Spiking Neural Network Predicting Temporal Sequence of 2D-patterns

Kohmei SHIOMI, Yuki UMEZAKI, Akane SAITO, and Makoto IWATA
School of Information, Kochi University of Technology,
Kami, Kochi, 782-8502 Japan

Abstract— *In recent years, the edge-side computation at the IoT devices is offering more intelligent functionality to adopt modern deep learning techniques. Most of input/ output data of the IoT devices might be sometimes noisy or error-prone due to lowering the cost of IoT devices and sensors. However, the existing deep neural networks (DNN's) are not good at learning and recognizing such temporal sequences of noisy data.*

Therefore, this study aims to propose a novel spiking neural network to learn and infer those kinds of data at the IoT devices. The proposed neural network is called the Boltzmann machine with single spike axon (SSA-BM), where we introduce the probability distribution of global states based on the Boltzmann machine and synaptic currents of neurons based on the leaky integrate and fire model. The evaluation results revealed that the proposed network predicts temporal sequence of 2D-patterns even if the input sequence involves some exceptions such as timing deviation or frame loss.

Keywords: SSA-BM, DyBM, STDP, LIF model neuron

1. Introduction

With the proliferation of Internet of Things (IoT) devices, the edge-side computation at the IoT devices is offering higher functionality to adopt modern artificial intelligence techniques like deep learning. This is because the cloud centric systems dealing with trillions of sensor IoT devices might be facing the overload in the near future. In the IoT device, sensors and actuators blend seamlessly with the environment around us in order to measure, infer, and understand the environmental conditions. Most of those data are streams of data, e.g., video and audio stream, a sequence of sensor signals, and so on, and those data might be sometimes noisy or error-prone due to lowering the cost of IoT devices and sensors. However, the existing deep neural networks (DNN's) are not good at learning and recognizing such temporal sequences of noisy data. Therefore, this study aims to propose a novel spiking neural network to learn and infer those kinds of data at the IoT devices.

Typical deep neural networks (DNN's) such convolutional neural network (CNN) [1], [2], [3] and autoencoder (AE) [4] reproduce the function of the animal's brain cells, but in most cases the treatment of the actual neuron membrane potential and spike signals is omitted. Therefore, the usual DNN's only deal with discretized information. As a computation model for continuous information, IBM Corp. has proposed dynamic Boltzmann machine (DyBM) [5]. DyBM is a kind of spiking neural network (SNN) [6], [7] which faithfully reproduces the behavior of neural cells and propagates information using spike signals in contrast with the traditional artificial neural network (ANN).

However, the spike timing handled by the DyBM is binary, i.e., whether the neuron fires or not within the predefined time window. For this reason, it cannot process input signals at an arbitrary timing and has the drawback of not having robustness. In other words, it can be said that it is difficult to detect when the input pattern has a temporal error or fallout. As a result, there are concerns that this might be a serious problem, e.g., in case of that an IoT application with vulnerable sensors is stably operated under harsh environment.

In this paper, we propose a spiking neural network with high prediction rate for normal sequence of data and with robustness against abnormal one including exceptions, which we call Boltzmann machine with single spike axon (SSA-BM). SSA-BM is a fully-coupled spiking recurrent neural network (RNN) whose neurons behave based on leaky integrate and fire (LIF) model and spike timing dependent plasticity (STDP) learning model [8]. Attractive features of SSA-BM is to employs a novel axon model that allows only a single spike within the time window and it adopts a fastest response decoder. By virtue of these features, we aim to achieve robust learning and reasoning of spatiotemporal patterns and this is revealed through applying SSA-BM to a prediction problem for a sequence of 2D-patterns.

This paper is organized as follows. The next section describes some basic neural models, i.e., LIF, Boltzmann machine, and DyBM. Those are introduced in SSA-BM. Then, pros and cons of those models in terms of spatiotemporal sequence learning and reasoning are discussed. Section 3 proposes SSA-BM through the explanation of distinctive neural behavior and spike-based information coding strategy. Section 4 describes evaluation results on the prediction problem for a sequence of 2D-patterns. Finally, section 5 concludes the paper.

