An Evolving Gene Regulatory Network based Spiking Neural Network for Online Human Behavior Recognition

Yan Meng, Yaochu Jin, Jun Yin, and Matthew Conforth

Abstract— Automatically understanding human behaviors online in a video stream under various scenes is a challenging task. The major difficulty of this task lies in how to combine the spatial and temporal features of video sequences together to extract behavior patterns. To tackle this problem, we propose an evolving gene regulatory network (GRN) based BCM (E-GRN-BCM) model, which is a spiking neural network combined with biological cellular phenomenon. Basically, the weight, weight plasticity, and meta-plasticity of the spiking neural network will be regulated by the GRN, and the GRN will also be influenced by the activity of the neurons it resides in, in a closed loop. Furthermore, an efficient evolutionary algorithm, the covariance matrix adaptation evolution strategy (CMA-ES), is used to justify the GRN parameters, which can significantly reduce the computational cost compared to traditional genetic algorithms. Extensive experimental results have demonstrated the efficiency and robustness of the proposed E-GRN-BCM model for online human behavior recognition under various scenes.

Index Terms—Spiking Neural Network, BCM model, Synaptic plasticity and meta-plasticity, Gene regulatory network, covariance matrix adaptation evolution strategy, online human behavior recognition.

I. INTRODUCTION

Automatically recognizing human behaviors online in a video stream under various complex scenes is a challenging task for intelligent robots and video surveillance applications due to its large amount of video data to be analyzed and its real-time requirements. Traditionally, most activity recognition work has focused on representing and learning the sequential and temporal characteristics in activity sequences. This has led to the widespread use of dynamic models such as the hidden Markov model (HMM) [1][2][3]. While the HMM is a simple and efficient model for learning sequential data, its performance tends to degrade when the range of activities becomes more complex, or the activities exhibit long-term temporal dependency that is difficult to deal with under the Markov assumption.

On the other side, neural network has played a key role for intelligent and cognitive systems. The artificial neuron is a node which computes a weighted sum of its inputs and passes the result through a transfer function. This simple model has proved very useful in a variety of connectionist computing and nonlinear modeling tasks. However, we question whether removing the details of biological neurons from artificial neuron has also removed some important neuron behaviors or capabilities.

The scientific understanding of biological neurons and neural networks has advanced considerably in the time since artificial neural networks were originally devised. The Bienenstock, Cooper, and Munro (BCM) model [5] is one of the most famous and well-supported modern models of biological neurons [6][7][8]. The BCM model is, more precisely, a model of the plasticity and meta-plasticity of synapses in biological neural networks. Detailed simulations of BCM neurons can provide biologically sound predictions.

Edelman et al. [9] used a single layer network and a learning rule based on BCM to extract a visual code in an unsupervised manner. The role of calcium concentration in the spike time dependent plasticity (STDP) of BCM model neurons was investigated by Kurashige and Sakai [10]. Shouval et al. [11] demonstrated evidence for the calcium control hypothesis by comparing biologically motivated models of synaptic plasticity. A reinforcement learning STDP rule developed by Baras and Meir was shown to exhibit statistically similar behavior to the BCM model [12].

While neuroscience has made important progress in modeling the brain, neuro-inspired engineering models are still primarily using simple weighted sum and transform models for artificial neural networks. Detailed biologically accurate neural models are too computationally expensive to replace the simple models for most engineering applications. We intend to develop a compromise model that retains more powerful learning and remembering capability, while sacrificing biological accuracy for better computational efficiency.

In this paper, we aim to adapt the BCM model into a computer engineering tool. Since the behavior of a BCM neuron is governed by the local neural activation history, we need another learning mechanism by which to optimize a BCM neural network to a particular task, especially for those tasks that need to extract the temporal features from the input data, such as behavior pattern recognition.

To this end, we turn our attention to gene regulatory networks. During the biological morphogenesis, genes in each cell are expressed, resulting in various cellular functions. The expression of the genes is regulated by their own protein

Yan Meng, Jun Yin, and Matthew Conforth are with Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ 07670, USA (Phone: 201-2165496, e-mail: yan.meng@stevens.edu, matthew.conforth@stevens.edu and jyin@stevens.edu).

