Chapter 3
A Spike-Timing Based Integrated Model for Pattern Recognition

Abstract During the last few decades, remarkable progress has been made in solving pattern recognition problems using network of spiking neurons. However, the issue of pattern recognition involving computational process from sensory encoding to synaptic learning remains underexplored, as most existing models or algorithms only target part of the computational process. Furthermore, many learning algorithms proposed in literature neglect or pay little attention to sensory information encoding, which makes them incompatible with neural-realistic sensory signals encoded from real-world stimuli. By treating sensory coding and learning as a systematic process, we attempt to build an integrated model based on spiking neural networks (SNNs), which performs sensory neural encoding and supervised learning with precisely timed sequences of spikes. With emerging evidence of precise spike-timing neural activities, the view that information is represented by explicit firing times of action potentials rather than mean firing rates has received increasing attention recently. The external sensory stimulation is first converted into spatiotemporal patterns using latency-phase encoding method and subsequently transmitted to the consecutive network for learning. Spiking neurons are trained to reproduce target signals encoded with precisely timed spikes. It is shown that using a supervised spike-timing based learning, different spatiotemporal patterns are recognized by different spike patterns with a high time precision in milliseconds.

3.1 Introduction

Everyday we recognize plenty of familiar and novel objects. However, we know little about the underlying mechanism of the sophisticated computation involved in the recognition process of human nervous system. Throughout our brain, neurons propagate information by generating clusters of electrical impulses called action potentials (APs) [1]. Analogue stimuli are encoded into spatiotemporal patterns and the neural representation of external world is the basis for perception and reaction [2]. Different encoding methods have been proposed by researchers, and among these approaches rate-based encoding (rate codes) and spike-based encoding (temporal codes) are the most widely studied coding schemes [3, 4]. Traditionally, it is believed that

Q. Yu et al., *Neuromorphic Cognitive Systems*, Intelligent Systems
Reference Library 126, DOI 10.1007/978-3-319-55310-8_3

information is carried by the temporal average of spikes [5–7], and rate-based coding has been widely used in previous learning models such as performing stochastic gradient learning [8] and solving recognition problem relying on variance of input currents [9]. Although rate codes work well when the stimulus is constant or varying slowly, which is not common in real-world stimulations. Unlike the rate coding, temporal encoding schemes assume that information is carried by the precisely timed spikes, which provides more information capacity than the mean firing rate of neurons [10, 11]. It has been found that temporally varying sensory information such as visual and auditory signals is processed and stored with high precision in brain [12, 13], and precisely timed spikes are important for the integration process of cortical neurons [14]. Therefore, temporal codes can describe neural signal more precisely which enable us to exploit time as a resource for communication and computation in spiking neural networks.

Recent neurophysiological results show that the precision of temporal spikes may be triggered by the rapid intensity transients [15] and even a single spike can carry substantial information about visual stimuli [16]. The low response variability of retinal ganglion cells shows that the most important information of a firing event generated by visual neurons may be reserved by the time of the first spike and the number of spikes [17]. Furthermore, experimental results show that most information carried by spikes is the timing of the first spike after stimulus onset [16]. In human retina, visual signal from 10^8 photoreceptor cells are projected to 10^6 retinal ganglion cells (RGCs) in the form of spike trains [15]. Hence the information compression is indispensable during the projection. In addition, action potentials have been shown to be related to the phases of the intrinsic sub-threshold membrane potential oscillations [18, 19]. The phase locking between action potential and gamma oscillation has also been discovered in electric fish [20] and the entorhinal cortex [21]. Phase coding has been successfully utilized to perform sequences learning and episodic memory in hippocampus via phase precession in previous works [22–24]. The phase information of spikes is exploited within each receptive field. As each ganglion cell receives information from the photoreceptor cells in its receptive field, phase coding is used to reserve spatial information during compression as described in Sect. 2.2. Thus, we believe that the combination of temporal and phase coding offers a new way to implement the compression as well as to explain the compression process.

