

Research Article

Position Estimation Based on Grid Cells and Self-Growing Self-Organizing Map

Baozhong Li ¹, Yanming Liu,¹ and Hailin Li²

¹*Xidian University, Xi'an, China*

²*College of Information and Navigation, Air Force Engineering University, Xi'an, China*

Correspondence should be addressed to Baozhong Li; bzh-li@163.com

Received 27 October 2018; Accepted 13 January 2019; Published 26 February 2019

Academic Editor: Michele Migliore

Copyright © 2019 Baozhong Li et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

As the basis of animals' natal homing behavior, path integration can continuously provide current position information relative to the initial position. Some neurons in freely moving animals' brains can encode current positions and surrounding environments by special firing patterns. Research studies show that neurons such as grid cells (GCs) in the hippocampus of animals' brains are related to the path integration. They might encode the coordinate of the animal's current position in the same way as the residue number system (RNS) which is based on the Chinese remainder theorem (CRT). Hence, in order to provide vehicles a bionic position estimation method, we propose a model to decode the GCs' encoding information based on the improved traditional self-organizing map (SOM), and this model makes full use of GCs' firing characteristics. The details of the model are discussed in this paper. Besides, the model is realized by computer simulation, and its performance is analyzed under different conditions. Simulation results indicate that the proposed position estimation model is effective and stable.

1. Introduction

Rapid development of unmanned vehicles has raised the research of autonomous navigation in recent years [1]. Getting a robust position estimation is very important for vehicles to achieve autonomous navigation tasks. Over the past 30 years, there has been much effort in solving this problem by building a map of the environment and navigating based on estimation of position in that map, the methodology of which has come to be known as simultaneous localization and mapping (SLAM) [2, 3]. Traditional approaches of SLAM are the probabilistic methods, the typical representative of which is extended Kalman filter (EKF) [4]. The computation cost of EKF is quadratic with respect to the number of the landmarks, which may result in performance degradation in large-scale environment. Many improved methods are provided to solve this problem [5], but they can only alleviate it in some extent, rather than fundamentally eliminate this problem.

Many animals exhibit perfect navigation capability in complex and large-scale environments, even when the

perceived information is not exact. Taking inspiration from animals, researchers begin to propose bioinspired navigation architectures for unmanned vehicles to improve their autonomous navigation capability. Natal homing is a remarkable and common navigation behavior which enables animals such as fish, rats, and pigeons to perform long migrations to return to their natal areas or nests [6, 7]. The foundation of natal homing in animals' brains is the path integration (PI) mechanism. It can continuously accumulate self-motion information and update and provide current position vector relative to the reference position (e.g., the initial starting point) even in the absence of other external perception information such as vision [8], and this position vector provides the key navigation information for natal homing. Traditionally, PI is implemented by mathematical formulas, but the neural mechanism in animals' brains determines that PI must be realized through neural networks [9, 10]. Neural recordings from laboratory and theory research studies indicate that there are a variety of cells associated with navigation, which include place cells (PCs) [11], head direction cells (HDCs) [12, 13], grid cells (GCs)

[14], and border cells (BCs) [15]. Their firing patterns are closely related to the surrounding environments and the animals' positions. Although there is no definite conclusion about how navigation behaviors are achieved in brains, it is considered that GCs are the neural basis of path integration [16].

GCs are a kind of neurons that exist in the entorhinal cortex (EC) of rats' brain. A GC shows multiple spatial firing fields during the movement of the animal, and these firing fields distribute in a regular hexagonal array that uniformly and periodically covers the whole environment visited by the animal [14, 17]. These spatial firing fields are independent of environmental characteristics. Figure 1(a) shows a single GC's firing pattern from laboratory. The abstract geometry of the firing fields is also illustrated in Figure 1(b). There are mainly four parameters to characterize the firing pattern of GCs: grid period, grid orientation, firing field size, and grid phase. As presented in Figure 1(b), the grid period is defined as the median distance between the center firing field and the six surrounding firing fields. The grid orientation is defined as the angle between a fixed reference line going through the center firing field and the closest of the three main diagonals of the hexagon in counterclockwise direction. The firing field size is defined as the radius of the individual firing fields. In contrast to the above three parameters, the grid phase is not an absolute value. It describes the relative displacement between the firing fields of two co-located GCs, i.e., GCs with similar grid period, grid orientation, and firing field size. GCs that are proximate within the medial entorhinal cortex (mEC) are organized into a function module named as a GC module. The firing patterns of the cells belonging to the same GC module share the same grid period and orientation but a fixed grid phase relative to one another. The relative grid phases appear to be conserved across all environments visited by the animal. Grid periods increase discontinuously between function modules along the dorsoventral axis of mEC, with the smallest being around 25 cm and the largest so far recorded exceeding 300 cm [14, 16–19].

Currently, many types of position estimation models based on GCs have been proposed. Samsonovich and McNaughton [20] presented an attractor model that consists of two levels: a P-level, which is a two-dimensional sheet of neurons representing the current position of the animal, and an I-level which controls updating of the position. The position of the neural activity packet on the P-level can be continuously moved according to the input velocity. Conklin and Eliasmith [21] described a novel attractor network model which incorporates representation and updating of position into a single layer of neurons, eliminating the need for a large external control population, and without making use of multiplicative synapses. Samsonovich and McNaughton [20] proposed improved continuous attractor models that are capable of generating accurate regular hexagonal grid responses to the animal's position in 2-D space. Mhatre et al. [22] described a simple and general mathematical property of the trigonometry of spatial navigation and developed a neural model which was named as GRIDSmap. The model can learn to exploit the trigonometric relationship and convert path integration signals into

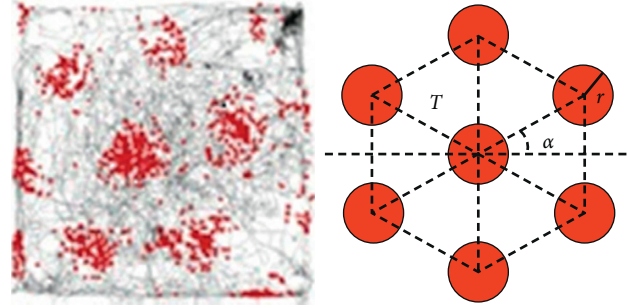


FIGURE 1: Firing pattern of a single grid cell (GC) from laboratory. (a) Firing locations and running trajectory (the gray line is the running trajectory of the rat, and each spike is plotted in red) [18]; (b) The abstract geometry of the grid fields.

