Biologically-Inspired Episodic Memory Model Considering the Context Information

Gyeong-Moon Park, Sanghyun Cho, and Jong-Hwan Kim, Fellow, IEEE
School of Electrical Engineering
KAIST
Daejeon, Republic of Korea
Email: {gmpark, scho, johkim}@rit.kaist.ac.kr

Abstract—Episodic memory can store time sequential events and retrieve them anytime with specific cues. However, if the episodic memory only stores events comprised of actions and objects, execution of episodes may fail if current situation is different from the settings it learned in. As a solution, we propose Deep C-ART (Context-Adaptive Resonance Theory) which considers not only time sequential events but also their contexts. In addition to the learning process of Deep ART, Deep C-ART stores context information such as situation of objects, states of robots, place, and time of episodes. Since context changes over each event in an episode, Deep C-ART forms an episode with an event sequence and a context sequence. During retrieval and execution of episode, it compares the current situation with the learned one to verify that it is executable or in an anomaly situation. The effectiveness of Deep C-ART is demonstrated through computer simulations.

Index Terms—Episodic memory, situated object, context, Deep C-ART.

I. INTRODUCTION

Episodic memory is about one’s past experiences while semantic memory stores general facts about objects, environment, or the world [1]. Thus, people may remember common features and details about an object but recall different personal experiences related to it. Episodic memory stores an episode in a form of temporal sequences of events. Using the episodes in the memory, human can fully demonstrate what she has learned or even infer or predict events based on them.

Using this concept of episodic memory, many learning models and memory systems were developed [2]–[5]. Based on the unsupervised neural network, Adaptive Resonance Theory (ART) [2], episodic memory-ART (EM-ART) was proposed to memorize time sequential events using the decaying factor [3], [4]. This parameter makes a geometric sequence between events, and the sequence becomes a new input for encoding an episode. For the robust inference of the target episode from events, and the sequence becomes a new input for encoding an episode. For the robust inference of the target episode from events, and the sequence becomes a new input for encoding an episode. For the robust inference of the target episode from events, and the sequence becomes a new input for encoding an episode.

As a solution, Deep C-ART (Context-Adaptive Resonance Theory) is proposed to consider both time sequential events and context information along with time sequential events, based on a newly designed architecture. Above all, the context field can have multiple activated nodes to encode a context that consists of objects with their features and situations, states of robots, spatial and temporal information. Then, the context field stores sequences of changing contexts in between action events. Second, during learning and encoding processes, it considers the context of before and after the task and focuses on what is actually changed and ignores the rest. If there are five objects on a table but only three of them are used in a task, the system stores only the events of those three objects which would make it more effective by pruning out unnecessary context.

Another advanced memory model, Deep ART, was proposed for learning episodic memory robustly [12]. Using its unique encoding and decoding processes, it can store and retrieve episodes without errors, which may occur in previous models. This robustness of the system is suitable to be used for robots to learn and execute specific tasks. A task can be an episode which is a sequence of action and object pairs. With Deep ART memory, a robot can learn and remember various tasks and successfully retrieve them from input cues. However, all these memory models have shortcomings which limit them to be actively used for robots in real life. They only concern about time sequential events related to specific objects, under an assumption that the episode would always be successfully retrieved regardless of current situation.

According to the article, human episodic memory is highly context dependent [13]. During episode encoding, items and contexts are bound together [14], [15]. It directly affects the retrieval process, so that humans can enhance memory performance when a current context is well matched with the learned one. With such reasons, we suggest that when the proposed episodic memory learns episodes, it should consider not only time sequential events but also context information of events for each episode. The memory model binds event and context sequences into one episode, and they are sequentially recalled in the retrieval process. By considering and comparing the current situation with the situation of the time it learned, we are able to successfully apply biological concept of human’s episodic memory, and enhance memory performance in various contexts.

