chen, "PN-WSNA-based eye–hand–leg coordination with a FIRA HuroCup basketball game," *IEEE/ASME Trans. Mechatronics*, vol. 18, no. 3, pp. 854–866, Jun. 2013.

[17] J.-S. Hu, M.-C. Chien, Y.-J. Chang, S.-H. Su, and C.-Y. Kai, "A ball-throwing robot with visual feedback," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2010, pp. 2511–2512.

[18] Z. Ren, D. Dong, H. Li, and C. Chen, "Self-paced prioritized curriculum learning with coverage penalty in deep reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2216–2226, Jun. 2018.

[19] B. Luo, Y. Yang, and D. Liu, "Adaptive Q-learning for data-based optimal output regulation with experience replay," *IEEE Trans. Cybern.*, vol. 48, no. 12, pp. 3337–3348, Dec. 2018.

[20] H. Ge, Y. Song, C. Wu, J. Ren, and G. Tan, "Cooperative deep Q-learning with Q-value transfer for multi-intersection signal control," *IEEE Access*, vol. 7, pp. 40797–40809, 2019.

[21] G. RepovšÂă and A. Baddeley, "The multi-component model of working memory: Explorations in experimental cognitive psychology," *Neuroscience*, vol. 139, no. 1, pp. 5–21, 2006.
[22] D. L. Schacter and E. F. Loftus, "Memory and law: What can cognitive neuroscience contribute?" *Nature Neurosci.*, vol. 16, no. 2, p. 119, 2013.
[23] S. Vinogradov, M. Fisher, and E. de Villers-Sidani, "Cognitive training for impaired neural systems in neuropsychiatric illness," *Neuropsychopharmacology*, vol. 37, no. 1, p. 43, 2012.
[24] X. Cai, D. V. Prokhorov, and D. C. Wunsch, II, "Training winner-take-all simultaneous recurrent neural networks," *IEEE Trans. Neural Netw.*, vol. 18, no. 3, pp. 674–684, May 2007.
[25] G. E. Hinton, S. Osindero, and Y. W. Teh, "A fast learning algorithm for deep belief nets," *Neural Comput.*, vol. 18, no. 7, pp. 1527–1554, 2006.
[26] T. He, W. Huang, Y. Qiao, and J. Yao, "Text-attentional convolutional neural network for scene text detection," *IEEE Trans. Image Process.*, vol. 25, no. 6, pp. 2529–2541, Mar. 2016.
[27] Y. Shi and R. Eberhart, "A modified particle swarm optimizer," in *Proc. IEEE Int. Conf. Evol. Comput. IEEE World Congr. Comput. Intell.*, May 1998, pp. 69–73.
[28] K. Lei, Y. Qiu, and Y. He, "A new adaptive well-chosen inertia weight strategy to automatically harmonize global and local search ability in particle swarm optimization," in *Proc. 1st Int. Symp. Syst. Control Aerosp. Astronaut.*, Jan. 2006, pp. 1–4.
[29] Y. Shi and R. C. Eberhart, "Parameter selection in particle swarm optimization," in *Proc. Int. Conf. Evol. Program.*, 1998, pp. 591–600.
[30] V. Kalivarapu, J.-L. Foo, and E. Winer, "Improving solution characteristics of particle swarm optimization using digital pheromones," *Struct. Multidiscipl. Optim.*, vol. 37, no. 4, pp. 415–427, 2009.
[31] *Experimental Video*. Accessed: Jan. 16, 2019. [Online]. Available: https://youtu.be/U3xklh3nMUs

TZUU-HSENG S. LI (S'85–M'90) received the B.S. degree from the Tatung Institute of Technology, Taipei, Taiwan, in 1981, and the M.S. and Ph.D. degrees from National Cheng Kung University (NCKU), Tainan, in 1985 and 1989, respectively, all in electrical engineering. From 1996 to 2009, he was a Researcher with the Engineering and Technology Promotion Center, National Science Council, Tainan. From 1999 to 2002, he was the Director of the Electrical Laboratories, NCKU. From 2009 to 2012, he was the Dean of the College of Electrical Engineering and Computer Science, National United University, Miaoli City, Taiwan. From 2009 to 2016, he was the Vice President of the Federation of International Robot-Soccer Association. Since 1985, he has been with the Department of Electrical Engineering, NCKU, where he is currently a Distinguished Professor. Since 2014, he has been the Director of the Center for Intelligent Robotics and Automation, NCKU. His current research interests include artificial and biological intelligence and applications, fuzzy system and control, home service robots, humanoid robots, mechatronics, 4WIS4WID vehicles, and singular perturbation methodology. He was a CACS fellow and a RST fellow, in 2008 and 2018, respectively. He was a recipient of the Outstanding Automatic Control Award, in 2006, from the Chinese Automatic Control Society (CACS), Taiwan, and the Outstanding Research Award, in 2017, from the Ministry of Science and Technology, Taiwan. He was a Technical Editor of the IEEE/ASME TRANSACTIONS ON MECHATRONICS and an Associate Editor of the *Asia Journal of Control*. He is currently a Co-Editor-in-Chief of *iRobotics* and Associate Editors of the *International Journal of Electrical Engineering*, the *International Journal of Fuzzy Systems*, and the IEEE TRANSACTIONS ON CYBERNETICS. He was elected as the President of the CACS, from 2008 to 2011, and the Robotics Society of Taiwan, from 2012 to 2015.

PING-HUAN KUO received the B.S., M.S., and Ph.D. degrees from the Department of Electrical Engineering, National Cheng Kung University, Tainan, Taiwan, in 2008, 2010, and 2015, respectively. Since 2017, he has been with the Computer and Intelligent Robot Program for Bachelor Degree, NPTU, where he is currently an Assistant Professor. His major research interests include fuzzy control, intelligent algorithms, humanoid robot, image processing, robotic application, big data analysis, machine learning, and deep learning applications.

CHIEN-YU CHANG received the B.S. degree from the Department of Electronic Engineering, Chung Yuan Christian University, Taoyuan, Taiwan, in 2009, and the M.S degree from the Department of Electrical Engineering, National Cheng-Kung University, Tainan, Taiwan, in 2016. His major research interests include fuzzy control, intelligent systems, humanoid robot, image processing, robotic application, and FIRA/RoboCup game.

HAO-PING HSU received the B.S. degree from the Department of International Bachelor Degree Program on Energy, National Cheng Kung University, Tainan, Taiwan, in 2019, where he is currently pursuing the M.S. degree with the Department of Electrical Engineering. His major research interests include fuzzy control, intelligent systems, humanoid robot, image processing, robotic application, and FIRA game.

YUAN-CHIH CHEN received the B.S. degree from the Department of Mechanical Engineering, National Taiwan University, Taipei, Taiwan, in 2019. He is currently pursuing the M.S. degree with the Department of Electrical Engineering, NCKU, Tainan, Taiwan. His major research interests include fuzzy control, intelligent systems, humanoid robot, image processing, robotic application, and FIRA game.

CHIEN-HSIN CHANG received the B.S. degree from the Department of Engineering Science, National Cheng Kung University, Tainan, Taiwan, in 2019, where he is currently pursuing the M.S. degree with the Department of Electrical Engineering. His major research interests include fuzzy control, intelligent systems, humanoid robot, image processing, robotic application, and FIRA game.