hexagonal grid cell patterns of multiple scales, and so on. The above-mentioned models are mainly designed to simulate or reproduce the firing patterns of cells to explore the mechanism of cells' firing from the biological point of view, not for the sake of practical navigation application or improvement of navigation performance. The encoding of spatial environments and positions by navigation-related cells' firing patterns is just the first step in the process of perceived information processing. How to decode the relevant information that can be applied to practical navigation, i.e., how to establish the relationship between the cells' firing patterns and the related navigation parameters is an important subject in the research of bionic autonomous navigation.

This paper puts forward a neural network model to decode the GCs' encoding information based on the improved traditional Kohonen self-organizing map (SOM), the details of which can be found in the section of Supplementary Materials. The model makes full use of GCs' firing characteristics and can be used as an effective and stable bionic position estimation method for unmanned vehicles. Therefore, this study is beneficial to explore the neural mechanism of animals' navigation behavior as well as to provide important reference to develop bionic autonomous navigation architecture for unmanned vehicles.

The rest of this paper is organized into four sections. In Section 2, we introduce the input data model of our proposed decoding model. Then, we analyze the relationship between index vectors and positions and set up a two-level self-growing self-organizing map (SGSOM) neural network model for the estimation of positions in Section 3. Section 4 is a simulation study of the proposed model and performance analysis. Finally, we conclude the paper in Section 5.

2. Data Model

A GC module is defined as a set of GCs with the same firing characteristics of grid period, grid orientation, and firing field size but a fixed grid phase relative to each other. Besides, the firing fields of all the GCs belonging to one GC module can completely cover the entire 2-D plane. Define that a GC module's period is the firing grid period of the GCs contained in the module. Hence, one GC module with a particular grid period can only encode the spatial positions

periodically and ambiguously through the firing of the GCs, but GC modules of different grid periods can jointly encode the spatial positions accurately [23]. This corresponds to the well-known residue number system (RNS) which is based on the Chinese remainder theorem (CRT) [24]. The details of the CRT are described as follows.

Given N pairwise coprime periods T_1, T_2, \dots, T_N as the moduli, let x be a positive integer, and R_1, R_2, \dots, R_N be the N remainders of x , i.e.,

$$R_i = \text{mod}(x, T_i), \quad (1)$$

where $1 \leq i \leq N$. Then, x can be uniquely reconstructed from its N remainders if and only if

$$0 \leq x < \text{lcm}\{T_1, T_2, \dots, T_N\} = \prod_{i=1}^N T_i, \quad (2)$$

where lcm stands for the least common multiple of the integers. The conclusion is also established if any pair of the moduli have a common greatest common divisor (gcd) T . Then, the range of x must be limited as

$$0 \leq x < \text{lcm}\{T_1, T_2, \dots, T_N\} = \frac{1}{T^{N-1}} \prod_{i=1}^N T_i. \quad (3)$$

The encoding process of spatial positions using GC modules is presented in Figures 2 and 3. We apply the competitive attractor network (CAN) model to simulate the firing patterns of GCs as they have a high degree of neural plausibility [25]. In order to generate individual GCs' firing patterns, the GCs of one module need to be arranged topologically as a rhombus to form a CAN with opposite edges cyclically connected as in Figure 2 [21].

Due to the dynamic of the network, an activity packet representing the neurons' firing rates will be formed and can be moved freely by the vehicle's running velocity. When considering one axis, e.g., the x -axis in Figure 2, the currently most active cell's (red dot) index x_1 along this axis provides the integral residue of the division of the current coordinate of the vehicle on this axis by the module's period T_1 . The information from one network, e.g., (x_1, y_1) , which can be named as the index coordinate, allows to locate the vehicle periodically and ambiguously. But, according to the well-known residue number system (RNS) which is based on the Chinese remainder theorem (CRT) [24], the vector $(x_1, y_1, x_2, y_2, \dots, x_N, y_N)$ provided by a set of NGC modules with N different periods in Figure 3, which are named as the multiperiod GCs' firing index vector (FIV), can jointly encode the spatial location accurately in a certain range [24]. Therefore, given N grid cell modules with N integer periods T_1, T_2, \dots, T_N , which satisfy the following conditions:

$$\begin{cases} 0 < T_1 < T_2 < \dots < T_N, \\ \text{gcd}(T_i, T_j) = 1, & (i \neq j, 1 \leq i, j \leq N), \end{cases} \quad (4)$$

and the radius r of the firing field, as illustrated in Figure 2, the number of grid cells on each side of the N diamond neural sheets can be calculated as

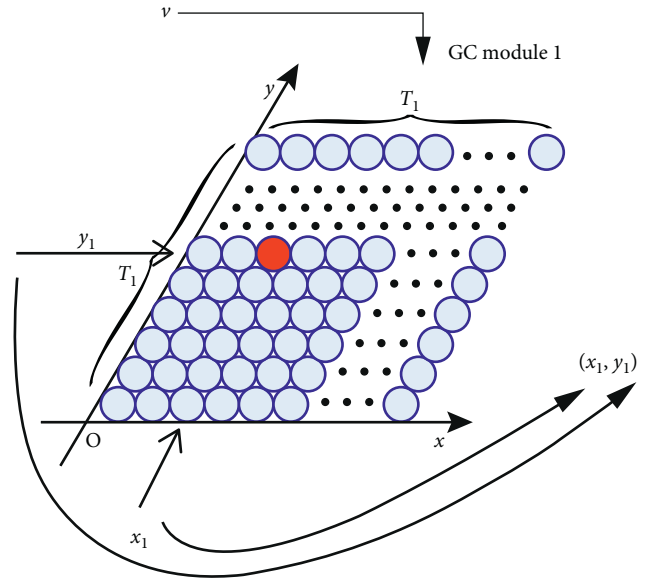


FIGURE 2: Process of generating the index coordinate in a GC module based on CAN (note that the angle of the two axes is 60 degrees). T_1 : period of the GC module; v : the vehicle's running velocity; red dot: the currently most active cell in the GC module.