To deal with these issues, in this paper, we propose Deep C-ART that has four main contributions. First, it can learn context information along with time sequential events, based on a newly designed architecture. Above all, the context field can have multiple activated nodes to encode a context that consists of objects with their features and situations, states of robots, spatial and temporal information. Then, the context field stores sequences of changing contexts in between action events. Second, during learning and encoding processes, it considers the context of before and after the task and focuses on what is actually changed and ignores the rest. If there are five objects on a table but only three of them are used in a task, the system stores only the events of those three objects which would make it more effective by pruning out unnecessary context. Third, during retrieval, it can combine event sequences and context sequences appropriately so that the output sequence will have events and contexts alternately in correct order. Lastly, during the execution, it can detect anomaly situation based on the context it learned. If the current
context is different from expectation, it will tell that it is inexcusable or even skip the step. The effectiveness of Deep C-ART is demonstrated with simulation results.

This paper presents the proposed Deep C-ART in the following order. Section II describes Deep ART as a background of our proposed memory model, and Section III presents the proposed model in detail with descriptions of each process. Section IV presents the simulation results, and concluding remarks follow in Section V.

II. BACKGROUND

A. Deep ART

To model bio-inspired episodic memory, Deep ART neural model was proposed [12]. Based on unsupervised neural network, it can categorize events and learn high-level features like episodes incrementally by building up deep structure. To categorize inputs concretely, Deep ART has attribute field in the bottom layer, which gets semantic information for each input. The attribute field can specify the inputs with their features through multi-channel system which can get various inputs at once and categorize them. Using the specified inputs, Deep ART encodes events and their temporal sequences using the recurrent concept. The equation of time encoding process is as follows:

\[ \bar{y_i} = \frac{1}{M} \sum_{k=0}^{n-1} b_i^k x_{i-k} \]

where \( b_i^k \), \( x_{i-k} \), and \( \bar{y}_i \) are input, buffer, and output vectors, and \( n \) is the maximum positional number among all elements of the vector. This encoding process is different from those of previous models [3], [4], which memorize relative sequences between events. They can result in encoding error when an episode has the same duplicate events. However, with its distinctive encoding process, Deep ART can memorize any elements of the vector. This encoding process is different from expectation, it will tell that it is inexcusable or even skip the step. The effectiveness of Deep C-ART is demonstrated with simulation results.

This paper presents the proposed Deep C-ART in the following order. Section II describes Deep ART as a background of our proposed memory model, and Section III presents the proposed model in detail with descriptions of each process. Section IV presents the simulation results, and concluding remarks follow in Section V.

III. DEEP C-ART

Deep C-ART is an advanced memory model based on Deep ART network, to learn both sequential events and context information. The existing model, Deep ART, memorizes time sequential events without considering their contexts. Thus, this memory model is not adaptive to different environments. It is critical for the memory system to be used in various situations. Thus, as a solution, we propose a new memory architecture that can learn contexts continuously changed by the time sequential events. Since the proposed architecture considers contexts when robots perform the learned episodes, it can adapt to various circumstances. The overall architecture of the proposed model is shown in Fig. 1. First, we define key terms used in this paper and then explain the learning process of Deep C-ART in details.

A. Representations

In this paper, event is a verb clause represented by a form of tuple as follows:

\[ V_n = \{ t_n, 1_n, \gamma_n, z_n, e_{n} \} \]

where \( t_n \) denotes the order of event, \( v_n \) is the action, \( 1_n \) and \( 2_n \) are objects, and \( \gamma_n \) is a relation between two objects. Agents, like robots, recognize an action with manipulated objects continuously, and categorize an event \( V_n \) by a unsupervised manner, using their episodic memory.

Next, situated object is a currently recognized object, which contains its attributes and situations. Object’s attributes are such as name, color, and shape, and situations are conditions of object. Thus, the situated object (\( O_i \)) can be represented as follows:

\[ O_i = \{ \text{name}_i, \text{color}_i, \text{shape}_i, \text{type}_i, \text{pos}_i, \text{dist}_i, \text{cont}_i \} \]

where \( i \) is the order of context, \( n \) and \( x \) are the order of context, \( \text{name}_i \), \( \text{color}_i \), \( \text{shape}_i \), and \( \text{type}_i \) are attributes of object, which mean name, color, shape, and type, respectively, and \( \text{pos}_i \) contains position, orientation, distance between object and hand, and contents, respectively. Generally, there can be various situated objects in a current environment. Each situated object is an input to the Deep C-ART memory model.