After sensory encoding, the neural system needs to learn neural signals that represent external sensory stimulation. Spike-based learning algorithms compute with firing times and make use of the inter spike intervals so that they are compatible with temporal codes. Hebbian synaptic plasticity has been viewed as the basic mechanism for learning and memory [25, 26], in which the synaptic efficacy is increased if the presynaptic neuron repeatedly contributes to the firing of postsynaptic neuron. As precise spike timing [27] and relative timing between pre- and post-synaptic firing [28] are discovered, learning with millisecond precision has received intensive interests. Spike-timing-dependent plasticity (STDP) is believed to play an important role in learning, memory and the development of neural circuits [29]. However, many existing learning models use rate codes as the neural representation of information, and learning with temporal codes remains an open research topic. The objective of learning is to train output neurons to respond selectively to inputs and generate desired output spike patterns by adjusting synaptic plasticity. Since the membrane potential of postsynaptic neuron is determined by the spikes of afferent neurons, the generation of postsynaptic spike is the result of the cooperative integration and synchronization of presynaptic input spikes [30, 31]. When the input spikes arrive in synchrony and a sufficiently large depolarization of postsynaptic membrane potential is achieved, a firing event will be triggered. Since we consider explicit desired patterns for recognition task, supervised learning is preferred due to its efficiency and accuracy. Moreover, growing evidences indicate that supervised learning is also employed in cerebellum and cerebellar cortex [32, 33]. It has also been demonstrated to be a successful form of learning to establish network with cognition functions [34, 35]. We adopt a spike-timing based supervised learning algorithm recently developed by [36], in which the error between the target spike train and the actural one is used as the supervisory signal. In addition, the firing intervals between pre- and postsynaptic spikes are recorded for synaptic plasticity modification, through which the actual output patterns approximate the desired output patterns gradually.

The contribution of this work is to bridge the gap between sensory encoding and synaptic information processing by proposing an integrated computational model with spike-timing based encoding scheme and learning algorithm. This helps to reveal the neural mechanisms starting from visual encoding to synaptic learning and the computational process in central nervous system. Such an encoding and learning algorithms in the proposed spike-based model are integrated in a consistent scheme: temporal codes. The encoding method provides a possible mechanism for converting visual information into neural signals. The spiking neurons are trained to classify spatiotemporal patterns based on the temporal configuration of spikes rather than firing rates of neurons.

This chapter is organized as follows: In Sect. 3.2, we introduce the general structure, encoding method and learning algorithm of the proposed integrated model. In Sect. 3.3, the performance and properties of the integrated model are demonstrated by numerical simulations. Section 3.4 reviews the related works while Sect. 3.5 concludes and discusses the limitations and extensions of the integrated model proposed in this work.

3.2 The Integrated Model

3.2.1 Neuron Model and General Structure

In our proposed integrated model, all neurons are modeled with the leaky integrate-and-fire (LIF) model [37], which is defined as:

$$\tau \frac{dV}{dt} = -(V - V_r) + R(I_0 + I_{in} + I_n) \tag{3.1}$$

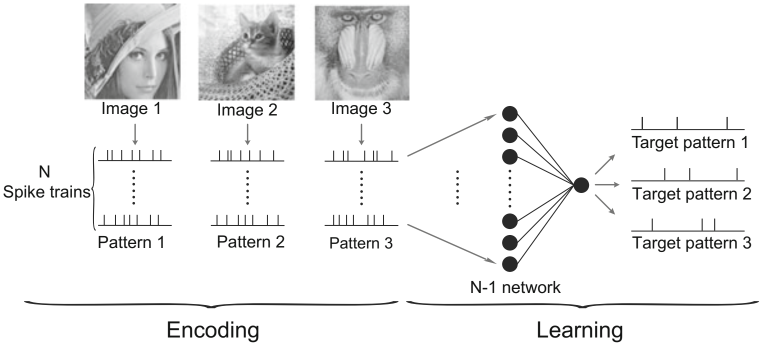

Fig. 3.1 General structure and information process of the integrated model. The main components of the model are the encoding part and the learning part. The spike-based model employs temporal codes as the neural representation of external information. The latency-phase encoding as discussed in Sect. 2.2 is used to convert the image into spatiotemporal patterns consisting of N spike trains. After sensory encoding, each spike train is received by one input neuron of the spiking neural network. With a predefined target pattern for each input pattern, the spiking neural network equipped with a supervised spike-timing based learning as described in Sect. 2.3 is trained to recognize the different spatiotemporal patterns

where $\tau = RC$ is the membrane time constant, $C = 1\,\text{nF}$ is the membrane conductance, $R = 10\ \text{M}\Omega$ is the membrane resistance, V is the membrane potential and $V_r = -60\,\text{mV}$ is the rest potential, $I_0 = 0.1\,\text{nA}$ is the constant inject current, I_{in} is the summation of presynaptic input currents, and I_n is a background noise modeled as a Gaussian process with zero mean and variance 1 nA. Once the membrane potential reaches the threshold $V_{thr} = -55\,\text{mV}$, it will be reset to $V_{res} = -65\,\text{mV}$ and held there for the refractory period.