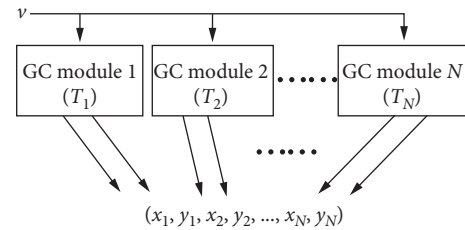


FIGURE 3: Process of generating the multiperiod GCs' firing index vector. T_i : period of the i th GC module; v : the vehicle's running velocity; N : number of GC modules.

$$k_i = \frac{T_i}{2r} \quad (1 \leq i \leq N). \quad (5)$$

Obviously, if the value of r is properly settled, such as $r = 0.5$ or 0.05 to guarantee that k_i ($1 \leq i \leq N$) is an integer, then we can get the following result:

$$\begin{cases} 0 < k_1 < k_2 < \dots < k_N, \\ \text{gcd}(k_i, k_j) = \frac{1}{2r} \quad (i \neq j, 1 \leq i, j \leq N). \end{cases} \quad (6)$$

According to the CRT and (3), the vector $\text{FIV} = [x_1, y_1, x_2, y_2, \dots, x_N, y_N]$ formed by N pairs of index numbers from N grid cell modules, which we defined as the index vector, can uniquely encode the animal's current spatial position as long as the position coordinate (x, y) in the xoy coordinate system satisfies

$$\begin{aligned}
0 &\leq x, \\
y &\leq 2r * \text{lcm}\{k_1, k_2, \dots, k_N\} \\
&= 2r * \prod_{i=1}^N \frac{k_i}{(2r)^{N-1}}.
\end{aligned} \tag{7}$$

Hence, the FIV can be used as the input data model of our proposed decoding model.

3. Position Estimation with SGSOM

GC modules with different grid periods jointly encode the spatial positions based on the CRT. Theoretically, the decoding of the information involves complex multiplications and modular inversion operations [24], which make it difficult to be implemented through neural network. Hence, based on the properties of the RNS and the characteristics of SOM, we propose a neural network model to estimate the current position of the vehicle according to the input FIV. The main structure of our model is shown in Figure 4. It consists of a two-level SGSOM neural network. The first-level SGSOM is an ordering process, and the topological order of input vectors is reproduced in the two-dimensional space. The second-level SGSOM is a 1-1 mapping process from one two-dimensional space to another two-dimensional space. Each cell of the second map represents a position. The details of each level will be discussed in the following sections.

3.1. First-Level SGSOM. The real-time requirement of position estimation determines that there are no sample vectors for the training of SOM when the vehicle begins to explore a new environment. Thus, the weight vectors of competitive layer neurons should be provided in advance rather than through training. This is also neural plausibility as many cells' firing patterns become stable when animals reach adult ages [16]. In addition, a fixed number of neurons for position estimation cannot adapt to scale changes of the environments. Therefore, we propose a SGSOM based on the characteristics of traditional SOM in our model. In contrast to traditional SOM, the number of its competitive layer neurons is not fixed. New neurons can be produced and added to the competitive layer of the SGSOM with the exploration of the environment, and the weight vectors of the newly added neurons are generated by prediction of the exploring trajectory. The network working steps are as follows.

Step 1. Initialize the network.

Generate the first neuron of the competitive layer and take the first input FIV as its weight vector:

$$W_1 = \text{FIV}_1, \tag{8}$$

where $W_1 = [u_1^1, v_1^1, u_2^1, v_2^1, \dots, u_N^1, v_N^1]$ is the weight vector of the first competitive layer neuron and $\text{FIV}_1 = [x_1^1, y_1^1, x_2^1, y_2^1, \dots, x_N^1, y_N^1]$ is the first input index vector.

Step 2. Calculate the Euclidean distance between the input vector and each competitive layer neuron's weight vector,

and identify the winning neuron which has the minimum distance.

Firstly, we introduce some properties of the RNS. Consider that x and y are two arbitrary integers and m and n are two positive integers as moduli with $m < n$, then

$$\begin{cases} r_x^m = \text{mod}(x, m), \\ r_x^n = \text{mod}(x, n), \\ r_y^m = \text{mod}(y, m), \\ r_y^n = \text{mod}(y, n). \end{cases} \tag{9}$$

According to the relative properties of the RNS, the following relationship is established:

$$\begin{aligned}
\langle x - y \rangle_{\{m, n\}} &= \{r_x^m, r_x^n\} \\
&= \{ \langle r_x^m - r_y^m \rangle_m, \langle r_x^n - r_y^n \rangle_n \},
\end{aligned} \tag{10}$$

where $\langle x - y \rangle_{\{m, n\}} = \{\text{mod}(x - y, m), \text{mod}(x - y, n)\}$ and $\langle x \rangle_m = \text{mod}(x, m)$. Suppose $d = x - y$. We can get the following conclusion:

$$\begin{cases} r^m = r^n = d, & \text{if } 0 \leq d < m, \\ m - r^m = n - r^n = -d, & \text{if } -m < d < 0. \end{cases} \tag{11}$$

These conclusions are still established when there are more than two moduli. Integers x and y can also be extended to multidimensional vectors, each dimension of which can be treated independently.

