In this paper, we propose a robotic cognitive system which can learn itself to do a specific assignment by accumulating experiences through bottom-up thinking to make decision by itself via top-down thinking according to the experiences. That is, the cognitive system has a self-learning ability which can accumulate its experiences to make itself smarter. In essence, the cognitive system possesses a perception model, a memory model, and a hypothesis model. The perception model converts image information into perception codes. The memory model stores experiences in the past and present to provide to the perception model and the hypothesis model. The hypothesis model, which generates the next decision according to the experiences from the memory model, is the most important part of the proposed cognitive system. To validate the performance of the proposed system, we utilize Chinese calligraphy writing tasks by a virtual robot through simulation to evaluate the abilities of the cognitive system. In order to generate the coordinates of the writing brush, we made the virtual robot practice to learn Chinese calligraphy through bottom-up thinking to construct the writing patterns. The illustrative examples in this paper show that the virtual robot can learn to write Chinese calligraphy by top-down thinking according to its own experiences.
robot will only be able to perform according to the previously recorded coordinates. It won’t be able for the robot to know the causal relationship of input and output, which may cause it to react incorrectly. To solve this problem, a fundamental cognitive system needs to be established to allow the robot in recognizing the relevance of the task’s input and output.

By combining cognitive psychology and robotics, [6], [7] revealed a psychological structure with schema and chain for robots to learn simple tasks. These studies showed good effect on the application of higher-level cognitive strategies for robots. However, it appeared difficult for these methods to learn high-dimensional and complex low-level cognitive processing. Note that the relationship between the image observed by the robot and the motion trajectory for the robot is complicated. It would be difficult to evaluate these high-order nonlinear relationships via a statistical approach. Artificial neural networks, on the other hand, are well suited for finding these complex associations. Thus, the proposed cognitive system is designed using a deep learning-based approach. Furthermore, by referring to the concept of cognitive computing, the proposed cognitive system also allows the robot to adapt to different environments.

The cognitive computing systems are generally information adept, dynamic training and adaptive, probabilistic, highly integrated, meaning-based, and highly interactive [9], [10], [11] in nature. Information adept property is an ability that can synthesize ideas or answers through integrating multiple heterogeneous sources. Dynamic training and adaptive property makes the model possible to learn when it receives new information. Probabilistic property predicts the probability of valuable connections between relevant patterns based on context. Highly integrated property requires a central learning system which can interact with historical data of all modules. Meaning-based property concerns the structure of language, semantics, and relationships which are explainable. Highly interactive property is designed for human-model interaction, which allows the model to be able to communicate with a human.

To design a cognitive system, it is necessary to use the components of cognitive computing [8], [10], [12], [13], [14], [15]. We refer to the mental process of human thinking as the design concept of the cognitive system, and then use the framework of cognitive computing to design the system. In a human cognitive system, perception, memory [16], and hypothesis generation are the key operating elements. Therefore, to establish a cognitive system that resembles a human’s cognition, these three elements must be considered.

Most cognitive computing research focuses on perception and information organization, similar to methods for pattern recognition [17], [18], [19], [21], [21], by referring to human perceptual processing computer vision is used to implement feature extraction process. The proposed cognitive system extends the perception process to the sensory motor system. In addition to perceptual processing, we also add processing of memory and hypothesis generation. Through the interaction of these models, the cognitive learning process of image to action is established.

The first key element in cognitive systems is perception [22], [23], [24], [25], [26], where the brain organizes and interprets incoming stimulus signals. The processing method of the perception is divided into bottom-up and top-down processing [27], [28], [29]. Upon receiving the stimulus signal, the human brain will extract, analyze, and organize the characteristics of the stimulus. This process is called bottom-up processing. Top-down processing establishes a connection between stimulus signals and experiences based on memory. Humans perceive the world and establish concepts through two perceptual ways of processing (bottom-up and top-down processing). The perception process is divided into three stages [28]: sensation, perceptual organization, and recognition. These functions correspond to the current convolutional neural network’s (CNN’s) [29], [30] input layer, convolutional layer and fully connected layer, respectively. In designing the cognitive system of robots, we have not used it for visual recognition. The proposed cognitive system only includes the process of perceptual organization of robot perception, which enables the robot to convert physical signals to neural signals.

Memory is a space for storing and retrieving information. The proposed cognitive system also includes a memory model to make sure that the system has the ability to remember its past experiences. The most classic memory storage operation in psychology is called the Multi-Store Model [31], [32], which describes how information is processed in memory, where the memory is divided into three different types: sensory register (SR), short-term memory (STM), and long-term memory (LTM) [27], [28], [31], [32], [33], [34]. Through the interaction of the three types of memory, we can adopt the concept of Multi-Store Model theory to the proposed cognitive system, so that the robot has the memory and can use its past experiences for future decision-making.