The spike-based model presented here consists of two components: the latency-phase encoding and the supervised spike-timing based learning. Starting from environmental stimuli, we first encode images into spatiotemporal patterns and then transmit them to a spiking neural network for learning. The entire structure of the model is illustrated in Fig. 3.1.

3.2.2 Latency-Phase Encoding

With a combination of temporal encoding and phase encoding, a feature-dependent phase encoding algorithm has been proposed in [38]. Inspired by the information processing in the retina, the visual information is encoded into the responses of neurons using precisely timed action potentials. The intensity value of each pixel is converted to a precisely timed spike via a latency encoding scheme. Various experiments show that a strong stimulation leads to a short spike latency, and a weak stimulation results in a long reaction time [39–41]. Therefore, a monotone decreasing function could be used for the conversion from sensory stimuli to spatiotemporal patterns. Here, a logarithmic intensity transformation is adopted, which is similar to that used in [42].

$$t_i = f(s_i) = t_{max} - ln(\alpha \cdot s_i + 1) \tag{3.2}$$

where t_i is the firing time of neuron i, t_{max} is the maximum time of encoding window, α is a scaling factor, and s_i is the intensity of the analog stimulation. One advantage of the logarithmic function is that the time differences of spike latencies are invariant with different contrast level, e.g., it depends on the relative strength of the stimulation.

Ganglion cells have been observed to be firing in synchrony in several species [43–45], which illustrates the involvement of oscillations in the retina. We assume that the phases of oscillations are related to action potentials and contribute to the information compression from photoreceptor cells to ganglion cells. To take advantage of the phase information, spikes are assigned with phases related to their respective oscillations. Since each ganglion cell receives spikes from a group of photoreceptor cells, which is defined as the receptive field of this ganglion cell, we assign different initial phases to their subthreshold membrane oscillations. The periodic oscillation is described as cosine function for simplicity,

$$i_{osc} = A\cos(\omega t + \phi_i) \tag{3.3}$$

where A is the magnitude of the subthreshold membrane oscillations, ω is the phase angular velocity of the oscillation, and ϕ_i is the phase shift of the ith neuron in the receptive field.

In order to distinguish photoreceptor cells in the same receptive field, we set a constant phase gradient among photoreceptor neurons. The phase of subthreshold membrane oscillation for the ith photoreceptor neuron ϕ_i is defined as:

$$\phi_i = \phi_0 + (i-1) \cdot \Delta\phi \tag{3.4}$$

where ϕ_0 is the reference initial phase, and $\Delta\phi$ is the constant phase difference between nearby photoreceptor cells ($\Delta\phi < \frac{2\pi}{N_{RF}}$, N_{RF} is the number of photoreceptor cells in each receptive field).

The spikes generated by the photoreceptor cells in each receptive field are compressed into one spike train by the ganglion cell. In order to utilize the phase information of spikes to reconstruct the original visual stimuli, the alignment operation is required to link each spike in the spike train with the corresponding photoreceptor cell in the receptive field. The alignment procedure is implemented by forcing photoreceptor cells to fire only when the subthreshold membrane potentials reach their nearest peaks as illustrated in Fig. 3.2b, c. After compression as shown in Fig. 3.2c, d, each spike in the compressed spike train is linked to one particular photoreceptor cell in the receptive field according to the phase of the subthreshold oscillations. Consequently, the phase information and the alignment together build an one-to-one relationship between the photoreceptor cells and spikes generated by the corresponding ganglion cell. With the latency-phase coding scheme, external stimulation is encoded into precisely timed spikes and then compressed into spike trains. The intensity information is encoded into firing times while the spatial information is reserved by the phases of spikes. When the spike trains are transmitted to coupled networks with respect to the encoding area, latency-phase encoded spikes generated by photoreceptor cells can be reconstructed from the compressed spike trains with a same phase reference as shown in Fig. 3.2d, e. The visual stimulus can then be reconstructed via a simple latency decoding process as shown in Fig. 3.2e, f. The complete latency-phase scheme is illustrated in Fig. 3.2.