Assume that there are M neurons in the competitive layer with weight vector as $W_i (1 \leq i \leq M)$ and $\text{FIV} = [x_1, y_1, x_2, y_2, \dots, x_N, y_N]$ is the current input index vector. According to the above conclusions, we can have

$$\begin{aligned}
\langle \text{FIV} - W_i \rangle_{\{k_1, k_2, \dots, k_N\}} &= \{d_1^x, d_1^y, d_2^x, d_2^y, \dots, d_N^x, d_N^y\} \\
&= \{ \langle x_1 - u_1^i \rangle_{k_1}, \langle y_1 - v_1^i \rangle_{k_1}, \langle x_2 - u_2^i \rangle_{k_2}, \\
&\quad \cdot \langle y_2 - v_2^i \rangle_{k_2}, \dots, \langle x_N - u_N^i \rangle_{k_N}, \\
&\quad \cdot \langle y_N - v_N^i \rangle_{k_N} \}.
\end{aligned} \tag{12}$$

According to the geometric relationship and the cosine theorem, the Euclidean distance between FIV and $W_i (1 \leq i \leq M)$ can be calculated as

$$\text{dis}(\text{FIV}, W_i) = \sqrt{(d^x)^2 + (d^y)^2 - 2 * d^x * d^y * \cos\left(\frac{2\pi}{3}\right)}, \tag{13}$$

$$\begin{cases} d^j = d_1^j, & \text{if } d_1^j = d_1^j, \\ d^j = T_1 - d_1^x, & \text{if } T_i - d_1^j = T_i - d_1^j, \\ d^j = +\infty, & \text{other cases,} \end{cases} \tag{14}$$

where $i, i' = 1, 2, \dots, N; j = x, y$.

Figure 5 presents the calculation process of case 1. It is obvious that the less the Euclidean distance is, the closer the two vectors are, so does the two positions contained in the two vectors. The neuron which has the minimum Euclidean distance with FIV is considered as the winning neuron.

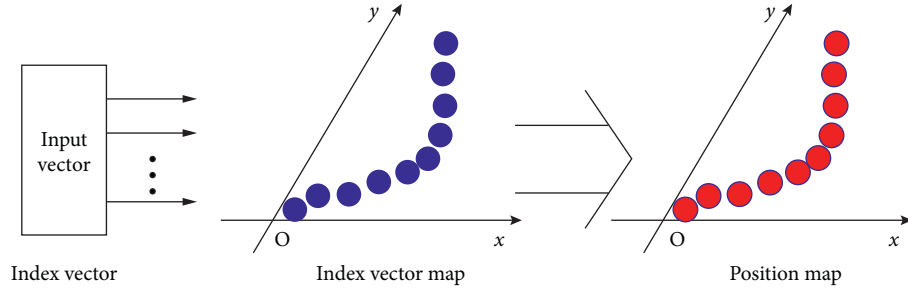


FIGURE 4: Architecture of the two-level SOM.

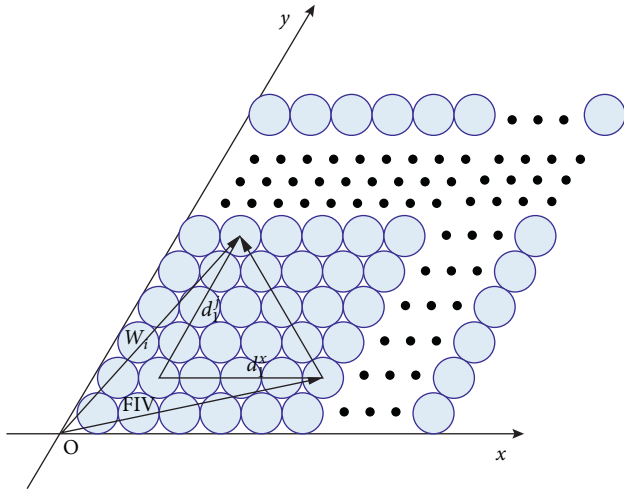


FIGURE 5: Calculation process of Euclidean distance of one case.

Suppose that the current perceived velocity is $v(t)$, the running direction is $\theta(t)$, and they remain unchanged during the next sampling interval Δt . Then, the displacement components along two axes of the xoy coordinate system during Δt can be predicted as

$$\begin{cases} d_x = v(t) * \Delta t * \cos(\theta(t)) - v(t) * \Delta t * \frac{\sin(\theta(t))}{\tan(\pi/3)}, \\ d_y = v(t) * \Delta t * \frac{\sin(\theta(t))}{\sin(\pi/3)}, \end{cases} \quad (15)$$

which are illustrated in Figure 6. In order to guarantee that there must be a winning neuron in this step, according to (11), the following expression must be satisfied:

$$\begin{cases} \text{abs}(d_x) < 2r * k_1, \\ \text{abs}(d_y) < 2r * k_1, \end{cases} \quad (16)$$

where $\text{abs}(d)$ means to take the absolute value of d .

Step 3. Predict a weight vector and add a new neuron to the competition layer.

The displacement presented in (15) can be transformed into an index vector based on the N grid cell modules as illustrated in Figure 6. Thus, we can get the displacement index vector as

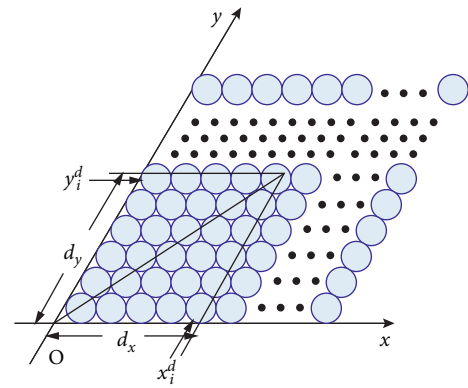


FIGURE 6: Displacement during a sampling interval and its transformation to an index vector.

$$v_d = \{x_1^d, y_1^d, x_2^d, y_2^d, \dots, x_N^d, y_N^d\}, \quad (17)$$

where

$$\begin{cases} x_i^d = \text{round}\left(\frac{\text{mod}(d_x, k_i)}{(2r)}\right), \\ y_i^d = \text{round}\left(\frac{\text{mod}(d_y, k_i)}{(2r)}\right), \end{cases} \quad (18)$$

where $i = 1, 2, \dots, N$, and $\text{round}(x)$ means to take the integer nearest x . Suppose that there are M neurons in the competitive layer, the j th neuron is the current winning neuron, and its weight vector is $W_j = [u_1^j, v_1^j, u_2^j, v_2^j, \dots, u_N^j, v_N^j]$, then the predicted weight vector of the neuron that may be added to the competitive layer can be written as

$$W = \{u_1, v_1, u_2, v_2, \dots, u_K, v_K\}, \quad (19)$$

where

$$\begin{cases} u_i = \text{mod}(x_i^d + u_i^j, k_i), & (i = 1, 2, \dots, N), \\ v_i = \text{mod}(y_i^d + v_i^j, k_i), & (i = 1, 2, \dots, N). \end{cases} \quad (20)$$

Check whether or not W exists in the weight vectors of the M competitive layer neurons. If the answer is yes, turn to step 2. Otherwise, add a new neuron to the competitive layer and set its weight vector as W and then turn to step 2.