Context is comprised of several situated objects that are recognized in the current environment, states of robots, place, and time. Thus, the context (\( C_m \)) is represented by

\[ C_m = \{ O_1, O_2, \ldots, O_{m} \} \]

where \( j \) is the number of robot’s states, \( t_{j} \) is the \( j^{th} \) state of robot, such as gripper information, \( \text{place}_{1} \) is the current location, and \( \text{time}_{1} \) is the current time. These inputs are memorized by the proposed memory model to distinguish the present context.
Episode is a mixed sequence of events and contexts. Since an event has a precondition and postcondition, i.e., effect, the event $V_n$ always has two contexts, such as

$$C_{n-1} \rightarrow V_n \rightarrow C_n$$

(6)

where $C_{n-1}$ is the precondition of the event $V_n$, and $C_n$ is the effect of the event $V_n$. Therefore, the episode ($E$) constituted by time sequential events can be represented by

$$E = C_0 \rightarrow V_1 \rightarrow C_1 \rightarrow \ldots \rightarrow V_n \rightarrow C_n$$

(7)

where $C_0$ denotes the initial context. From this equation, the number of contexts is determined by $(n + 1)$, when the number of events is $n$. Since an episode is a high-level concept constituted by contexts and events, the memory model should have a deeper structure for encoding the episodes.

**B. Event Encoding Process**

The event encoding process of Deep C-ART is the same as that of Deep ART. Deep C-ART gets continuous events as inputs through the attribute field which has multi-channels from $^1F_1$ to $^3F_1$ as in Fig. 1. The attribute channels for action inputs get sequential actions, relations and modifiers for each action, and those for two objects get objects with their attributes to construct an event.

Using Eq. (1) of Deep ART, Deep C-ART encodes the sequence of input events recursively. The output channel $^4F_1$ in the event field $F_1$ finally memorizes the whole sequence of input events. The output vector $^4y_1^n$ is then normalized to $^4y_1^n$ by Eq. (2).

**C. Context Encoding Process**

To memorize changes of contexts during the event learning process, a new structure for learning time sequential contexts is designed. The enlarged view of the new structure is depicted in Fig. 2.

This structure has an attribute field to get a current context, and continuous input contexts are sequentially encoded through three channels which are input, buffer, and output channels. Finally, the output vector representing the sequence of contexts is changed to a new input vector with post-processes of normalization and deleting backgrounds. The details of each process are illustrated below.

1) **Encoding the sequence of contexts:** To constitute a context input, currently recognized situated objects should be encoded. The nodes representing situated objects are connected to multi-channels from $^4F_1$ to $^4F_4$ to get attributes and conditions of objects. Since there may be several objects in the current context, the input channel $^4F_2$ can have several activated nodes that represent situated objects. It is different from the conventional ART networks, which have only one node with the activation value of one and all the other nodes of zero in the category field. It is called winner-take-all strategy.

In addition, states of robots, place, and time should be encoded with those situated objects, and they are comprised of single vector for each context. In Fig. 2, each channel $^4F_3$ gets a single vector to represent states, place, and time. Therefore, to categorize the context input in the input channel $^4F_2$ from the attribute field, the activated nodes representing the situated objects are augmented with other nodes that stand...
for the states, place, and time.

After encoding the input vector for the current context, the time sequence of contexts is memorized using the encoding process of Deep ART. Even if the activated nodes are multiple to constitute a context, the encoding process of Deep ART can be adopted without changes. It is another advantage of this encoding process because the conventional encoding process assumes that only one node is activated at once. Finally, the output channel $F_{E}^{i}$ has the output vector representing the sequence of contexts.

2) Deleting unrelated information: If the proposed memory structure in Fig. 2 memorizes all context information, unrelated backgrounds with the current episode would also be encoded together. It is critical when executing the learned episode again. For example, the unrelated objects that were stored but not present after would make execution fail, even if all other related objects were present. To avoid these situations, a filtering process is needed to remove unrelated information, after encoding the sequence of contexts.