2. Related Neural Models

The proposed SSA-BM is a neural network that introduces biological neurons into the traditional ANN with recursive structure. Specifically, we adopt the LIF model which is one of biological neural models, Boltzmann machine which minimizes the whole energy of the network to derive the optimal solution, and DyBM which is a kind of spiking and recurrent neural network incorporating the LIF model.

2.1 Leaky Integrate and Fire Model

There are many types of biological neural models, e.g., the firing-rate model, leaky integrate-and-fire (LIF) model, Hodgkin-Huxley model, Multi-compartment model, and so on.

The LIF model is one of the electrical input-output membrane voltage models, where the membrane potential V_m is calculated with neuronal membrane current and the leak term reflecting the diffusion of ions through the membrane. Once the membrane voltage of a neuron reaches a constant threshold V_{th}, a delta function spike occurs and the voltage is reset to its resting potential voltage. In this model, the time derivative of the membrane voltage dV_m/dt is formulated as follows.

$$\frac{dV_m}{dt} = \frac{-(V_m - E_L)}{\tau_m} + \frac{I_{syn}(t) + I_e}{C_m} \tag{1}$$

where E_L denotes the resting membrane potential, τ_m represents the membrane time constant, C_m is the capacitance of the cell membrane, and I_e is the neuronal membrane current. This model calculates the synaptic current by spike signals from another neuron. $I_{syn}(t)$ is the sum of the current values by $alpha$. Here, based on alpha-function-shaped post-synaptic current in LIF model, the I_{syn} and $alpha$ currents can be calculated as follows.

$$I_{syn}(t) = \sum_j w_j alpha(t - t_j) \tag{2}$$

$$alpha(t) = \frac{et}{\tau_s} exp(-\frac{t}{\tau_s}) \cdot Heaviside(t) \tag{3}$$

where t_j represents the arrival time of the j-th spike signal. w_j represents the synaptic weight of the connection through which the spike at time t_j arrived. e is the Euler's number and τ_s represents the time it takes for the current value by the $alpha$ function to be maximized, i.e., $Heaviside(t)$ represents the Heaviside step function $Heaviside(t)$ = 1 for t > 0 and $Heaviside(t)$ = 0 else.

2.2 Boltzmann Machine

The Boltzmann machine propagating binary data is considered to be a model compatible with neurons such as LIF that use spike signals for information transmission. Moreover, since it has a recursive structure, it is possible to handle time series patterns. Therefore, by combining with SNN, learning and prediction of spatiotemporal pattern will be possible.

The Boltzmann machine has a structure similar to that of a Hopfield network in which each neuron in the network is coupled to each other. Also, some of the connections between neurons have a recursive structure that is a link to itself. On the other hand, binary data is determined by stochastic dynamics, unlike statistical dynamics such as Hopfield network. That is, instead of setting the value obtained from the expression 4 as the output value from the neuron, it is used as the probability that its output value is 1. This makes it possible to sample a binary data vector with a small error function value.

$$P_k = \frac{1}{1 + exp(-\frac{\Delta E_k}{T})} \tag{4}$$

T of the expression 4 corresponds to the temperature in the neural network, which indicates the temperature of the whole neural network. When this temperature is high, the solution violently fluctuates. Even if it falls into a local solution, it is possible to get out of it. Conversely, when the temperature decreases, the fluctuation becomes small and it is possible to stay at the optimum solution. Specifically, in $T \to \infty$, P_k is always 0.5 regardless of the value of ΔE_k. In $T \to 0$, depending on whether ΔE_k is positive or negative, P_k is determined to be 0.0 or 1.0. Thus, the Boltzmann machine learns to generate binary data vectors with a high probability in the learning stage.

The Boltzmann machine probabilistically determines the value to be propagated from the external input and the state of the current neuron. Therefore, the energy representing the state of the neuron inside the Boltzmann machine varies depending on the input received from the outside. That energy is defined by the equation 5.

$$E = -\frac{1}{2} \sum_{ij} w_{ij} s_i s_j - \sum_i (\eta_i - \theta_i) s_i \tag{5}$$

where i and j represent the *i-th* and the *j-th* neuron, respectively, and s has a value of 0 or 1. η_i represents the external input to neuron i, w_{ij} represents the weight between neuron i and neuron j, and θ_i represents the threshold.