Step 4. Repeat Steps 2 and 3 until the exploration of the environment is completed.

3.2. Second-Level SGSOM. The second-level SGSOM is a 1-1 mapping from the first level. They have the same number of neurons. Each neuron in this level represents a position, which is obtained according to the additions of the first-level neurons. The network working steps are as follows.

Step 1. Initialize the Network.

Generate the first neuron of this level, set its contained position coordinate as $(0, 0)$ or (x'_0, y'_0) in the Cartesian coordinate system, and associate it with the first neuron of the first level.

Step 2. Output the estimated position coordinate.

For any input index vector, output the position coordinate contained in the neuron which is associated with the winning neuron of the first level. This position coordinate can be directly used as the estimation of the vehicle's current position.

Step 3. Add a new neuron to this level.

If a new neuron is just added to the first level, then a new neuron should also be added to this level and associated with the newly added neuron of the first level. Suppose that a new neuron has been just added to the first level and second level, respectively, the j th neuron is the current winning neuron of the first level, and its associated neuron of the second level contains position coordinate (x'_j, y'_j) . According to the predicted index vector of the displacement shown in (17), the position coordinate (x', y') contained in the second level's newly added neuron can be written as

$$\begin{cases} x' = x'_j + \Delta x', \\ y' = y'_j + \Delta y', \end{cases} \quad (21)$$

where $\Delta x'$ and $\Delta y'$ are calculated as

$$\begin{cases} \Delta x' = \left(d^x + d^y * \cos\left(\frac{\pi}{3}\right) \right) * r, \\ \Delta y' = d^y * \sin\left(\frac{\pi}{3}\right) * r, \end{cases} \quad (22)$$

where d^x and d^y are calculated as

$$\begin{cases} d^j = d_1^j, & \text{if } d_i^j = d_{i'}^j, \\ d^j = d_1^j - T_1, & \text{if } d_i^j - T_i = d_{i'}^j - T_{i'}, \end{cases} \quad (23)$$

where $i, i' = 1, 2, \dots, N$; $j = x, y$.

4. Results and Analysis

In our model, the relationship between the firing of GCs and the vehicle's position is discussed, and we mainly focus on the position estimation performance that the proposed

model can decode from GCs' firing patterns. Hence, the analysis of this model is based on the comparison between the position information encoded by GCs' firing patterns and the position information decoded by the proposed model. Simulations are divided into two parts. First, position estimations are carried in linear motion and curvilinear motion. Second, the influence of model parameters is analyzed.

4.1. Realization of the Model. In this section, simulations are carried out to verify the effectiveness and stability of the proposed scheme. The simulation parameters are listed in Table 1. The performance of the model is evaluated by the position estimation error (PEE) which is defined as follows:

$$PEE = \sqrt{(x' - \hat{x}')^2 + (y' - \hat{y}')^2}, \quad (24)$$

where (x', y') and (\hat{x}', \hat{y}') are the real and estimated position coordinates of the vehicle in the rectangular Cartesian coordinate system. According to (15) and (16), in order to guarantee that the proposed model can work properly, the velocity should be limited by the following formula:

$$v * \Delta t < 2r * k_1 = 67 \text{ (m)}. \quad (25)$$

In other words, the value of velocity v should be less than 134 m/s.

Firstly, assume that the perceived velocity and running direction are exact and remain unchanged during a sampling interval and the running direction also remains unchanged during the total running time. Figure 7 presents an actual running trajectory and a trajectory formed by the estimated positions in linear motion. The running direction is taken as 45° relative to the x -axis. As illustrated, the two trajectories almost overlap each other, which indicates that the proposed model can decode the position information contained in the index vectors very well. Figure 8 presents the PEE of one trajectory and average PEE of 50 trajectories in a fixed running direction. Figure 9 presents the results when the vehicle runs in different directions. In this simulation, the vehicle runs along 50 trajectories with random velocity v which is less than 74 m/s in each direction. The results presented in Figure 9 are obtained by first averaging the PEEs of all the sampled positions for each trajectory and then averaging all the obtained average PEEs of 50 trajectories in each direction. The maximum of average PEEs is less than 0.045 m. The difference of average PEEs between different running directions is smaller than 0.02 m. The standard deviations of PEEs in different running directions are also offered based on these average PEEs, which is less than 0.005 m. Thus, these simulation results indicate that the proposed model is effective and stable in the case of linear motion. Simulations are also carried out when the vehicle runs randomly in a 2-D environment. Figure 10 presents a real running trajectory and its corresponding estimated running trajectory. The two trajectories also almost overlap each other, which indicates the proposed model can also decode the position information contained in the curvilinear motion. Figure 11 presents the PEE of one trajectory

TABLE 1: Simulation parameters.

Parameter	Value
Number of GC modules	3
Periods of GC modules	67 m, 68 m, 69 m
Radius of GCs' firing fields	0.05 m
Sizes of GC modules	670 * 670, 680 * 680, 690 * 690
Sampling interval	0.5 s
Total running time	500 s

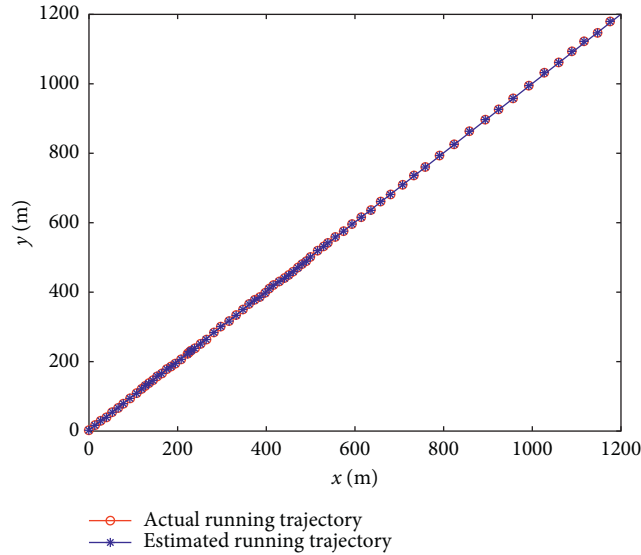


FIGURE 7: Actual running trajectory and estimated running trajectory in linear motion.