In the encoding process of Deep ART, unchanged inputs are always activated, so the activation value is kept to be one from the first to the last context because of the winner-take-all strategy. Therefore, unchanged nodes should have the following maximum value from Eq. (1):

$$y_{\text{max}} = \max \left\{ y_{w} \mid w \in \mathbb{N} \right\}$$

If the activation value of node is the same as $y_{\text{max}}$, this node can be regarded as unrelated contexts with the current episode. Thus, the activation values of all nodes that have the same value of $y_{\text{max}}$ are set to zero in order to delete the unrelated information. As a result, the input vector $F_{E}^{i}$ is needed to have any unnecessary contexts. The output vector is then normalized using the maximum positional number, as in Eq. (2).

D. Episode Learning

Deep C-ART model memorizes both event sequence and context sequence simultaneously for each episode. In order to encode the episode using two sequences, a categorization technique grouping different types of inputs into one category is needed. For this purpose, Fusion ART network is used to encode an episode from both sequences. Fusion ART can get different types of inputs in multiple input channels, which is a main difference from Fuzzy ART [2]. Therefore, the function of Fusion ART enabled Deep C-ART to learn a new episode using two unrelated output vectors, and finally the episode is categorized in the input channel $F_{E}^{i}$.

E. Episode Retrieval

After learning all episodes, Deep C-ART can retrieve the entire sequences of events and contexts for the selected episode.

In the input channel $F_{E}^{i}$, the $i$th episode can be retrieved by substituting the weight vector for the input vector, such as

$$x = \bar{x}_{a} = \begin{cases} \bar{x} & y_{\text{max}} = \max \{ y_{w} \mid w \in \mathbb{N} \} \\ \hat{x} & y_{\text{max}} = \min \{ y_{w} \mid w \in \mathbb{N} \} \end{cases}$$

where $\bar{x}_{a}$ is the weight vector connecting the input channel $F_{E}^{i}$ and the multi-channels $F_{E}^{i}$ and $F_{E}^{i}$. As already explained in Eq. (7), an episode is made of sequential events and contexts, and the orders of them are important in the retrieval process. Considering the orders of them and several situated objects, each input vector $\bar{x}_{a} = \{ \bar{x}^{i}, \bar{x}^{i} \}$ for $e = 0, \ldots, 2n$ and $i \in \mathbb{N}$ can be retrieved from the retrieval process as follows:

$$e = 2i \rightarrow (k = e/2) \land (A = V)$$
$$e = (2i - 1) \rightarrow (k = (e - 1)/2) \land (A = C)$$

$$y_{E,a,k} = \begin{cases} \bar{x} & y_{w} = M_{x}^{i}, \text{if } k = 0 \\ \hat{x} & y_{w} = 0 \end{cases}$$

$$E_{a, k} = \max \left\{ y_{w} \mid w \in \mathbb{N} \right\}$$

$$F_{E, k} = \max \left\{ y_{w} \mid w \in \mathbb{N} \right\}$$

$$i_{x} = \begin{cases} \bar{x}_{a} & e = 2i \land (A = V) \\ \hat{x}_{a} & e = (2i - 1) \land (A = C) \end{cases}$$

F. Anomaly Detection

After retrieving whole sequences, Deep C-ART predicts all multi-channels $F_{E}^{i}$ and $F_{E}^{i}$ and the multi-channels $F_{E}^{i}$ and $F_{E}^{i}$. As already explained in Eq. (7), an episode is made of sequential events and contexts, and the orders of them are important in the retrieval process. Considering the orders of them and several situated objects, each input vector $\bar{x}_{a} = \{ \bar{x}^{i}, \bar{x}^{i} \}$ for $e = 0, \ldots, 2n$ and $i \in \mathbb{N}$ can be retrieved from the retrieval process as follows:

$$e = 2i \rightarrow (k = e/2) \land (A = V)$$
$$e = (2i - 1) \rightarrow (k = (e - 1)/2) \land (A = C)$$

$$y_{E,a,k} = \begin{cases} \bar{x} & y_{w} = M_{x}^{i}, \text{if } k = 0 \\ \hat{x} & y_{w} = 0 \end{cases}$$