Over the past years, a lot of research has designed explainable AI models [35], [36]. However, the proposed method focuses only on information adept, dynamic training and adaptive, probabilistic, and highly integrated characteristics of cognitive computing. Although meaning-based property is also an important problem to be deal with, the behavior of our proposed model can only be analyzed in an indirect way by checking long-term memory.

The proposed cognitive system is capable of self-adaptation and accumulating experiences for learning, where the operating mechanism uses the perception model (the sensation and perceptual organization), memory model (Multi-Store Model), and hypothesis generation model [12], [38] of cognitive psychology. These models can be established using the learning ability of deep neural networks. Thus, we use convolution neural networks to perform complex multi-model training and simulate basic mental processes.

How to establish cognitive abilities for robots is an important research topic. The research revealed in [39] combined
cognitive science and reinforcement learning to design a cognitive controller to establish a state-of-the-art cognitive system for cognitive tracking radar. By referring to cognitive science, the structure of their design consisted of a cognitive controller, a short-term memory, a cognitive preceptor, and an environment. This cognitive system was successfully applied to the field of communication with satisfactory results. Unfortunately, in the field of robotics, robots are often required to generate an entire motion at one time. That is to say, the robot won’t be able to receive a reward corresponding to a single output pose in the motion. This has made it difficult to implement such a reward-based reinforcement learning method in this situation. Through the proposed cognitive system of robots, the relationship between outputs and inputs can be established by itself without a reward function. The outputs can be modified according to the understood state associations, so that the robots have a high environmental adaptability. By combining cognitive computing, robot theory and cognitive psychology, a fundamental robot cognitive system can therefore be established in this paper.

II. COGNITIVE SYSTEMS OF ROBOTS

The important parts of the proposed cognitive system include a memory model, hypothesis model, and perception model. Figure 1 illustrates a learning-based robot with the proposed cognitive system, in which the architecture of the proposed cognitive system is shown in Fig. 2. For clarity and easy reference, the nomenclature is shown in Table 1.

In Fig. 1, the perception model encodes image data (inputs) to extract information. The memory model records past perception codes and past hypotheses. The hypothesis model creates hypotheses to generate actions to send to the motor system. The motor system controls the motions (outputs) of the robot according to the actions sent from the hypothesis model. $R_{mp} = \{M_k, O_k\}$ is the information from the memory model which helps the perception model adapt to various environments. $R_{pm} = \{C_{M_k}, C_{O_k}, C_{O_k}, C_{O_k}\}$ and $R_{hm} = \{A_k\}$ are the information from the perception model and the hypothesis model that need to be retained in the memory model. $R_{ph} = \{C_{O_k}, C_{M_k}\}$ is the perception code which helps the hypothesis model make new hypotheses. $O_k^*$ is the expected observation image as the input of the system. $O_k$ is the observation image as the output of the system which is captured by a camera after finishing the motion.

With these fundamental components, the proposed cognition system can learn to solve simple tasks. Most robotic research focuses on offline programing instead of online learning to solve uncertain tasks. With the proposed cognitive system, the robot is able to learn tasks online from accumulated experiences.

In Fig. 2, the proposed cognitive system can learn to make the current observation vector $O_k$ (output) approximate the expected observation vector $O_k^*$ (input) through a learning process. A Hypothesis Net (DNN1) can generate the hypothesis $A_k$ as the action according to the expected observation vector $O_k^*$ (input). When switch $S_1$ is closed, the robot performs actions through the motor system according to the hypothesis $A_k$. The motor system controls the robot’s movement based on the hypothesis of the action which is sent by the hypothesis model. After the motor system finishes the action, the proposed cognitive system observes the result (output) as the observation vector $O_k$. Then, the cognitive system retains the observation vector $O_k$, perception code $C_{O_k}$, and action vector $A_k$ in the memory model. The memory consolidation process controlled by switches $S_2$, $S_3$ and $S_4$ helps Memory Net (DNN2) memorize past experiences. Finally, the hypothesis model can generate new hypotheses $A_{k+1}$ as actions of the motor system via these experiences.

![Figure 1: Schematic diagram showing the learning-based robot with the proposed cognitive system.](image-url)
III. PERCEPTION MODEL

Image processing technology has solved many problems and made robots more intelligent than ever before. However, it is hard to describe abstract information in image form. Inspired from human perception, we built neural networks for robot perception in order to encode images into the form of neural information, because humans do not process image data directly, but rather covert images into neural information in the brain.