Also, ΔE_k in the equation 4 represents the total energy difference caused by a certain neuron k taking true or false value as follows.

$$\Delta E_k = \sum i w_k s_i + \eta_k - \theta_k \tag{6}$$

In this way, the Boltzmann machine changes the state of each neuron in response to an external input, and obtains the probability of whether the signal propagated from the variation is 1 or 0. Then, data is propagated according to the probability. Since the state of the neuron changes whenever an input is received from the outside, it is possible to return the output based on the past inputs. Therefore, it is possible to handle time series patterns. In other words, it can be said that it can deal with input patterns with temporal errors and information loss.

2.3 DyBM (Dynamic Boltzmann Machine)

The dynamic Boltzmann machine (DyBM) is a neural network introducing elements of spiking neural network (SNN) to the Boltzmann Machine. It deals with information of true or false from Boltzmann Machine as a spike signal. In DyBM, the neuron generates spikes when a neuron outputs true, i.e., it is equal to 1 or more than 0.5, and the neuron generates no spike otherwise. The DyBM adopts the spike timing dependent plasticity (STDP) that makes possible learn the strength of synaptic connections between neurons. In the STDP learning model, long-term potentiation or long-term depression of the synapse is determined along with relative temporal order of the arrival time of the pre-synaptic neuron's spike and the firing time of the post-synaptic neuron.

In addition, the DyBM adopts a FIFO (first-in first-out) queue that causes conduction delay between the neurons. After the pre-synaptic neuron fires, a spike signal at a post-synaptic neuron is delayed the arrive for the length of the FIFO queue that is randomly set. The arriving spike signal continues to influence the post-synaptic neuron permanently as well as the arrival timing. The influence amount gradually decreases. The connection between each neuron is not one but plural, and the DyBM takes the structure of Spike Prop which learn different weights respectively. The biggest feature of the DyBM is that it is possible to store a time series patterns. Therefore, the DyBM learns a time series patterns so that the output for input data from the outside at a time t is equal to the input data at the next time $t+1$. By repeating this learning, it is possible to output all the time series patterns learned as output without inputting all the time series patterns.

Each element such as STDP and Spike Prop mentioned in this chapter is adopted also in the SSA-BM proposed in this research, so it will be explained together in the next chapter.

3. Boltzmann Machine with Single Spike Axon: SSA-BM

SSA-BM proposed in this paper is a recurrent spiking neural network incorporating a biological neuron model and a Boltzmann probability distribution. It is well known that the spike timing dependent plasticity (STDP) is a biological process that adjusts the strength of connections between neurons in the brain. We introduce it as a learning method for SSA-BM.

3.1 Basic Structure of SSA-BM

A conventional Boltzmann machine is a neural network that operates on the basis of the probability. The probability P of true or false (1 or 0) is determined from the energy difference ΔE to determine the output. Compared to the biological neuron, this determination digitizes biological motion of the actual nerve cell. Consequently, the arrival timing of the spike signal is omitted so that the irregular timing spikes cannot be dealt with in this model. This disadvantage makes it difficult to apply the neural network to the inference problem of error-prone signals in the inexpensive IoT devices. Therefore, if we increase temporal resolution of those spike timing in the model, we could take care of such irregular timing spikes. Fig. 1 shows a basic structure of the SSA-BM network.

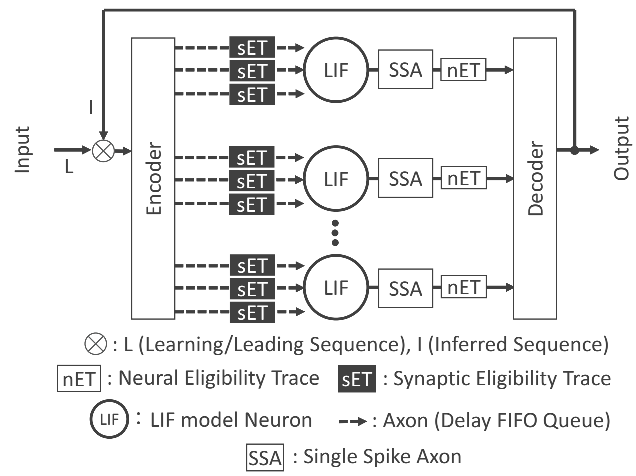

Fig. 1: Basic structure of SSA-BM.