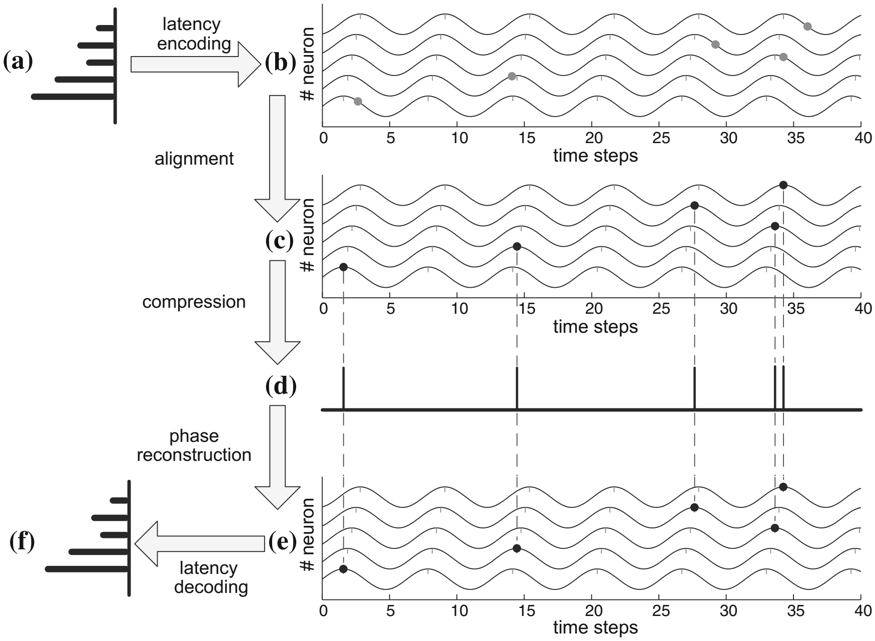

Fig. 3.2 Flowchart of the latency-phase encoding scheme. (**a**) Original stimuli. Stimulations with different intensities are the inputs to the photoreceptor cells. (**b**) The latency-encoded pattern. The visual information carried by the intensities is converted into the latencies of spikes. The spikes are assigned with phase information according to their corresponding oscillations. (**c**) Encoded spikes after latency encoding and alignment operation. The spikes are forced to be generated at peaks of the sub-threshold oscillations. (**d**) Compressed spike train. The spikes generated by the photoreceptor cells from the same receptive field are compressed into a spike train. (**e**) Reconstructed latency-encoded spikes. Spatial information within the receptive field could be retrieved from the compressed spike train via a phase reconstruction. (**f**) Decoded stimuli. By an inverse latency transformation, the original stimuli are reconstructed from the reconstructed spikes [38]

3.2.3 Supervised Spike-Timing Based Learning

It is known that learning from instructions is an important way for our brain to obtain new knowledge. As proposed in [36], the remote-supervised-method (ReSuMe) is compatible with temporal codes and is capable of performing spike-timing based learning precisely with millisecond timescale. The learning algorithm is based on a STDP-like process and synaptic modification during training depends on the pre- and postsynaptic firing times. After the training is successful, responses of output neurons will converge to the target patterns with a high time precision.

It is common that error signal between the target and the actual output is used in supervised learning. Similar to Widrow-Hoff rule applied in rate-based neuron models [46], the modification of synaptic efficacy in ReSuMe is triggered by either the target output ($S_d(t)$) or the actual output ($S_o(t)$). At the same time, the sign of error signal ($S_d(t) - S_o(t)$) decides the direction of the modification. To take the spike-timing into consideration, a STDP-like term is incorporated in the kernel a_{di}:

$$a_{di}(-s) = A \cdot exp(\frac{s}{\tau}), \quad \text{if } s < 0 \tag{3.5}$$

where A is the maximal magnitude of the STDP window, and s is the delay between the pre- and postsynaptic firing. Similar to the STDP process, if a presynaptic spike precedes a postsynaptic spike within a time interval, the synapse is strengthened. When the phase relation is reversed, the synapse is weaken. The magnitude of modification is determined by the lag s between pre- and postsynaptic spikes and is calculated by the convolution $a_{di}(t) * S_i(t)$. The complete learning rule is described as in Ponulak and Kasinski [36],

$$\frac{d}{dt}w_{oi}(t) = [S_d(t) - S_o(t)][a_d + \int_0^{\infty} a_{di}(s)S_i(t-s)ds] \tag{3.6}$$