The striate cortex (primary visual cortex) is the first image processing area in the occipital cortex, which processes the edge and contour information at the first stage of the occipital lobe [40], [41], [42]. Then the neural signal is divided into two paths: the dorsal stream and the ventral stream [43]. The dorsal stream projects the signal to the posterior parietal lobe [40], [41], [42]. Then the neural signal is divided into the primary visual cortex to the inferotemporal cortex which identifies the concept of the object.

Convolutionsal neural networks extract features with rings or edges in the first layer, which also occurs in classification and auto encoder [44] tasks. We tried to train the auto encoder with the cifar-10 dataset, and visualized the first layer of the auto encoder [44] tasks. We tried to train the auto encoder and auto encoder [44] tasks.

IV. MEMORY MODEL

The memory model can help the robot store and consolidate memories, making the cognitive system to remember past experiences. The hypothesis model does not generate hypotheses randomly, but rather according to the memories to achieve the expected results. The memory model shown in Fig. 5 includes a Memory Net (DNN2), a short-term memory (STM), and a long-term memory (LTM). A_k is the hypothesis generated by the Hypothesis Net (DNN1). A_k is the set of the past hypotheses from the short-term memory and long-term memory. X_k is the sampling data vector which consists of the past observation vector X_k and the past perception code X_k of the observation vector. S_k is a switch which controls the training process of Memory Net (DNN2) to store and consolidate memories. S_k is a switch that chooses the data, M_k or C_{M_k}, to consolidate memories. S_k is a switch that chooses the memory retrieval process A_k or the memory consolidation process A_k. The main parts of memory model are shown below:

1. Memory Net (DNN2): A 16-layer deep neural network, which is the important part of the memory model [8]. It learns to imagine the result of the observation image according to the hypotheses from the hypothesis model. Memory Net (DNN2) can help Hypothesis Net (DNN1) generate better hypotheses through accumulating experiences M_k.

2. Short term memory (STM): A memory space which stores recent hypotheses A_k, observation vectors, O_k and O_k, and perception codes, C_{O_k}, C_{O_k}, and C_{O_k}, with a structure like a queue. The size of short-term memory is limited to l_k.
As soon as a new hypothesis is created, the hypothesis, observation vectors, and perception codes are added into the short-term memory shown in Fig. 6.

(3) Long term memory (LTM): A memory space that stores past experiences including hypotheses, observations and perception codes from the short-term memory. LTM consists of three memory subspaces: positive memory, negative memory, and impressive memory. The positive memory space stores past experiences that have a good quality score of the past perception code. In other words, the past perception code evaluates if the expected observation \( C_{O_T} \) and the current observation \( C_{O_k} \) are similar. On the other hand, the negative memory is a memory space which stores the past experiences that have the bad quality score of the past perception code. The size of the positive memory is limited to \( l_1^P \), and the size of negative memory is limited to \( l_1^N \). The impressive memory is defined as a memory space which stores past experiences that have various scores of the past perception code to the long-term memory. The size of the impressive memory is limited to \( l_1^I \). The Long-term memory is the important part to consolidate memories in Memory Net (DNN2). The positive memory provides experiences which could be used to solve the tasks, and the hypothesis model can easily refer to these experiences to make better quality hypotheses that lead the cognitive system in the right way. The negative memory remembers bad hypotheses generated by the hypothesis model, to avoid making the same mistakes. The impressive memory ensures the diversity of the long-term memory, and allows the hypothesis model to avoid always making the same hypotheses. This increases the imagination of hypothesis maker by preventing Hypothesis Net (DNN1) from overfitting.

(4) Memory consolidation: Considering the number of neurons is limited, we cannot provide the cognitive system with infinite memory space. As like a human, the cognitive system needs to consolidate its memories by recalling them again and again. The consolidation process is divided into two stages. The first stage is observation imagination, where Memory Net (DNN2) learns to imagine the observation vector \( X_k^O \) related to the hypothesis \( A_k^* \) shown in Fig. 6. At this stage, \( S_2 \) is closed, \( S_1 \) switches to \( M_1 \) and \( S_4 \) switches to \( A_k^* \). In this process, \( M_j \) approximates \( O_j \), where \( j = 1, 2, \ldots, k \). The second stage is perception code imagination. The memory model consolidates the past perception code \( X_k^C \) of the observation vector, and \( S_3 \) switches to \( C_{M_k} \). In this process, \( C_{M_j} \) approximates \( C_{O_j} \), where \( j = 1, 2, \ldots, k \).

V. HYPOTHESIS MODEL

Figure 7 shows the proposed hypothesis model for making new hypotheses, which consists of a Hypothesis Net (DNN1), an action switch \( S_1 \), a judgement switch \( S_3 \), and a memory retrieval. The Hypothesis Net (DNN1) is a 11-layer neural network that generates hypothesis \( A_k \) according to its weights and the expected input image \( O_k^* \). The action switch \( S_1 \) decides if hypothesis \( A_k \) will be sent to the motor system. The judgement switch \( S_3 \) changes hypotheses through updating the weight of Hypothesis Net (DNN1).