As shown in this figure, the SSA-BM receives a sequence of binary data, e.g., a frame sequence of a moving picture, and produces its consecutive sequence of binary data. Thus, we intend to apply the SSA-BM model to inference or error correction of various sequence of sensory signals or IoT sensor signals. It operates as long as an input sequence of binary patterns is given from the outside. Each neuron stores neural eligibility traces (nET) and synaptic eligibility traces (sET). Each synapse or each pair of neurons that are connected via FIFO queue has the weight of long term potentiation, LTD weight, and the weight of long term depression (LTD weight).

The SSA-BM operates at higher resolution of time Δt than the period of input binary data ΔT. When simulating the SSA-BM, ΔT is treated as a single time window and its discrete event simulation is performed at an interval Δt ($\ll \Delta T$). The smaller the value of Δt with respect to ΔT, the higher the accuracy of the simulation. The encoder shown in the figure translates an input binary data into a set of spikes along with their input timing. The decoder translates a set of spikes within a single time window into an output binary data. During ΔT, the energy of each neuron is associated with the sum of alpha-shaped synaptic current and the output spike timing of the neuron is determined based on its spiking probability. After that, only the fastest output spike of the neuron is propagated to its post-synaptic neurons. This is a function of single spike axon shown in the figure.

3.2 Neuron Model in SSA-BM

The membrane potential V_m of a LIF neuron is calculated by the equation 1. When the membrane potential V_m exceeds the threshold value, the neuron ignites. Thereafter, V_m drops to the resting membrane potential. The point different from the original LIF model is the synaptic current I_{syn}. In SSA-BM, $I_{syn}(t)$ of a certain post-synaptic neuron j derived from N presynaptic neurons is defined in equation 7.

$$\begin{aligned} I_{syn}(t) = -b_j {x_j}^t - \sum_{i=1}^{N}\sum_{k=1}^{K} u_{i,j,k}\alpha_{i,j,k}^{t-1}x_j^t \\ + \sum_{i=1}^{N}\sum_{k=1}^{K} v_{i,j,k}\beta_{i,j,k}^{t-1}x_j^t \\ + \sum_{i=1}^{N}\sum_{l=1}^{L} v_{j,i,l}\gamma_{i,j,l}^{t-1}x_j^t \end{aligned} \tag{7}$$

where b_j represents a bias of neuron j and x represents an input spike. u and v represent a LTP weight and a LTD weight. Each weight u and v is updated based on the STDP process. LTP is used to learn to enhance coupling between neurons and LTD is used to learn to suppress binding. α represents a synaptic eligibility trace. β represents the suppression amount of the ignition derived from the spike signal stored in the delay FIFO queue. γ represents a neural eligibility trace. K and L represent the number of traces between a pair of neurons respectively.

Suppose there is a synaptic eligibility trace $\alpha_{i,j}$ in the spike signal propagation between a presynaptic neuron i and a postsynaptic neuron j. In this case $\alpha_{i,j}$ records the sum of the influence quantities received from the spike signal arriving after neuron j has been delayed by d_{ij} from the firing of neuron i. Neural eligibility trace Γ records the sum of the influence amount of presynaptic neuron i due to the firing of postsynaptic neuron j. Eligibility traces α and γ are obtained from equations 8 and 9.

$$\alpha_{i,j,k}^{t-1} = \sum_{s=-\infty}^{t-d_{i,j}} alpha_k(s)x_i[s] \tag{8}$$

$$\gamma_{j,l}^{t-1} = \sum_{s=-\infty}^{t-1} alpha_l(s)x_j[s] \tag{9}$$

The $alpha$ function is an influence amount which the spike signal affects to the neuron, which is obtained from equation 8. The time τ_s before the $alpha$ function reaches the peak value, 1, differs according to k and l in equations 8 and 9. β is a value that suppresses firing derived from the spike signal in the delay FIFO queue and is obtained by the following equation 10.

$$\beta_{i,j,l}^{t-1} = \sum_{s=t-d_{i,j}+1}^{t-1} alpha_l(s-t)x_j[s] \tag{10}$$

The STDP learning with the LTP and LTD weights can be realized due to these eligibility traces. Because of this, the learning mechanism in the SSA-BM model thus takes into account of spike timing of each neuron and spike timing derived from the input sequence.