Yaochu Jin is with Honda Research Institute Europe, Carl-Legien-Str. 30, 63073, Offenbach, Germany(e-mail: yaochu.jin@honda-ri.de).
products as well as proteins produced by other genes in the same cell or neighboring cells through intracellular and intercellular diffusion, forming a gene regulatory network that can be described by a set of coupled ordinary differential equations. A GRN models the collection of DNA segments in a cell and their indirect interactions with each other via their RNA and protein expression products. GRNs also include some other chemical concentrations in and around the cell which are deemed relevant in their particular context. GRNs play a central role in understanding natural evolution and organism development [13]. Therefore, it is natural for us to use GRN to optimize the BCM model for specific tasks.

Therefore, in this paper, we propose an evolving gene regulatory network based BCM (E-GRN-BCM) model, which is a spiking neural network combined with the biological cellular phenomena, and then apply this model to online human behavior recognition.

The basic idea of this E-GRN-BCM model is as follows. The BCM based spiking neural network is a graph with weighted, directed edges replacing synapses (since the model need not exist physically in 3D space, dendrites and axons are not necessary). The weight, weight plasticity, and meta-plasticity of the spiking neural network will all be regulated by the GRN, and the GRN will be influenced by the activity of the neurons it resides in, in a closed loop. Furthermore, an efficient evolutionary algorithm, i.e., the covariance matrix adaption evolution strategy (CMA-ES) [19][20], is used to evolve the GRN parameters to optimize the BCM neural network for a specific task, such as online human behavior recognition in a video stream. The neurons will have a spiking rather than continuous output behavior, and each neuron will maintain a recent history for the inclusion of heavily abstracted effects such as spike timing dependent plasticity, ion concentration, membrane potential, etc.

The rest of the paper is organized as follows. Section II describes some backgrounds such as the BCM model, the GRN model, and CMA evolution strategy. The E-GRN-BCM model is presented in Section III. Section IV discusses how to apply the E-GRN-BCM model into human behavior recognition. The experimental results on human behavior recognition are shown in Section V. The paper is concluded by Section VI.

II. BACKGROUNDS

A. The BCM Spiking Neuron Model

For computational efficiency we use a simplified, discrete-time version of the BCM model [5]. The equations governing the behavior of synaptic weights are:

\[
w_i(t+1) = \eta x_i(t) \phi(y(t), \theta(t)) + (1-\epsilon)w_i(t)
\]

\[
\phi(y(t), \theta(t)) = y(t)(y(t) - \theta(t))
\]

\[
\theta(t) = \frac{\sum_{t'=\tau-h}^{\tau} y^2(t') \lambda e^{-\lambda(t'-\tau)}}{2 \sum_{t'=\tau-h}^{\tau} e^{-\lambda(t'-\tau)}}
\]

where \(w_i\) denotes the synaptic weight of the \(i\)-th synapse. \(\eta\) is the constant learning rate. \(x_i\) is the pre-synaptic input of the \(i\)-th synapse. \(y\) is the post-synaptic output activation level. \(\theta\) is the sliding modification threshold. \(\phi(\cdot)\) is the non-linear activation that swings with the sliding threshold \(\phi\), and it can be defined by Eqn. (2). \(\epsilon\) is the time decay constant that is uniform for all synapses. The interpretation of \(\theta\) in Eqn. (3) clearly shows the sliding threshold to be a time-weighted average of squared post-synaptic signals \(y\) within the time interval of \(h\), where \(\lambda\) is the forgetting factor.

The neuron spiking behavior is conceptually relatively simple. If the weighted sum of the pre-synaptic activation levels is less than \(\theta\), or if the neuron fired in the previous time step, then the neuron activation level will decrease. Otherwise, the neuron will fire.

B. The Gene Regulatory Network

Multi-cellular morphogenesis is under the control of gene regulatory networks. When a gene is expressed, information stored in the genome is transcribed into mRNA and then translated into proteins. Some of these proteins are transcription factors that can regulate the expression of their own or other genes, thus resulting in a complex network of interacting genes termed as a gene regulatory network (GRN). To understand the emergent morphology resulting from the interactions of genes in a regulatory network, reconstruction of gene regulatory pathways using a computational model has become popular in systems biology. A large number of computational models for GRNs have been suggested [14-18]. Among others, ordinary or partial differential equations have widely been used to model regulatory networks.