It is important to update the weights of Hypothesis Net (DNN1) to change hypotheses. The key part of the memory retrieval process is to evaluate the similarity between the perception code \( C_{O_T}^* \) of the expected observation and the perception code \( C_{M_j} \) of the memorized observation vector as shown in Fig. 8. The loss function is defined as:

\[
\text{loss}_h = \frac{1}{n_c} \sum_{i=1}^{n_c} \left( C_{O_T}^* - C_{M_i} \right)^2,
\]

where \( n_c \) is the length of the perception code, and \( C_{O_T}^* \) and \( C_{M_i} \) are the perception codes sent from the perception model. By minimizing the loss function (1) between the perception code of the expected image and the perception code of the memorized observation, Hypothesis Net (DNN1) can generate the next hypothesis with better quality.

VI. CHINESE CALLIGRAPHY WRITING TASK

The challenge of writing calligraphy is not the same as writing with a pen, because the brush tip made of goat or wolf hair of the writing brush will always change in different shapes corresponding to the posture of the shaft of the writing brush. When writing with a pen, only the position coordinates on the 2D plane \([x, y]^T\) need to be considered. When writing with a writing brush, multi-dimensional postures \([x, y, z, \theta, \bar{\theta}]^T\) need to be considered. In Chao et al. [46] used generative adversarial nets to allow robots to learn to write calligraphy characters. Through the deep learning models, the robot can practice writing calligraphy on its own. However, this method
ended up controlling only the position coordinates \([x, y, z]^T\)
and failed to consider the movements of rotation and tilt.

Our proposed cognitive system, however, not only allows the
robot to learn to generate position coordinates \([x, y, z]^T\)
but also the movements of rotation and tilt \([x, y, z, \theta, \bar{\theta}]^T\).

To evaluate the abilities of the proposed cognitive system
in this paper, we chose a Chinese calligraphy writing task
which requires complex motions to be controlled. With a cal-
ligraphy writing sample from a human, the system can gen-
erate coordinates to control a virtual robot \([8], [47]\) to write
Chinese calligraphy which is shown in Fig. 9. In this paper,
the Hypothesis Net (DNN1) makes hypotheses as actions
to write Chinese calligraphy, and the Memory Net (DNN2)
judges these hypotheses. Then the virtual robot can improve
the writing result by retrieving historical experiences. Instead
of using the mean squared error (MSE) of the image data as
the loss function, we use the MSE of the perception code of
the image as the loss function in (1). The proposed cognitive
system has the perception, the memory, and the hypothesis
models to enable the virtual robot to learn and write Chinese
calligraphy.

**A. PERCEPTION MODEL SETTING**

Image processing has been rapidly developed to solve many
problems to make robots more intelligent. Unfortunately,
images are very difficult to describe. Inspired by human
perception, we built neural networks for robot perception in
order to encode the writing of Chinese calligraphy to the
neural information, because humans do not process image
data directly, but covert the image to neural information in
the brain.

In \([8]\), a Writer Net split a stroke of calligraphy writing
into several \(20 \times 20\) region of interest (ROI) images to reduce
the complexity of the vision data. To integrate encoder with
Hypothesis Generation Networks (HGN) in \([8]\), we design an
Encoder in Fig. 10 with a sequence of multiple ROI images
to encode the image of Chinese calligraphy writing.

The architecture of the Encoder in the proposed perception
model in Fig. 10 is shown in Table 2. The Encoder is similar
to, but smaller than Writer Net in \([8]\). All of the nonlinear
activation functions in the convolution layer are ReLU \([48]\)
and all of the masks are $3 \times 3$ filters. Instead of a max pooling layer, we used average pooling to reduce the dimension. Calligraphy words are binarized images with filled shape. Therefore, average pooling can keep more information about the shape than max pooling. There are 13 layers in both Encoder and Decoder in the perception model, and they have a symmetric network architecture.

In order to generate the stroke data of Chinese calligraphy, we split the stroke to several ROI images. Figure 11 shows the ROI images of different strokes. Even though different strokes in the image domain are not similar, the pattern of writings is relevant. If we pick some of the ROI images, these images may also exist in other stroke images or appear in the future.

B. HYPOTHESIS NET (DNN1) AND MEMORY NET (DNN2) IN CHINESE CALLIGRAPHY LEARNING TASK

In [8], we have already designed a fundamental Hypothesis Generation Networks (HGN) in Chinese calligraphy learning tasks. In this paper, we proposed a cognitive system utilizing a 15-layer Writer Net [8] and a 16-layer Estimator Net [8] as the Hypothesis Net (DNN1) and Memory Net (DNN2), respectively. The difference between the Writer Net in [8] and the Hypothesis Net (DNN1) in this paper lies in the memory retrieval process shown in Fig. 12.