3.3 Information Coding Based on Spikes

Externally applied input data is propagated to the axon to each LIF neuron after converted to a single spike signal at the encoder. Each neuron propagates the spike signal to the SSA after firing the spike signal. The decoder generates the output to the outside based on a set of synchronized spikes within the time window.

At the learning phase, a sequence of input binary patterns from the external environment is encoded to a sequence of spike signal sets in period ΔT. When the input data is 1, it is assumed that the neuron fires at the first simulation timing in the time window ΔT and a spike signal is generated, and when the input data is 0, there is no spike signal in the time window at all. Similarly, only the fastest output spike of the neuron is propagated to its post-synaptic neurons on the basis of ΔT period.

On the other hand, at the operating/ testing phase, a leading sequence of binary patterns are fed from the external world. In this former phase, the sequence might involve some timing jitters or signal losses. After the leading phase, the SSA-BM begins to infer or predict the remaining sequence of patterns as its output sequence of binary data. In this latter phase, the encoder has to accept the feedback sequence of output binary data at the time (t + ΔT) of the SSA-BM. Therefore, in both phases, the decoder deals with them in the shorter period Δt. Similarly, only the fastest output spike of the neuron is propagated to its post-synaptic neurons on the basis of Δt period.

At the decoder, the results of the neurons that fired earliest in the time window are set to 1, while the results of neurons that fired later than the former neurons or did not fire are set to 0. During the learning phase, input data from the outside is always given at the earliest simulation timing in the time window ΔT. At the testing or operating phase, the input data might arrive at an arbitrary timing within the time window. Because of that, the decoder decides a set of earliest spikes.

3.4 Axon Model

The SSA-BM has an axon composed of a delay queue of the FIFO structure. After a pre-synaptic neuron fires, the spike signal at the post-synaptic neuron is delayed for the time associated with the length of its delay queue. The length of the delay queue is set at random. Even if the pre-synaptic neuron's spike signal arrives at several post-synaptic neurons, the arrival timing is different. Furthermore, the axons of a pair of pre-synaptic neurons and post-synaptic neurons are not one but are set to multiple, and take the structure of Spike Prop [5]. In this case, the lengths of the delay queue of each axon are equal, but the weights and τ_s of $alpha$ function are different. In addition, the alpha function to convert the spike signal to the influence amount on the membrane potential of the neuron is also different. The post-synaptic neuron receives the spike signal and the result converted to the influence quantity by the alpha function is stored as the eligibility trace. Each neuron stores neural eligibility traces and synaptic eligibility traces, each of which stores the amount of influence from firing and the arrival of spike signals.

The SSA-BM has an axon that consists of a delay queue of the FIFO structure as shown in Fig. 1. When the spike signal propagates from the pre-synaptic neuron I to the post-synaptic neuron j and a constant conduction delay is d_{ij}, the length of the delay FIFO queue is $d_{ij} - 1$. Moreover, when the neural network is initialized, $d_{ij} - 1$ is randomly set the length at zero or more. The SSA-BM can express the difference of the distance between neurons, hence the delay FIFO queue. Also, we can use information on when the post-synaptic neuron is propagated the spike signal for calculation and learning of the neuron's membrane potential V_m. Consequently, more precise behavior of the neural network is reproduced by the SSA-BM since we can adjust diverse axon connections to the change of spike timing.

4. Experimental Evaluation

To evaluate the potentiality of the proposed SSA-BM, we used some temporal sequence of simple 2D patterns. This is supposed to infer the trajectory of an object included in a moving picture. In the experimental evaluation, the size of an image frame is set to 4 * 4 pixels and the size of an object included in each image frame is set to 2 * 2 pixels. Thus, the number of neurons in the SSA-BM is 16. The parameters of each LIF neuron are -70 mV for resting membrane potential, -55 mV for firing threshold, and 250 pF for cell membrane electrical capacity. These neurons are simulated at the time resolution, Δt =1 $msec.$

The time window ΔT was set to 10 $msec.$, which is equal to the sampling period of input image frames. Thus, the SSA-BM will learn an output image frame as a succeeded image frame inferred from the present input image at every 10 $msec.$ At the testing or operating phase, when a part of input frame sequence is fed to the SSA-BM, it will infer the succeeded or remaining image sequence. In the evaluation, we will discuss the basic prediction capability and its robustness of the proposed SSA-BM.