Mjolsness et al. (1991) proposed a GRN model for describing the gene expression data of developmental processes, which can be considered as a generalized reaction-diffusion model in a continuous form with a sigmoid function:

\[
\frac{dg_{ij}}{dt} = -\gamma_j g_{ij} + \phi \left[ \sum_{l=1}^{n_p} W_{ij}^l g_{ij} + \theta_j \right] + D_j \nabla^2 g_{ij}, \quad (4)
\]

where \(g_{ij}\) denotes the concentration of \(j\)-th gene product (protein) in the \(i\)-th cell. The first term on the right-hand side of Eqn. (4) represents the degradation of the protein at a rate of \(\gamma_j\), the second term specifies the production of protein \(g_{ij}\), and the last term describes protein diffusion at a rate of \(D_j\).

\(\phi\) is an activation function for the protein production, which is usually defined as a sigmoid function \(\phi(z) = 1/(1 + \exp(-\mu z))\). The interaction between the genes is described with an interaction matrix \(W^i\), the element of which can be either active (a positive value) or repressive (a negative value). \(\theta_j\) is a threshold for activation of gene expression. \(n_p\) is the number of proteins.

C. The CMA Evolution Strategy

The covariance matrix adaption evolution strategy (CMA-ES) is a stochastic, iterative optimization method belonging to the class of evolutionary algorithm proposed by Hansen and Ostermeier [19][20]. The covariance matrix adaptation (CMA) is a method to update the covariance matrix of the multivariate
normal mutation distribution in the evolution strategy. New candidate solutions are sampled according to the mutation distribution. In contrast to classical methods, only the ranking between candidate solutions is exploited during learning. Neither derivatives nor even the function values itself are required by the method.

Compared to the traditional evolution algorithms, two important principles of the CMA-ES are: (1) the covariance matrix of the distribution is updated such that the likelihood of previously realized successful steps to appear again is increased; (2) two evolution paths are applied to record the two paths of the time evolution of the distribution mean of the strategy, where one path contains significant information of the correlation between consecutive steps and the other path is used to conduct an additional step-size control. The adaptation of covariance matrix can expedite the overall convergence to an optimum meanwhile can effectively reduce the computational cost of the overall system. The block diagram of this E-GRN-BCM model is shown in Fig. 1.

Another obvious advantage of the CMA-ES is that it does not require a tedious parameter tuning for its applications such that its computational cost can be significantly reduced compared to other traditional genetic algorithms, and this is the main reason for us to choose CMA-ES for as the evolutionary algorithm in this paper due to the real-time performance requirement in online human behavior recognition.

III. THE E-GRN-BCM MODEL

A. The Framework of the E-GRN-BCM Model

We use Eqn. (1)-(3) as the BCM model for our spiking neural network. This spiking neural network can be applied to various real-world applications, such as human behavior recognition in a video stream, or intrusion behavior recognition in a cognitive radio network. To facilitate the online learning capability of this BCM-based spiking neural network, we propose to use a gene regulatory network (GRN) to regulate the tunable parameters ($\eta$, $\epsilon$, $\lambda$) of the BCM model in Eqn. (1)-(3), which are used as the output to the GRN model.

To apply the GRN dynamics to regulate the parameters of the BCM model to describe the multi-cellular phenomenon, we will make the following assumptions. First, we assume that each BCM parameter corresponds to the expression level of one gene. Second, we assume that, through some chain of mechanisms, the concentrations of the relevant proteins can be deduced from the sodium ion and/or calcium ion concentrations. Finally, we assume that the sodium ion concentration (represented by $c_{Na^+}$) is proportional to the input current and the calcium ion concentration (represented by $c_{Ca^{2+}}$) is proportional to the activation threshold. Therefore, the inputs of this GRN regulation consist of two concentrations: sodium ion concentration $c_{Na^+}$ and calcium ion concentration $c_{Ca^{2+}}$.