$$E_{a, k} = \max \left\{ y_{w} \mid w \in \mathbb{N} \right\}$$

$$F_{E, k} = \max \left\{ y_{w} \mid w \in \mathbb{N} \right\}$$

$$i_{x} = \begin{cases} \bar{x}_{a} & e = 2i \land (A = V) \\ \hat{x}_{a} & e = (2i - 1) \land (A = C) \end{cases}$$

where $M$ is the maximum positional number for the de-normalization process, $L_{E, k}^{a}$ is the index of the maximum value of $y_{E,a,k}$, and $k_{E}^{a}$ is the number of remained events to be retrieved. Using the equation above, each event and context are retrieved sequentially, following Eq. (7). After retrieving the input vectors in the input field $F_{E}^{i}$, attribute inputs can be recalled recursively using Eq. (9).

IV. SIMULATIONS

In this section, the effectiveness of Deep C-ART is demonstrated by simulations. First, we provided episodes consisting of events and contexts as a dataset to Deep C-ART for learning. Then, we tested the model whether it could recall...
the episodes considering the current context properly. By comparing with the previous memory model, Deep ART, the anomaly detection capability was also demonstrated.

A. Simulation Setup

For simulation, we used three episodes. The lists of events of the episodes are enumerated as follows:

- **Arrange a red toy**: Approach a red toy on a table, Grasp the red toy, Move the red toy to Box A, Release the red toy on Box A,
- **Make cereal**: Approach the milkbox on the table, Grasp the milkbox, Move the milkbox to the bowl, *Tilt* the cereal to the bowl, Release the cereal to the bowl, *Tilt* the cereal to the bowl, Release the cereal on the table,
- **Toast a slice of bread**: Approach bread on a dish, Grasp the bread, Move the bread to a toaster, *Release* the bread in the toaster, Push a lever of the toaster, *Take out* the bread from the toaster, Release the bread on the dish.

The context information for each episode was also defined from the initial to the final context. In this research, for simplicity, we did not consider robot’s states except grippers and time information during the simulations. Therefore, the context input from Eq. (5) was as follows:

\[
C_n = \{O^*_1, O^*_2, \ldots, O^*_n, \#p_n, \text{place}_n\} \tag{11}
\]

where \(\#p_n\) is the gripper state of the robot. As already explained above, the number of contexts was automatically determined by the number of events of each episode. The numbers of situated objects used for each episode were four, eight, and six. To check the performance of context learning process, the situated objects included some unrelated objects with learned episodes.

B. Simulation Results

For comparison, we used both Deep C-ART and Deep ART models to learn the above scenarios. However, the inputs for Deep ART did not have any context information. We tested performances of two models using the input cues that were fully and partially given. Moreover, the simulations were performed in various contexts for testing the anomaly detection capability. The simulation results were described below.

1) **Episode retrieval from full-length inputs**: To compare the retrieval results of both memory models, we gave the memory models inputs with full sequences. Since the input completely matched the node representing the target episode, they could retrieve the target episode correctly. Let the target episode be ‘arrange a red toy’. In case of Deep ART, there were no attribute inputs for context information. For this reason, Deep ART recalled the following sequential events only:

\[
E_D = V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4 \tag{12}
\]

where \(E_D\) means the retrieval result of Deep ART. However, Deep C-ART retrieved the following sequential events with context sequences:

\[
E_C = C_0 \rightarrow V_1 \rightarrow C_1 \rightarrow \cdots \rightarrow V_4 \rightarrow C_4 \tag{13}
\]

where \(E_C\) means the retrieval result of Deep C-ART. The retrieval results of both models are shown in Table I. Here, the point was each context had two situated objects, the red toy \((O^*_n)\) and Box A \((O^*_A)\). When Deep C-ART learned this episode, there were four situated objects from the simulation setup. However, the unrelated object, like bottle, was filtered automatically because Deep C-ART deleted this node value during the context learning process by Eq. (8). It demonstrated the performance of the model which could successfully filter out unrelated objects during the context learning process.