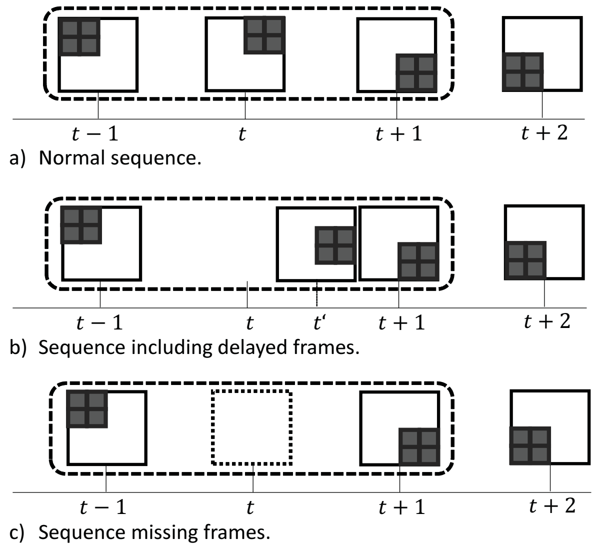

Fig. 2: Learning and testing sequences of 2D-patterns.

4.1 Prediction Capability

In order to evaluate the basic prediction capability of the proposed SSA-BM, we used an image frame sequences shown in Fig. 2(a). At the leaning phase, a whole sequence is fed to the SSA-BM as the input stimulus, while at the testing phase, a former half of the sequence are fed to the SSA-BM and the latter half of the sequence are inferred by the SSA-BM.

In this evaluation, the number of training periods necessary for learning a sequence is decided when the SSA-BM certainly reproduces the half of the correct output sequence. The number of training periods are altered along with the characteristics of the sequence of 2D-pattern. The results are shown in Tables 1, 2, and 3. In those tables, the alphabetical symbol ($\in \{a, c, e, g\}$) in the column of pattern represents one of 2D objects shown in Fig. 3. The training periods corresponding to each training sequence is an average and a deviation of five trials in case of 4, 8 and 16 frames included in a training sequence respectively.

These results indicate that the proposed SSA-BM has the prediction capability of temporal sequence of 2D objects which can be used for object tracking applications or object trajectory prediction embedded in the IoT devices.

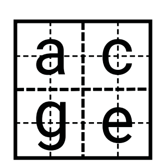

Fig. 3: Symbol corresponding to each 2D-object pattern.

Table 1: Training periods (4 frames).

	Traning sequence	Training periods (ave.± s.d.)
(A)	aceg	2 ± 0
(B)	aeag	3 ± 0
(C)	aage	4.8 ± 1.5
(D)	aege	3.4 ± 0.8
(E)	aaee	24 ± 0
(F)	aeea	20 ± 0.6
(G)	aeee	6 ± 3.7
(H)	eeae	13 ± 0

Table 2: Training periods (8 frames).

	Traning sequence	Training periods (ave.± s.d.)
(A)	aecg aegc	30 ± 0
(B)	aecg agec	50 ± 0
(C)	aecg eagc	30 ± 0
(D)	aecg ecga	70 ± 0
(E)	aecg cega	120 ± 0
(F)	aecg cgae	20 ± 0
(G)	aecg gace	14 ± 4.9
(H)	aecg geac	180 ± 0

Table 3: Training periods (16 frames).