Since we will use differential equations to represent the GRN model, there are several parameters in the differential equations need to be defined. As we know, it is very tedious to optimize these parameters for each specific application if we do it manually. Therefore, an evolutionary algorithm is applied here to automatically optimize these parameters for the GRN model to each specific application tasks at hand. More specifically, the covariance matrix adaptation evolution strategy (CMA-ES) is applied to expedite the convergence and reduce the computational cost of the overall system. The block diagram of this E-GRN-BCM model is shown in Fig. 1.

\[
\eta(t+1) = (1 - \gamma_\eta)\eta(t) + \alpha_\eta f(c_{Na^+}(t), c_{Ca^{2+}}(t)) \quad (5)
\]

\[
\epsilon(t+1) = (1 - \gamma_\epsilon)\epsilon(t) + \alpha_\epsilon g(c_{Na^+}(t), c_{Ca^{2+}}(t)) \quad (6)
\]

\[
\lambda(t+1) = (1 - \gamma_\lambda)\lambda(t) + \alpha_\lambda h(c_{Na^+}(t), c_{Ca^{2+}}(t)) \quad (7)
\]

and the protein expression levels for sodium ion and calcium ion concentration ($c_{Na^+}$, $c_{Ca^{2+}}$) are defined as:

\[
c_{Na^+}(t+1) = (1 - \gamma_{c_{Na^+}})c_{Na^+}(t) + \alpha_{c_{Na^+}} \sum_i x_i(t) w_i(t) \quad (8)
\]

\[
c_{Ca^{2+}}(t+1) = (1 - \gamma_{c_{Ca^{2+}}})c_{Ca^{2+}}(t) + \alpha_{c_{Ca^{2+}}} \theta(t) \quad (9)
\]

where $\gamma_\eta$, $\gamma_\epsilon$, $\gamma_\lambda$, $\gamma_{c_{Na^+}}$, and $\gamma_{c_{Ca^{2+}}}$ are decay factors and $\alpha_\eta$, $\alpha_\epsilon$, $\alpha_\lambda$, $\alpha_{c_{Na^+}}$, and $\alpha_{c_{Ca^{2+}}}$ are coefficients. Functions $f(\cdot)$, $g(\cdot)$, and $h(\cdot)$ are defined as the following sigmoid functions:

\[
f(c_{Na^+}, c_{Ca^{2+}}) = \left(1 + e^{-(k_1c_{Na^+} + k_2c_{Ca^{2+}})}\right)^{-1} \quad (10)
\]

\[
g(c_{Na^+}, c_{Ca^{2+}}) = \left(1 + e^{-(k_1c_{Na^+} + k_2c_{Ca^{2+}})}\right)^{-1} \quad (11)
\]

\[
h(c_{Na^+}, c_{Ca^{2+}}) = \left(1 + e^{-(k_3c_{Na^+} + k_4c_{Ca^{2+}})}\right)^{-1} \quad (12)
\]

where $k_1$ and $k_2$ are coefficients.

In this basic GRN-SNN model, there are several parameters $\gamma_\eta$, $\gamma_\epsilon$, $\gamma_\lambda$, $\gamma_{c_{Na^+}}$, $\gamma_{c_{Ca^{2+}}}$, $\alpha_\eta$, $\alpha_\epsilon$, $\alpha_\lambda$, $\alpha_{c_{Na^+}}$, $\alpha_{c_{Ca^{2+}}}$, $k_1$, and $k_2$ needed to be defined. As a general model, this basic GRN-SNN model can be applied to various applications, such as behavior pattern recognition in video streams or cognitive radio networks. However, for each specific task, these parameters need to be carefully tuned so that an optimal performance of the corresponding GRN-BCM model can be achieved. However, fine tuning these parameters for optimal
performance is very tedious and time consuming if we do it manually and empirically. Therefore, we plan to use an evolutionary algorithm to automatically generate these parameters for this GRN-BCM model.

C. Evolving the GRN-BCM Model using the CMA-ES

Since there are 12 parameters need to be adjusted for a single neuron in a spiking neural network, for a simple hierarchical spiking neural network, for example, 3 layers with 24x32 neurons in the input layer, it will be extremely computational expensive using standard evolutionary algorithms for this GRN-BCM model even if we use very small populations, such as 40. Therefore, we employ a relative efficient evolutionary algorithm, namely, covariance matrix adaptation evolution strategy (CMA-ES), to configure the parameters of the GRN model. The main advantages of using the CMA-ES method are the invariance property in adapting an arbitrarily oriented scaling of the search space and its computational efficiency.