VII. SIMULATIONS

Considering that there are multiple neural networks (DNN1, DNN2, encoder, and decoder) in the cognitive system, sufficient memory space is required in the training stage (including gradient computation). Although the computational load of individual neural networks is not heavy, sufficient memory is nevertheless required to train the multiple networks at the same time. Table 3 presents the time complexity and space complexity of the proposed cognitive system. Since the cognitive system is online learning, the complexity shown in Table 3 refers to training rather than just testing. By setting the batch size to 5 in our experiments, the space complexity is about 13GB. This has imposed a difficult burden for most GPU devices to train the cognitive system. As a result, the Taiwan Computing Cloud (TWCC) is used to execute the practicing process of Chinese calligraphy writing, where two Tesla V100 GPUs each with 64GB memory are used to simulate the learning process of calligraphy writing.

TABLE 3. Complexity of the proposed cognitive system (in the training stage).

| Model Type       | Time complexity | Space complexity |
|------------------|-----------------|------------------|
| Hypothesis Model | Batch size X 119.6 GFLOPS | Batch size X 820 MB |
| Memory Model     | Batch size X 214 GFLOPS | Batch size X 1,630 MB |
| Perception Model | Batch size X 10.2 GFLOPS | Batch size X 172 MB |
| Total            | Batch size X 343.8 GFLOPS | Batch size X 2,622 MB |

To write Chinese calligraphy, a virtual calligraphy writing robot [8], [47] is chosen to train the proposed cognitive system. Fig. 14 shows the Chinese calligraphy writing simulation system [47], where an action vector controls a series of the control points to draw the writing results.

The Chinese calligraphy writing task is to learn 8 fundamental strokes shown in Fig. 15, including pecking, horsewhip, passing lightly, sideways, jump, dismemberment, bridle, and crossbow. These 8 strokes can be used to build up

**TABLE 2. Detailed architecture of the proposed stroke encoder.**

| Layer | Purpose | Filter/Stride | Output Size |
|-------|---------|---------------|-------------|
| 1     | Image Input | 20x20x1 |            |
| 2     | Convolution and ReLU | 3x3/1 | 20x20x32 |
| 3     | Convolution and ReLU | 3x3/1 | 20x20x32 |
| 4     | Convolution and ReLU | 3x3/1 | 20x20x32 |
| 5     | Average Pooling | 2x2/2 | 10x10x32 |
| 6     | Convolution and ReLU | 3x3/1 | 10x10x64 |
| 7     | Convolution and ReLU | 3x3/1 | 10x10x64 |
| 8     | Convolution and ReLU | 3x3/1 | 10x10x64 |
| 9     | Average Pooling | 2x2/2 | 5x5x64 |
| 10    | Convolution and ReLU | 1x1/1 | 5x5x64 |
| 11    | Dropout (50%) |         |            |
| 12    | LSTM | 1x1x512 |            |
| 13    | Fully Connected | 1x1x128 |            |
many kinds of Chinese words. That is, the virtual robot with the proposed cognitive system can write many Chinese words through learning these 8 fundamental strokes.

In this experiment, the dimensions of action space are $n_0 = 5$, indicating an action vector $a_0^k = [x, y, z, \theta, \bar{\theta}]^T$. $n_1$, the number of action points $a_1^k$ in an action vector $A_k$, is set to 123. The number of chosen memories to help consolidate the memory $n_2$ is set to 5. The length of the perception code is $n_3 = 128$. The size of the short-term memory should be less than the size of the long-term memory. We chose the size of memory space $n_4 = 35$ which consists of short term memory $l_s = 5$, positive memory $l_p^r = 5$, negative memory $l_n^r = 5$ and impressive memory $l_i = 20$.

We pre-trained the perception model with a few calligraphy images to speed up the learning process of the cognitive system. Adam optimization algorithm was utilized as the optimizer for all neural networks in the cognitive system. The learning rates of Switches $S_2, S_3$, and $S_6$ were set to $10^{-5}$ and all of them used the loss function in (1).