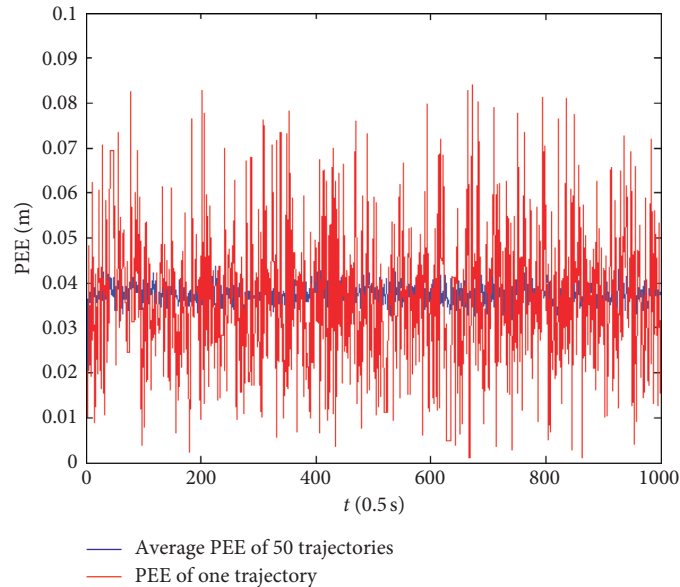


FIGURE 8: Average PEEs of 50 trajectories and PEEs of one trajectory in linear motion.

and average PEE of 50 random trajectories. The average PEEs and standard deviation of PEEs are also calculated for each trajectory in Figure 12. The difference of PEEs among different trajectories is still very small, which suggests that the

performance of the proposed position estimation model is stable.

To sum up, the proposed position estimation model based on the two-level SGSOM is effective and stable.

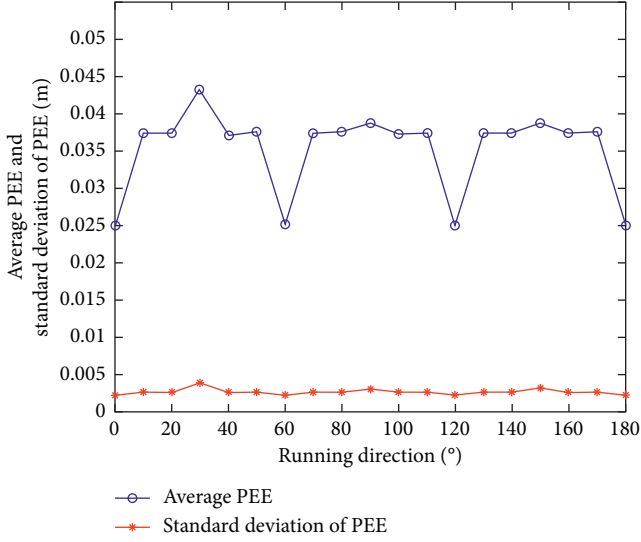


FIGURE 9: Average PEEs and standard deviations of PEEs of different running directions in linear motion.

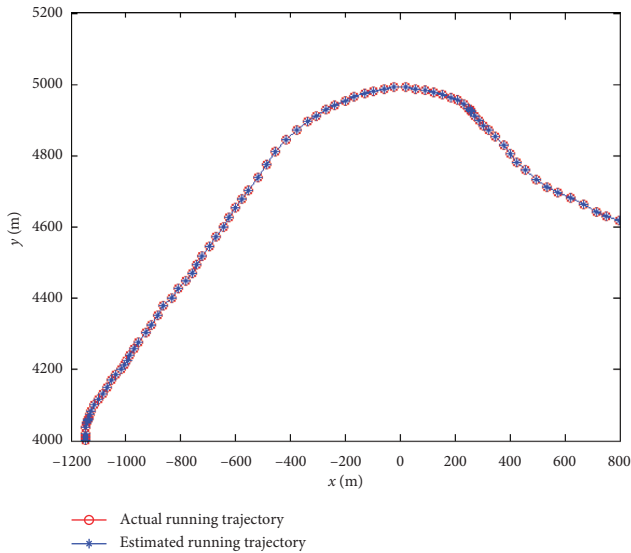


FIGURE 10: Actual running trajectory and estimated running trajectory in curvilinear motion.

4.2. Influence of Model Parameters on Position Estimation. The parameters related to position estimation results mainly include the number of grid cell modules, the grid periods of the grid cell modules, and the radius of the firing field. The sizes of grid cell modules depend on the above three parameters. Given N grid cell modules, there are N grid periods T_1, T_2, \dots, T_N . According to the CRT and (2), these three parameters determine the range of the explored environment within which our proposed model can work well. The greater these three parameters are, the larger the range of the explored environment is. In addition, according to (13) and (14), the minimum period T_1 as well as k_1 determines the max. velocity of the vehicle. The radius of grid cells' firing field can not only affect the range of the environment but also affect the PEE of the model. Figure 13 presents the

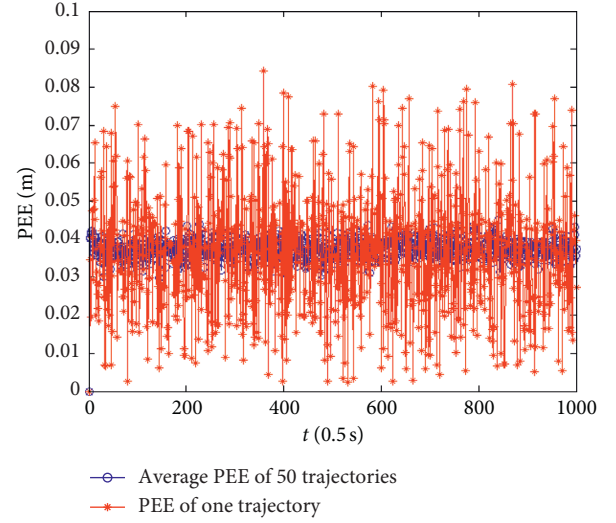


FIGURE 11: Average PEEs of 50 trajectories and PEEs of one trajectory in curvilinear motion.