The five gamma (γ₀, γₑ, γₐ, γₑ₊, and γₑ₋₊) and five alpha (α₀, αₑ, αₐ, αₑ₊, and αₑ₋₊) in Eqs. (5)-(9), and two k (k₁ and k₂) terms in Eqs. (10)-(12) are constant values to be optimized. Based on the CMA-ES [19], k offspring are generated from μ parents, where k denotes the population size (sample size) and μ denotes the parent population size, i.e., the number of selected points (μ ≤ k). According to the GRN-BCM model, candidate solutions are 12-dimensional vectors of real numbers. The quality of candidate solutions is determined by the objective function. Covariance matrix records the important information between consecutive steps. To increase the probability of repeating the previously successful steps, the covariance matrix of the mutation distribution is to be updated. In other words, this procedure defines the direction of search space and results in a fast convergence.

The new values of designated parameters are sampled in a normally distributed way, and the basic equation for each solution at different generations g = 0, 1, 2, ..., is defined as follows:

$$x_i^{(g+1)} \sim \mathcal{N}(0, \sigma_i^{(g)}) \quad \text{for} \quad i = 1, \ldots, k$$

where “∼” denotes the same distribution on the left and right side. $x_i^{(g+1)}$ is the $i$-th solution from generation $g + 1$, and $x$ is a vector $x \in \mathbb{R}^n$, which consists of five gamma (γ₀, γₑ, γₐ, γₑ₊, and γₑ₋₊), five alpha (α₀, αₑ, αₐ, αₑ₊, and αₑ₋₊), and two k (k₁ and k₂). The mean vector $m(g) \in \mathbb{R}^n$ represents the favorite solution at generation $g$. $\sigma(g)$ is the step size which controls the step length at generation $g$. $C(g)$ denotes the covariance matrix at generation $g$ and $\mathcal{N}(0, C(g))$ is the i-th multivariate normal distribution with zero mean and covariance matrix $C(g)$.

To determine the complete iteration step, $m(g+1)$, $\sigma(g+1)$, and $C(g+1)$ are defined as:

$$m(g+1) = m(g) + \sigma(g) y_w(g) \quad \text{(13)}$$

$$\sigma(g+1) = \sigma(g) \times \left( \frac{e}{d_{\sigma}} \left( \frac{\|p_{\sigma}^{(g+1)}\|_2}{e^{|K_N(0,\|0\|)}} - 1 \right) \right) \quad \text{(14)}$$

$$C(g+1) = \left( 1 - c_1 - c_\mu \right) C(g) + c_1 P_{\sigma}^{(g+1)} P_{\sigma}^{(g+1)T} + c_\mu \sum_{i=1}^{\mu} W_i y_i^{(g+1)}(y_i^{(g+1)})^T \quad \text{(15)}$$

where $c_\sigma < 1$ is the learning rate for the step-size control. $d_{\sigma} \approx 1$ is the damping parameter for step-size update. $c_1$ and $c_\mu$ are the learning rates for the rank-one update and the rank-$\mu$ update of the covariance matrix, respectively. $W_i$ is the $i$-th positive weight coefficient for recombination. In addition, the functions $y_w^{(g+1)}$, $P_{\sigma}^{(g+1)}$, and $P_{\sigma}^{(g+1)}$ can be obtained from generation $(g + 1)$ as follows,

$$y_w^{(g+1)} = \prod_{i=1}^{\mu} W_i y_i^{(g+1)}$$

$$P_{\sigma}^{(g+1)} = (1 - c_1) P_{\sigma}^{(g)} + \sqrt{c_\sigma (2 - c_\mu) \mu_{eff}} C(g)^{-1} m(g+1) - m(g) \sigma(g) \quad \text{(16)}$$

$$P_{\sigma}^{(g+1)} = (1 - c_1) P_{\sigma}^{(g)} + \sqrt{c_\sigma (2 - c_\mu) \mu_{eff}} m(g+1) - m(g) \sigma(g) \quad \text{(17)}$$

where $\mu_{eff} = \left( \sum_{i=1}^{\mu} W_i^2 \right)^{-1}$, is the variance effective selection mass and $y_i^{(g+1)} = (x_i^{(g+1)} - m(g)) / \sigma(g)$.