The simulation results are shown in Table 4 and Table 5 to evaluate the performance of the memory model and perception model for three methods, including the original hypothesis generation networks (original HGN) in [8], the hypothesis model with memory model (the proposed system without perception model), and the hypothesis model with memory model and perception model (the proposed system). Writing results of the three methods to practice the pecking, horsewhip, passing lightly, sideways, jump, dismemberment, bridle, and crossbow strokes 500 times are shown in Tables 4 and 5, respectively. With the memory model, the writing processes of the two proposed systems are more stable than the original HGN in [8]. These two proposed cognitive systems can remember past mistakes through retaining and consolidating memories to prevent making the same mistakes. With the loss function in (1), the loss curves of the writing processes are shown in Fig. 16 for (a) pecking, (b) horsewhip, (c) passing lightly, (d) sideways, (e) jump, (f) dismemberment, (g) bridle, and (h) crossbow, demonstrating the learning performance for Chinese calligraphy tasks. We utilized the three methods to compare their learning ability.

With reference to Fig. 16, the first method is the original HGN in [8] (in green) which does not have psychological memory processing (long term memory, short term memory and memory consolidation process) and the
perception model. The second method (in red) is the proposed system with the memory model which has psychological processing but doesn’t have the perception model. The third method (in blue) is the proposed system which has both of the memory model with psychological processing and the perception model. The original HGN in [8] sometimes converges to a local minimum (Fig. 16 (d) and (e)) because the memory in the HGN is unclear. With the memory model, the second and the third method are significantly more stable than the first method.

All Chinese words are composed of fundamental strokes in different positions, sizes and angles. In order to show the learning ability of the proposed cognitive system, we combined these eight strokes by linear transformation to draw a full word “yong” (永) shown in Table 6. As a result, we let the cognitive system learn the 8 fundamental strokes of calligraphy first. We then assign each stroke a specific position, size and angle through linear transformation whose parameters are determined through trial and error. Finally, we can complete drawing a full Chinese word with the parameters of these strokes. To allow a fair comparison, these methods are evaluated using the same linear transformation parameters.

Figure 17 shows the MSE loss curves of writing process “yong”, where the two proposed methods have good writing results, and both of these methods converge. The proposed system (in blue) (i.e., the third method) has a better learning ability (lower mean squared error) than the proposed system without perception model (in red) (i.e., the second method). Table 6 shows the writing results at epoch 1, 250 and 500, so that the final result of the two methods can be compared. Note that the writing results by the proposed system with perception model at epoch 500 is very similar to the human writing sample.

B. THREAT TO VALIDITY

The experiment in this paper is to apply the proposed cognitive system to a virtual robot which learns to write calligraphy in a simulated environment. Different from the real environment, there are fewer factors that will affect the validity since there are few uncertainties in the simulated environment. Nevertheless, there are two factors that may threaten the validity of the experiment results in this paper:

(i) Inappropriate hyperparameters:
Different from most deep learning research, the proposed cognitive system is internally composed of four individual deep neural networks. Therefore, the number of hyperparameters is at least 4 times more than that of a conventional deep learning method. In addition, the hyperparameters of each neural network affect each other. For example, the process of memory consolidation in the cognitive system involves both the networks of the memory model and perception model. Also, in the memory retrieval process of the cognitive system, the hypothesis network, the perception network, and the memory network are simultaneously involved. These cognitive processes are similar to human cognitive processes in which the memory, perception, and hypothesis generation all affect each other. In this paper, we set the learning rate to the same value to allow a fair comparison.

(ii) Inappropriate initial network weight values:
Since deep neural networks in the cognitive system have a large number of weights, the initial weights will affect the behavior of the cognitive system to a certain extent. Fortunately, as long as the learning rate is sufficiently small, the cognitive system can eventually converge stably during the learning process. That is to say, even if the initial state of the cognitive system is undesired, the cognitive system can complete the task through acquired experiences. Nevertheless, different initial weights generally lead to different convergence time, which may affect the validity of the experiment. In order to reduce the influence of the uncertainty of the initial weight values, we use Xavier Initialization [49] to initialize the weight values of the deep neural networks in the cognitive system. Xavier Initialization method serves to limit the initial weight values to a range that is not too large, so that the initial state of the deep neural network will not deviate too much.

C. CONTRIBUTION

The contribution of this method is to add memory model, and hypothesis generation model to the cognitive computing
**TABLE 4.** Writing results of pecking, horsewhip, passing lightly and sideway.