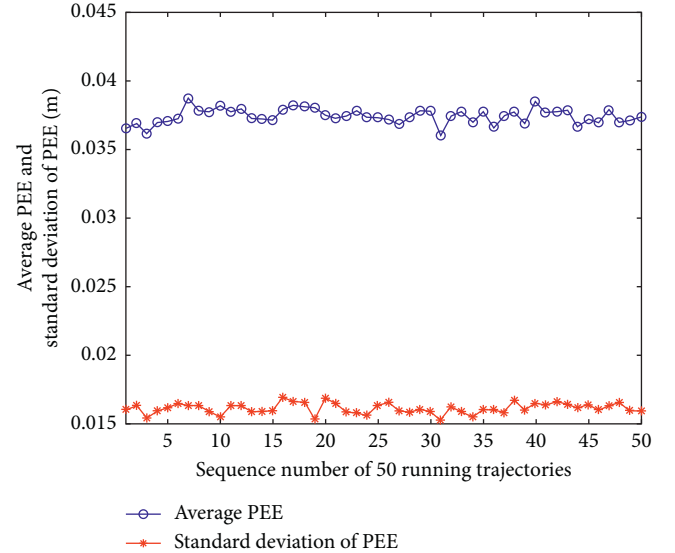


FIGURE 12: Average PEEs and standard deviations of PEEs of different trajectories in curvilinear motion.

simulation results according to three different radii of GCs' firing fields. The results show that the smaller the radius of grid cells' firing field, the smaller the average PEE and standard deviation of PEE are. But, it is obvious that the computation complexity of the model become larger with the decrease of the radius.

5. Conclusions and Discussion

In this paper, a scheme based on the firing patterns of GCs and SGSOM is set up for the estimation of positions. By using the topology preserving of SOM, we set up a map from firing index vectors to positions through SOM, which maps similar index vectors into adjacent positions. To illustrate the performance of the scheme, simulation experiments are done to test the model and the results indicate its

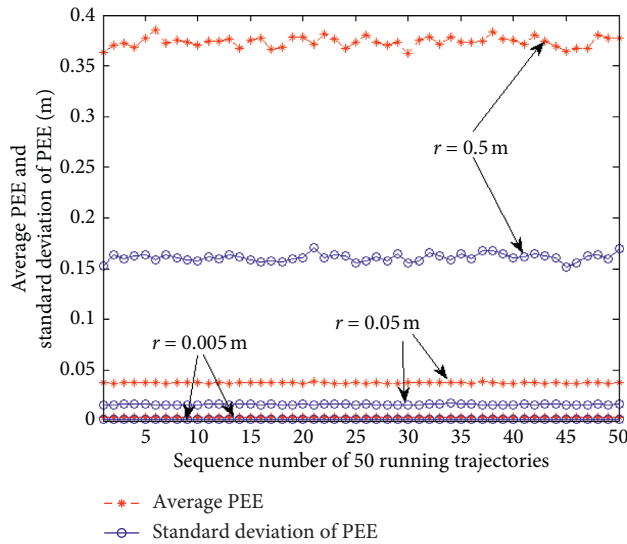


FIGURE 13: Average PEEs and standard deviations of PEEs of different firing field radii in curvilinear motion.

effectiveness and stability. Furthermore, the influence of model parameters on position estimation is also analyzed.

In this model, there is no need to carry out the complex multiplications and modular inversion operations like the solution of the CRT. When the vehicle arrives at a new place, just put the index vector generated by the grid cell models into the network, then the estimation of position coordinate can be obtained. Moreover, the size of SGSOM is changeable with the scales of the environments, which is very useful for conserving computation resource. In conclusion, the use of SGSOM network on position estimation is worthy of being applied in practice. Decades of research on the neurobiology of navigation focused on two-dimensional (2-D) navigation on flat surfaces, which laid the foundations for understanding the neural basis of 2-D spatial navigation. Thus, our research is also carried out in a 2-D environment. It is obvious that the results of our research can be directly extended to three-dimensional (3-D) environment.

In this paper, we mainly focus on the position estimation performance of the proposed model from the viewpoint of bionic navigation. As a decoder, our model can accurately restore the position information contained in the index vectors, and the estimation error can be neglected as for navigation. Therefore, we do not compare its performance with that of other different models of position estimation. Besides, considering that the perceived speed is not always exact due to kinds of factors such as noise in practical application, the position information contained in the index vectors and the decoded position coordinates will not be accurate compared with vehicles' real positions, and moreover, the errors will be accumulated in the future. This is a common and remarkable problem in the field of position estimation based on self-motion information. The main solutions for this problem are multisource information fusion methods such as loop detection based on vision information [26]. We will make further study on this topic in the next research work.

Data Availability

The data used to support the findings of this study are included within the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The project was supported by the National Natural Science Foundation of China (No. 61603412).