IV. Applying the E-GRN-BCM Model to Human Behavior Recognition

A. The System Framework

Now we need to apply the proposed E-GRN-BCM model to online human behavior recognition from video sequences. The system framework of the visual human behavior pattern recognition using the E-GRN-BCM model is shown in Fig. 1. The objective of this system is to online recognize different human behaviors, such as working, running, hand waving, etc., from video sequences automatically.

The E-GRN-BCM model takes the full advantage of the spiking neural network (SNN) to implement the human behavior recognition. According to the structure of SNNs, each neuron may have lots of connections with other neurons. For computational efficiency, however, we assume each neuron is only connected to its neighbors and the neural network has three layers, i.e., two middle layers and one top layer. As shown in Fig. 2, the neuron of (1, 1) in the lower-middle layer is connected to the local neighbor neurons of (1, 2), (2, 1) and (2, 2) within the lower-middle layer. It is noted that only the neurons in the two middle layers can be connected to neighbor neurons of the same layer. In addition, the neuron of (1, 1) has the connections with the neurons of (1, 1), (1, 2), (2, 1) and (2, 2) in the upper-middle layer.

When the video sequence comes, it will be preprocessed to extract corresponding spatial features frame by frame. Given the features of $m \times n$, each feature is fed into the lower-
middle layer as an input. Due to the assumption that connections only exist within local neighbors, the features at the border have less connections with the neurons in the lower-middle layer than those at the center. To maximally deliver the input information into the network, the lower-middle layer is designed with the size of \((m+2) \times (n+2)\). The upper-middle layer is set the same size with the input layer, \(m \times n\). The top layer is the label layer and is fully connected to the upper-middle layer, which distinguishes the human behavior patterns.

![Diagram of the E-GRN-BCM model](image)

Fig. 2. The framework of the E-GRN-BCM model for visual human behavior recognition.

### B. Spatial Feature Extraction

Generally, each video sequence consists of a collection of various frames, which is the bottom layer in this framework. We adopt 2D interest point detectors for the detection of spatial behavior features, which model the geometric relationships of the human body. These features of the co-occurrence of moving pixels represent the low-level visual features since they are more reliable in a video sequence.

One of the most popular approaches to interest point detection in the spatial domain is based on the detection of corners. Corners are defined as regions where the local gradient vectors point in orthogonal directions. The gradient vectors are obtained by taking the first order derivatives of a smoothed image \(I(x,y,\sigma) = I(x,y) * g(x,y,\sigma)\), where \(g\) is the Gaussian smoothing kernel, and \(\sigma\) controls the spatial scale at which corners are detected. The response strength at each point is then based on the rank of the covariance matrix of the gradient calculated in a local window.

Following the observation that human behaviors are the results of a sequence of poses, a single pattern can be represented according to the semantic relationships among these poses. Fig. 3 shows one example of extracting “walking” behavior features from a video sequence.

![An example of extracting the “walking” behavior](image)

Fig. 3. An example of extracting the “walking” behavior

### C. Supervised Learning using the E-GRN-BCM Model

The extracted spatial behavior features are then fed into a hierarchical BCM spiking neural network. Since it is a very challenging task to extract temporal features from the image frames directly, the major advantage of this hierarchical BCM model is that the temporal information has been naturally embedded into the BCM model so that the temporal features extraction can be skipped in the preprocessing, which can dramatically improve the overall system performance and reduce the system errors.

Since this is a supervised learning, during the training process, the GRN model is used to regulate three tunable parameters, i.e., \((\eta, \epsilon, \lambda)\) of the BCM spiking neural network to learn specific behavior patterns using example pairs of input video frames and output labels. The CMA-ES method is then applied to automatically generate parameters for the GRN model. The trained BCM spiking neural network should be able to produce an output that represents the behavior pattern category that current input video belongs to. So the architecture of the neural network must be constructed in a layered feed-forward way.