| Action            | Original HGN in [9] | The proposed system without perception model | The proposed system |
|-------------------|---------------------|---------------------------------------------|---------------------|
| **Pecking**       |                     |                                             |                     |
| Writing at 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch | at 410                         | at 330                      | at 475              |
| The best result in 500 epoch chosen by human | at 150                                      | at 325                      | at 475              |
| **Horsewhip**     |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch | at 170                        | at 495                      | at 445              |
| The best result in 500 epoch chosen by human | at 350                                      | at 185                      | at 445              |
| **Passing lightly** |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch | at 365                        | at 485                      | at 490              |
| The best result in 500 epoch chosen by human | at 495                                      | at 485                      | at 490              |
| **Sideway**       |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch | at 25                         | at 405                      | at 500              |
| The best result in 500 epoch chosen by human | at 25                                      | at 405                      | at 500              |
|                  | Original HGN in [9] | The proposed system without perception model | The proposed system |
|------------------|---------------------|---------------------------------------------|---------------------|
| **Jump**         |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch |                  |                                             |                     |
|                  | at 5                | at 445                                       | at 490             |
| The best result in 500 epoch chosen by human |          |                                             |                     |
|                  | at 30               | at 445                                       | at 490             |
| **Dismemberment** |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch |                  |                                             |                     |
|                  | at 155              | at 495                                       | at 485             |
| The best result in 500 epoch chosen by human |          |                                             |                     |
|                  | at 100              | at 500                                       | at 85              |
| **Bridle**       |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch |                  |                                             |                     |
|                  | at 425              | at 290                                       | at 250             |
| The best result in 500 epoch chosen by human |          |                                             |                     |
|                  | at 60               | at 290                                       | at 250             |
| **Crossbow**     |                     |                                             |                     |
| Writing in 500 epoch |                    |                                             |                     |
| The minimum MSE in 500 epoch |                  |                                             |                     |
|                  | at 265              | at 365                                       | at 400             |
| The best result in 500 epoch chosen by human |          |                                             |                     |
|                  | at 300              | at 365                                       | at 400             |
TABLE 6. Writing results of “Yong” (永).

| Human writing sample | Method                                      | EP:1 | EP:250 | EP:500 |
|----------------------|---------------------------------------------|------|-------|-------|
| Yong                 | The proposed system without perception model|      |       |       |
| 永                    | The proposed method                           |      |       |       |

which usually deals with the perception process only. The original HGN [8] has been able to learn action generation, but the convergence of its learning process is sometimes unstable. By adding the memory model and perception model to the original HGN, so that the HGN can converge stably. In addition to supervised learning through given training data, the ability to formulate hypotheses is also the focus of current and future research on artificial neural networks. The cognitive science combined with deep learning has great potential and may also serve as the fundamental part for the reasoning capabilities of robots in the future.

IX. CONCLUSION

Cognitive systems are an important area of research in the field of artificial general intelligence (AGI). According to research in modeling the learning processes of humans, robots could finally learn to solve different tasks via self-learning abilities. In this paper, we have presented the proposed cognitive system which can learn to solve tasks. By controlling switches, we can implement mental processes to perform different kinds of tasks. The simulation results show good learning abilities in generating complex motions by the proposed cognitive system for a virtual writing robot that learns to write Chinese calligraphy. The proposed cognitive framework can apply not only to Chinese calligraphy writing but also other different tasks. However, the neural networks in the proposed system (Hypothesis Net, Memory Net, encoder and decoder) are not universal. That is, different scenarios (other than Chinese Calligraphy writing) will require appropriate network architecture. Moreover, there are four artificial neural networks that interact with each other and many parameters have to be designed. Therefore, the cognitive system sometimes cannot produce good results because of the sub-optimal parameters or initial weights.

ACKNOWLEDGMENT

The authors are grateful to the National Center for High-Performance Computing for computer time and facilities to conduct this research.