Supplementary Materials

The self-organizing map introduced in this paper is a Kohonen self-organizing map, which is also named as Kohonen feature map [27]. It is an unsupervised, competitive feed-forward neural network. SOM can map high-dimensional data into one- or two-dimensional data keeping the same topological order as the original data, and then features of the input data will be visualized. Thus, SOM is an effective method for feature extraction and dimensionality reduction and usually used for classification or prediction of different problems. Traditional SOM network consists of two layers: input layer and competitive layer which is also called output layer. The number of input layer nodes is equal to the input vector's dimension. The number of competitive layer neurons is fixed, and the neurons are usually arranged in a two-dimensional rectangle or hexagon grid. Each neuron of the competitive layer contains a weight vector that has the same dimension as the input vector. The whole network works as follows. When an input sample vector is put into the network, the Euclidean distance between this vector and weight vectors of the competitive layer neurons is calculated to obtain the winning neuron which has the minimum distance. Then, the weight vectors of the winning neuron and its neighboring neurons are adjusted to make them more similar to the input sample vector. Eventually, the weight vectors of competitive layer neurons have a certain distribution as input sample vectors by such training. (*Supplementary Materials*)

References

- [1] A. Bachrach, S. Prentice, R. He, and N. Roy, "RANGE-Robust autonomous navigation in GPS-denied environments," *Journal of Field Robotics*, vol. 28, no. 5, pp. 644–666, 2011.
- [2] G. Dissanayake, H. Durrant-Whyte, and T. Bailey, "A computationally efficient solution to the simultaneous localisation and map building (slam) problem," *IEEE Transactions*, vol. 17, no. 3, pp. 229–241, 2013.
- [3] M. W. M. G. Dissanayake, P. Newman, H. F. Durrant-Whyte et al., "An experimental and theoretical investigation into simultaneous localisation and map building," in *Experimental Robotics VI*, Springer, Berlin, Germany, 2000.
- [4] S. Huang and G. Dissanayake, "Convergence and consistency analysis for extended kalman filter based slam," *IEEE Transactions on Robotics*, vol. 23, no. 5, pp. 1036–1049, 2007.

- [5] B. Kuipers, "An intellectual history of the spatial semantic hierarchy," in *Springer Tracts in Advanced Robotics*, vol. 38, pp. 243–264, Springer, Berlin, Germany, 2007.
- [6] B. W. Roy, A. L. Bass, S. Chow et al., "Natal homing in juvenile loggerhead turtles (*Caretta caretta*)," *Molecular Ecology*, vol. 13, no. 12, article 3797IC3808, 2004.
- [7] S. R. Thorrold, C. Latkoczy, P. K. Swart et al., "Natal homing in a marine fish metapopulation," *Science*, vol. 291, no. 5502, pp. 297–299, 2001.
- [8] B. Ronacher, "Path integration as the basic navigation mechanism of the desert ant *cataglyphis fortis* (forel, 1902) (hymenoptera: formicidae)," *Myrmecological News*, vol. 11, pp. 53–62, 2008.
- [9] R. P. Kesner and E. T. Rolls, "A computational theory of hippocampal function, and tests of the theory: new developments," *Progress in Neurobiology*, vol. 79, no. 1, pp. 1–48, 2006.
- [10] D. Kim and J. Lee, *Path Integration Mechanism with Coarse Coding of Neurons*, Kluwer Academic Publishers, Dordrecht, Netherlands, 2011.
- [11] E. T. Rolls, "Representations in the brain," *Synthese*, vol. 129, no. 2, pp. 153–171, 2001.
- [12] J. Taube, "Head direction cells recorded in the anterior thalamic nuclei of freely moving rats," *Journal of Neuroscience*, vol. 15, no. 1, pp. 70–86, 1995.
- [13] S. S. Winter and J. S. Taube, *Head Direction Cells: From Generation to Integration*, Springer, Vienna, Austria, 2014.
- [14] T. Hafting, M. Fyhn, S. Molden, M.-B. Moser, and E. I. Moser, "Microstructure of a spatial map in the entorhinal cortex," *Nature*, vol. 436, no. 7052, pp. 801–806, 2005.
- [15] B. L. McNaughton, F. P. Battaglia, O. Jensen, E. I. Moser, and M.-B. Moser, "Path integration and the neural basis of the 'cognitive map'," *Nature Reviews Neuroscience*, vol. 7, no. 8, pp. 663–678, 2006.
- [16] C. Parron and E. Save, "Evidence for entorhinal and parietal cortices involvement in path integration in the rat," *Experimental Brain Research*, vol. 159, no. 3, pp. 349–359, 2004.
- [17] E. I. Moser and M.-B. Moser, "Grid cells and neural coding in high-end cortices," *Neuron*, vol. 80, no. 3, pp. 765–774, 2013.
- [18] L. M. Giocomo, M.-B. Moser, and E. I. Moser, "Computational models of grid cells," *Neuron*, vol. 71, no. 4, pp. 589–603, 2011b.
- [19] L. M. Giocomo, S. A. Hussaini, F. Zheng, E. R. Kandel, M.-B. Moser, and E. I. Moser, "Grid cells use hcn1 channels for spatial scaling," *Cell*, vol. 147, no. 5, pp. 1159–1170, 2011.
- [20] A. Samsonovich and B. L. McNaughton, "Path integration and cognitive mapping in a continuous attractor neural network model," *Journal of Neuroscience*, vol. 17, no. 15, pp. 5900–5920, 1997.
- [21] J. Conklin and C. Eliasmith, "A controlled attractor network model of path integration in the rat," *Journal of Computational Neuroscience*, vol. 18, no. 2, pp. 183–203, 2005.
- [22] H. Mhatre, A. Gorchetchnikov, and S. Grossberg, "Grid cell hexagonal patterns formed by fast self-organized learning within entorhinal cortex," *Hippocampus*, vol. 22, no. 2, pp. 320–334, 2012.
- [23] I. R. Fiete, Y. Burak, and T. Brookings, "What grid cells convey about rat location," *Journal of Neuroscience*, vol. 28, no. 27, pp. 6858–6871, 2008.
- [24] A. Omundi and B. Premkumar, *Residue Number System: Theory and Implementation*, Imperial College Press, London, UK, 2007.
- [25] A. Guanella, D. Kiper, and P. Verschure, "A model of grid cells based on a twisted torus topology," *International Journal of Neural Systems*, vol. 17, no. 4, pp. 231–240, 2007.
- [26] A. Martinelli, *Closed-Form Solution of Visual-Inertial Structure from Motion*, Kluwer Academic Publishers, Dordrecht, Netherlands, 2014.
- [27] J. Vesanto and E. Alhoniemi, "Clustering of the self-organizing map," *IEEE Transactions on Neural Networks*, vol. 11, no. 3, pp. 586–600, 2000.