After using this evolving GRN-BCM model for training, the trained spiking neural network should not only be able to reproduce the correct output to a given training video sequence, but also be able to recognize the behavior patterns from those unseen input data if there is a similar pattern has been trained before.

### V. Experimental Results

#### A. Configuration of Spiking Neural Network

To evaluate the performance of the proposed E-GRN-SNN model, several experiments in different scenarios have been conducted. Basically, the effectiveness of the framework is measured by two criteria: how accurately and robustly the different classes of behavior patterns can be recognized. We test the proposed model using the KTH datasets (http://www.nada.kth.se/cvap/actions/) and Weizmann human motion datasets [21].

For these two dataset, the video is usually captured in the format of 320 x 240 pixels per frame. If we use the pixel-level input to the spiking neural network, the overall computational cost would be very expensive due the large neuron numbers.
and the employment of the evolutionary algorithms. Therefore, as shown in Fig. 1, in the features layer, the inputs are reduced to 32x24 in size in proportion to the original pixel-level inputs. That is, the size of the input layer in the Fig. 2 is 320 x 240, and the size of the features layer is 32 x 24. Then the lower-middle layer is composed of 34 x 26 neurons, and the upper-middle layer is composed of 32 x 24 neurons. The label layer is fully connected to the upper-middle layer. The CMA-ES doesn’t require a large population for its application, so here we set k = 20 and μ = 10.

B. Behavior Recognition on KTH Dataset

The KTH dataset contains several action classes performed by many subjects. Each video sequence has only one action. Fig.4 shows the example of walking, running, hand-waving and boxing behaviors. We test 400 video sequences including these four behaviors of different people. Fig. 5 demonstrates the extracted spatial features from the original frames.

![Example images from video sequences in KTH dataset](image)

Fig. 4. Example images from video sequences in KTH dataset: (a) walking behavior, (b) running behavior, (c) hand-waving behavior, and (d) boxing behavior.

The recognition rate is defined as the percentage of correctly recognized behaviors from the number of all samples, which is

\[ R = \frac{N_r}{N} \times 100\% \]  (18)

where \( R \) is the recognition rate, \( N \) is the number of input behaviors, and \( N_r \) is the number of correctly recognized behaviors. Table I shows the recognition results as follows,

<table>
<thead>
<tr>
<th>Behaviors</th>
<th>N</th>
<th>( N_r )</th>
<th>R(%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walking</td>
<td>100</td>
<td>86</td>
<td>86</td>
</tr>
<tr>
<td>Running</td>
<td>100</td>
<td>80</td>
<td>80</td>
</tr>
<tr>
<td>Hand-waving</td>
<td>100</td>
<td>87</td>
<td>87</td>
</tr>
<tr>
<td>Boxing</td>
<td>100</td>
<td>83</td>
<td>83</td>
</tr>
<tr>
<td>Overall</td>
<td>400</td>
<td>336</td>
<td>84</td>
</tr>
</tbody>
</table>
Fig. 5. In the KTH dataset, the original video sequences for behavior patterns of “walking”, “running”, “hand-waving” and “boxing” at top rows, and the extracted spatial features (represented by red shaded areas) from these behavior patterns at the bottom rows. (a) “walking” behavior, (b) “running” behavior, (c) “hand-waving” behavior, (d) “boxing” behavior.

From Table I, it can be seen that the behavior recognition rate on the KTH dataset using the proposed E-GRN-BCM model is high with the overall recognition rate of 84%. That shows that our proposed model turns out to be very effective for human behavior recognition.

In a comparison to the previous work of human behavior detection [22][23], our method shows competitive performance as listed in Table II. From Table II, we can see that our proposed E-GRN-BCM model has the highest average recognition rate of 84% compared to other alternative methods. In other words, the proposed E-GRN-BCM model is efficient in supervised learning for online human behavior recognition from video sequences.