From the above two equations, Deep C-ART retrieved context sequences to perform the task. It can make the robot adaptive to different contexts. Several examples to show the adaptiveness of the robot were illustrated in the following.

<table>
<thead>
<tr>
<th>Episode Retrieval Results: Arrange a red toy</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Deep ART</strong></td>
</tr>
<tr>
<td>(C_0) = {#O^<em>_1, #O^</em>_2, {\text{open}, \text{open}}, \text{room}}</td>
</tr>
<tr>
<td>(V_1) = Approach red toy on table</td>
</tr>
<tr>
<td>(V_2) = Grasp red toy</td>
</tr>
<tr>
<td>(V_3) = Move red toy to Box A</td>
</tr>
<tr>
<td>(V_4) = Release red toy on Box A</td>
</tr>
<tr>
<td>(E_D = V_1 \rightarrow V_2 \rightarrow V_3 \rightarrow V_4)</td>
</tr>
</tbody>
</table>

2) **Episode retrieval from partial inputs**: Here, we gave only the context input \(C_0\), which had recognized situated objects at that time. In Deep ART, it could not deal with context information, so it only got object inputs with their attributes. Then, Deep ART retrieved the episode that was activated the highest from the recognized objects. However, since there could be various situations for each episode, the robot might not be able to perform the task retrieved by Deep ART properly, though the appropriate task episode was recalled.

On the other hand, Deep C-ART considers the current situation based on learned contexts. There can be two cases: Deep C-ART may find a context memorized matching with the current context input \(C_0\), or it may not be able to find any.

There are several examples for the first case. When the robot got the context \(C_0\) that a red toy was already located in Box A, the robot did not perform any action. Similarly, when a slice of bread had different color from the original one, which means it was already toasted, the robot just served it without toasting again. Also, when the bowl was filled with milk, the
robot just poured the cereal to the bowl to complete this task. The retrieved actions from both memory models were listed in Table II. It showed that Deep C-ART could adapt different situations, as it learned events with corresponding contexts. Furthermore, this search-based system can be applied to human-robot interaction, as it helps the robot to cooperate with humans. When the robot performs an action from retrieved sequences, it predicts the next context from the previous action. If users help the robot as they do next procedures, it predicts the next context from the previous human-robot interaction, as it helps the robot to cooperate with situations, as it learned events with corresponding contexts. The robot just poured the cereal to the bowl to complete this task.

If the context input $C_i$ was different from learned contexts, Deep C-ART did not retrieve any learned sequence. It means the robot perceived the current situation was not suitable to perform the task. For instance, when there were not any situated objects for the task, the robot decided to find these objects instead of executing the learned actions. Or, when the cereal box was empty, the robot did not try to make cereal. Instead, it should request a new cereal box to the user. Finally, if the robot’s states were not matched with the current context, then the robot did other actions, like opening its grippers to make them empty. From the above examples, it was shown that Deep C-ART could retrieve the proper action matched with the current context robustly, and it could help the robot’s decision making in anomaly situations.

V. CONCLUSION

We proposed a new memory model, Deep C-ART that stores not any situated objects for the task, the robot decided to find these objects instead of executing the learned actions. Or, when the cereal box was empty, the robot did not try to make cereal. Instead, it should request a new cereal box to the user. Finally, if the robot’s states were not matched with the current context, then the robot did other actions, like opening its grippers to make them empty. From the above examples, it was shown that Deep C-ART could retrieve the proper action matched with the current context robustly, and it could help the robot’s decision making in anomaly situations.

### TABLE II

<table>
<thead>
<tr>
<th>Episode</th>
<th>Current Context ($C_i$)</th>
<th>Retrieved Action</th>
</tr>
</thead>
<tbody>
<tr>
<td>Arrange a red toy</td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td></td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td></td>
<td>$V_1$</td>
<td>No</td>
</tr>
<tr>
<td>Make cereal</td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td></td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td></td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td>Toast a slice of bread</td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td></td>
<td>$V_1$</td>
<td>$V_1$</td>
</tr>
<tr>
<td></td>
<td>$V_1$</td>
<td>No</td>
</tr>
</tbody>
</table>

### REFERENCES