	Traning sequence	Training periods (ave.± s.d.)
(A)	aceg agec aceg agec	40 ± 0
(B)	aceg geae aece aege	1524 ± 47.6
(C)	aecg aecg aaeg eacg	280 ± 0
(D)	aecg aecg aaeg geeg	670 ± 0
(E)	aecg aecg acgc eeec	2112 ± 7.5
(F)	aecg aecg agag acee	714 ± 4.9
(G)	aecg aecg accg aegc	730 ± 0
(H)	aecg aecg aaag caeg	610 ± 0

4.2 Rubustness

We also evaluated the robustness of the SSA-BM by using irregular testing sequences. The one involves one or more delayed frames due to some timing jitter as shown in Fig. 2(b). This delay was set to 5 msec. in this experiment and all testing sequences are composed of 16 frames. The other one misses one or more frames in the training sequence as shown in Fig. 2(c). We measured the number of delayed or missing frames within the former half of sequence is allowed to reproduce the latter half of correct sequence. The results are shown in the table 4. From this result, when mixing frames with temporal errors as exceptional frames, it was possible to obtain the correct output even if exceptional frames were mixed up to 5 frames out of the first 8 frames in case of the sequence (A). Likewise, when mixing missing frames, it was found that exceptional frames can be mixed in up to 4 frames out of the first 8 frames. However, in case of the sequence (B), the SSA-BM did not produce the correct sequence. Since this might be caused due to some convergence failures, we need further optimization of the learning parameters of the SSA-BM.

Table 4: Robustness of SSA-BM.

		allowable frames	
	Traning sequence	missing	delayed
(A)	aceg agec aceg agec	5	0
(B)	aceg geae aece aege	4	0

5. Conclusions

In this study, we proposed a Boltzmann machine with single spike axon (SSA-BM). The proposed model is a recursive spike neural network that incorporates a biological neuron model enable the STDP learning and a Boltzmann probability distribution. The axons that allow only a single spike and the fastest response type output decoder were added to the dynamic Boltzmann machine. Through the preliminary evaluation, the results indicated that the proposed model has a basic prediction capability of 2D-pattern sequence and some robustness against temporal jitter or frame losses of the input stimulus.

Since the evaluation described in the paper is not enough to measure the prediction capability and robustness of the proposed SSA-BM, the further evaluation must be conducted to show its definite inference capability, e.g., the upper boundary of the learnable size of image frame included in the sequence, the number of image frames in the sequence, the number of 2D objects. Furthermore, as for the incomplete sequence including temporal jitter or missing frames, practical robustness for the IoT devices should be evaluated. Finally, the computational complexity and power consumption needed for the SSA-BM on the tiny IoT devices should be measured under the actual environment.

Acknowledgement

Although it is impossible to give credit individually to all those who organized and supported our project, the authors would like to express their sincere appreciation to all the colleagues in the project.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional neural networks,” Proceedings of the 25th International Conference on Neural Information Processing Systems, Vol. 1, pp. 1097–1105, Dec 2012.

[2] H. Lee, R. Grosse, R. Roger, and A. Y. Ng, “Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations,” Proceedings of the 26th Annual International Conference on Machine Learning, pp. 609–616, Jun 2009.

[3] G. E. Hinton, S. Osindero, and Y. W. Teh, “A Fast Learning Algorithm for Deep Belief Nets,” Neural Compution, Vol. 18, pp. 1527–1554, Jul 2006.

[4] G. E. Hinton and R. R. Salakhutdinov, “Reducing the dimensionality of data with neural networks,” Science, Vol. 313, No. 5786, pp. 504–507, Jul 2006.

[5] T. Osogami and M. Otsuka, “Seven neurons memorizing sequences of alphabetical images via spike-timing dependent plasticity,” Scienti c Reports, Vol. 5, No. 14149, 2015.

[6] F. Ponulak and A. Kasinski, “Introduction to spiking neural networks : Information processing, learning and applications,” Acta Neurobiologiae Experimentalis, Vol. 71, pp. 409–433, 2011.

[7] Q. Yu, H. Tang, K. C. Tan, and H. Li, "Rapid feedforward computation by temporal encoding and learning with spiking neurons," IEEE Transactions on Neural Networks and Learning Systems, Vol. 24, No. 10, pp. 1539–1552, Oct 2013.

[8] G. Q. Bi and M. M. Poo, "Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type," Journal of Neuroscience, Vol. 18, pp. 10464–10472, 1998.

[9] H. Ackley, G. E. Hinton, and J. Sejnowski, "A learning algorithm for boltzmann machines," Cognitive Science, Vol. 9, pp. 147–169, 1985.