### TABLE II

**COMPARISON OF DIFFERENT METHODS**

<table>
<thead>
<tr>
<th>Methods</th>
<th>Recognition Rate</th>
<th>Learning</th>
</tr>
</thead>
<tbody>
<tr>
<td>Our method</td>
<td>84%</td>
<td>Labeled</td>
</tr>
<tr>
<td>Dollar et al. [23]</td>
<td>81.17%</td>
<td>Labeled</td>
</tr>
<tr>
<td>Schuldt et al. [22]</td>
<td>71.72%</td>
<td>Labeled</td>
</tr>
</tbody>
</table>

C. Behavior Recognition on Weizmann Dataset

Then we evaluate the performance of the E-GRN-BCM model another commonly-used dataset, Weizmann human action dataset [21]. This is a database with 90 low-resolution video sequences showing nine different people, each performing 10 natural actions such as “running,” “walking,” “skipping,” “jumping-jack” (or shortly “jack”), “jump-forward-on-two-legs” (or “jumping”), “jump-in-place-on-two-legs” (or “pjumping”), “galloping-sideways” (or “siding”), “waving-two-hands” (or “waving2”), “waving-one-hand” (or “waving1”), or “bending”. Fig. 6 shows the examples of these behaviors from this database. There are ten different behaviors in total.

![Fig. 6. The example images from different behavior video sequences in Weizmann dataset.](image)

From Table III, it can be seen that the recognition results are worse than the previous experiments on KTH dataset. The recognition rates drop a little bit, i.e. from 84% to 73.33%.

### TABLE III

**BEHAVIOR RECOGNITION RESULTS**

<table>
<thead>
<tr>
<th>Behaviors</th>
<th>N</th>
<th>N_R</th>
<th>R (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Walking</td>
<td>9</td>
<td>7</td>
<td>77.78</td>
</tr>
<tr>
<td>Running</td>
<td>9</td>
<td>6</td>
<td>66.67</td>
</tr>
<tr>
<td>Hand-waving</td>
<td>9</td>
<td>7</td>
<td>77.78</td>
</tr>
<tr>
<td>Pjumping</td>
<td>9</td>
<td>5</td>
<td>55.56</td>
</tr>
<tr>
<td>Siding</td>
<td>9</td>
<td>7</td>
<td>77.78</td>
</tr>
<tr>
<td>Jacking</td>
<td>9</td>
<td>6</td>
<td>66.67</td>
</tr>
<tr>
<td>Skipping</td>
<td>9</td>
<td>6</td>
<td>66.67</td>
</tr>
<tr>
<td>Waving1</td>
<td>9</td>
<td>7</td>
<td>77.78</td>
</tr>
<tr>
<td>Bending</td>
<td>9</td>
<td>8</td>
<td>88.89</td>
</tr>
<tr>
<td>Jumping</td>
<td>9</td>
<td>7</td>
<td>77.78</td>
</tr>
<tr>
<td>Overall</td>
<td>90</td>
<td>66</td>
<td>73.33</td>
</tr>
</tbody>
</table>
The reason is that the Weizmann dataset has less human behavior samples, but has more categories of behavior. In addition, some human actions are very close from the motion features.

To compare the experimental results with other alternative method, we conduct another set of experiments on the Weizmann dataset using the method proposed in [21]. The comparison results are listed in Table IV. From Table IV, we can see that the performance of our method is much better than the method used in [21]. This result further verified the efficiency of the proposed E-GRN-BCM model for online behavior recognition.

VI. CONCLUSION AND FUTURE WORK

This paper proposed a novel spiking neural network based model for online human behavior recognition, where the gene regulatory networks (GRN) dynamics is used to regulate the plasticity and meta-plasticity of the BCM model and an CMA evolution strategy is applied to automatically generate parameters of GRN dynamics to achieve optimal performance of the BCM model for specific tasks. This so-called E-GRN-BCM model takes full advantage of the capabilities of spiking neurons where the processing of temporal data is embedded into the model by their dynamic nature. Furthermore, the evolving GRN-based learning rule is proposed to tackle the intractable problem of tuning the parameters of BCM spiking neural network. Extensive experimental results using two different behavior datasets have demonstrated that the proposed E-GRN-SNN model is very efficient and accurate for online behavior pattern recognition.

Although we have made some progress in developing a novel framework for behavior recognition using spiking neural networks, only two different datasets for simple behaviors under structured environments have been tested in this paper to validate the proposed framework. In the future, we will extend this model into a more robust and efficient method for more complex behavior detection and representation in more complex scenes in unstructured environments.

REFERENCES