REFERENCES

[1] M. Sassin, “Creating user-intended programs with programming by demonstration,” in Proc. IEEE Symp. Vis. Lang., May 1994, pp. 153–160.
[2] N. Delson and H. West, “Robot programming by human demonstration: The use of human variation in identifying obstacle free trajectories,” in Proc. IEEE Int. Conf. Robot. Autom., Sep. 1994, pp. 564–571.
[3] S. Park, W. Lee, W. K. Chang, and K. Kim, “Programming by demonstration using the teleimpedance control scheme: Verification by an sEMG-controlled ball-trapping robot,” IEEE Trans. Ind. Informat., vol. 15, no. 2, pp. 998–1006, Feb. 2019.
[4] F. Steinmetz, V. Nitsch, and F. Stulp, “Intuitive task-level programming by demonstration through semantic skill recognition,” IEEE Robot. Autom. Lett., vol. 4, no. 4, pp. 3742–3749, Oct. 2019.
[5] B. Abibullaev, A. Zollanvari, B. Sadauanov and T. Alizadeh, “Design and optimization of a BCI-driven telepresence robot through programming by demonstration,” IEEE Access, vol. 7, pp. 111625–111636, 2019.
[6] Kumar, A. Giagkos, P. Shaw, R. Braud, M. Lee, and Q. Shen, “Discovering schema-based action sequences through play in situated humanoid robots,” IEEE Trans. Cognit. Develop. Syst., vol. 14, no. 3, pp. 1021–1035, Sep. 2022.
[7] Kumar, P. Shaw, A. Giagkos, R. Braud, M. Lee, and Q. Shen, “Developing hierarchical schemas and building schema chains through practice play behavior,” Frontiers Neurorobot., vol. 12, pp. 1–20, Jun. 2018.
[8] W.-Y. Wang, M.-J. Hsu, L.-A. Yu, Y.-H. Chien, and C.-C. Hsu, “Deep learning-based hypothesis generation model and its application on virtual Chinese calligraphy-writing robot,” IEEE Access, vol. 8, pp. 87243–87251, 2020.
[9] H. Reynolds and S. Feldman. (2014. Cognitive Computing: Beyond the Hype. KM World. [Online]. Available: http://www.kmworld.com/Articles/News/News-Analysis/Cognitive-computing-Beyond-the-hype-97865.aspx
[10] M. Chen, F. Herrera, and K. Hwang. “Cognitive computing: Architecture, technologies and intelligent applications,” IEEE Access, vol. 6, pp. 19774–19783, 2018, doi: 10.1109/ACCESS.2018.2791469.
[11] A. K. Noor, “Potential of cognitive computing and cognitive systems,” Open Eng., vol. 5, no. 1, pp. 167–189, Nov. 2014.
[12] L. Ogiela, “Cognitive informatics in automatic pattern understanding and cognitive information systems,” in Advances in Cognitive Informatics and Cognitive Computing, vol. 323. New York, NY, USA: Springer, 2010, pp. 209–226.
MIN-JIE HSU was born in Taipei, Taiwan, in 1993. He received the B.S. degree in electrical engineering from the National Taiwan Normal University, Taipei, in 2015, where he is currently pursuing the Ph.D. degree with the Department of Electrical Engineering. His research interests include artificial intelligence, fuzzy logic systems, neural networks, and reinforcement learning.

YI-HSING CHIEN was born in Taipei, Taiwan, in 1978. He received the M.S. degree in electrical engineering from Fu-Jen Catholic University, Taipei, in 2007, and the Ph.D. degree in electrical engineering from the National Taipei University of Technology, Taipei, in 2012. He is currently a Postdoctoral Researcher with the Department of Electrical Engineering, National Taiwan Normal University, Taipei. His current research interests and publications include the areas of fuzzy logic control, robust adaptive control, machine learning, and neural networks.

CHEN-CHIEN HSU (Senior Member, IEEE) was born in Hsinchu, Taiwan. He received the B.S. degree in electronic engineering from the National Taiwan University of Science and Technology, Taipei, Taiwan, in 1987, the M.S. degree in control engineering from the National Chiao-Tung University, Hsinchu, in 1989, and the Ph.D. degree from the School of Microelectronic Engineering, Griffith University, Brisbane, Australia, in 1997.

He was a Systems Engineer with IBM Corporation, Taipei, for three years, where he was responsible for information systems planning and application development, before commencing the Ph.D. studies. He joined the Department of Electronic Engineering, St. Johns University, Taipei, as an Assistant Professor, in 1997, and appointed as an Associate Professor, in 2004. From 2006 to 2009, he was at the Department of Electrical Engineering, Tamkang University, Taipei. He is currently a Professor with the Department of Electrical Engineering, National Taiwan Normal University, Taipei. He is the author or coauthor of more than 180 refereed journal and conference papers. His current research interests include digital control systems, evolutionary computation, vision-based measuring systems, sensor applications, and mobile robot navigation. He is an IET Fellow.

HSIN-HAN CHIANG (Senior Member, IEEE) received the B.S. and Ph.D. degrees in electrical and control engineering from the National Chiao-Tung University, Hsinchu, Taiwan, in 2001 and 2007, respectively. He was a Postdoctoral Researcher in electrical engineering at the National Taipei University of Technology, in 2008 and 2009. From August 2009 to January 2017, he was an Assistant and Associate Professor at the Department of Electrical Engineering, Fu Jen Catholic University, New Taipei City, Taiwan. From February 2017 to January 2022, he was an Associate Professor at the Department of Electrical Engineering, National Taiwan Normal University, Taipei, Taiwan. Since February 2022, he has been an Associate Professor with the Department of Vehicle Engineering, National Taipei University of Technology, Taipei. His research interests include the area of intelligent systems and control, autonomous driving systems, vehicle dynamics and control, and autonomous mobile robots.

LI-AN YU received the B.S. and M.S. degrees in electrical engineering from the National Taiwan Normal University, Taipei, Taiwan, in 2015 and 2017, respectively. His expertise includes neural networks and computer vision. He was a Postgraduate Research Assistant at the Computational Intelligence Laboratory, National Taiwan Normal University, where he was involved in Artificial Intelligence design, in 2018 and 2020.