where w_{oi} is the synaptic weight from the presynaptic neuron i to the postsynaptic neuron o. $S_d(t)$, $S_o(t)$ and $S_i(t)$ are the desired output, actual output and input spike train, respectively. a_d is a constant that helps speed up the learning process. From Eq. (3.6), we can see that the synaptic weights are updated when $S_d(t) \neq S_o(t)$, and the direction of modification is determined by the sign of the error signal $S_d(t) - S_o(t)$. No modification is induced when the actual output pattern is in agreement with the desired output pattern ($S_d(t) = S_o(t)$), which is used as the stopping criterion. The magnitude of modification is determined by the convolution term $a_{di}(t) * S_i(t)$. Thus, $S_i(t)$, $S_d(t)$ and $S_o(t)$ together are responsible for the synaptic modification. The learning rule is illustrated in Fig. 3.3.

The supervised signal is generated by the remote supervision scheme. Therefore, the target spike train is not directly delivered to the postsynaptic learning neuron and it determines the change of the synaptic efficacy from the presynaptic neuron

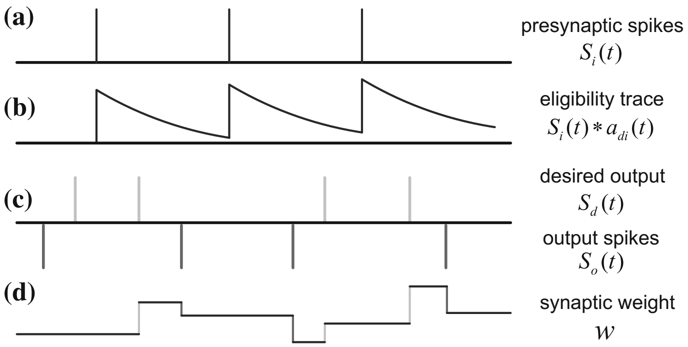

Fig. 3.3 Learning rule of ReSuMe. (**a**) The presynaptic input spikes, (**b**) The eligibility trace, (**c**) The desired output and actual output spikes, (**d**) The synaptic weight. The eligibility trace in (**b**) records the status of neuron according to the presynaptic spikes in (**a**). The desired output (positive direction) and the actual output (negative direction) in (**c**) together determine the sign of the supervisory signal. There is no other modification when the actual output spikes are generated at the desired times. The synaptic weight is updated when either a actual spike is generated or a desired spike should be induced. Meanwhile, the amount of synaptic weight change depends on the lag between pre- and postsynaptic spikes and the eligibility trace in (**b**) [36]

to postsynaptic neuron. It should be noted that both the excitatory synapses and inhibitory synapses exist in the model. During the learning, the synaptic weight is modified when either a target spike is needed or the postsynaptic learning neuron fires at the wrong time. When the modification occurs, the sign of error signal $(S_d(t) - S_o(t))$ decides the direction of change and the kernel $a_d + \int_0^\infty a_{di}(s)S_i(t-s)ds$ decides the amount of weight change. The synapses contributing to the firing of desired spikes are excitatory and adjusted to bring forward or hold off the firing times. On the other hand, the inhibitory synapses are used to suppress the firings at undesired times. The learning process stops as soon as the actual output patterns are identical to the target patterns.

3.3 Numerical Simulations

Real-world visual stimuli are often complex and contain a large amount of information. In this section, three 256 × 256 grayscale images are used to demonstrate the classification capability and the robustness of the integrated model. Images from the Urban and Natural Scene Categories of the LabelMe data set [47] are used here to explore the influence of parameter variations and the memory capacity of the system.

3.3.1 Network Architecture and Encoding of Grayscale Images

The receptive field (RF) of a sensory neuron is defined as a spatial region where the presence of stimulus affects the firing of that neuron. During the encoding phase, visual information from photoreceptor cells in the same RF is projected to retinal ganglion cells. Each ganglion cell then compresses the received spikes into a spike train. Therefore, the number of spikes in each spike train is determined by the number of pixels in each input image and the number of RFs.

$$N_{spike} = \frac{n}{N_{RF}} \tag{3.7}$$

where N_{spike} is the number of spikes in each spike train (number of pixels in each subfield assigned with an RF), n is the number of photoreceptor cells (number of pixels of each image), and N_{RF} is the number of retinal ganglion cells (i.e., the number of RFs). Since each ganglion cell connects to one input neuron of the consecutive spiking neural network, the number of input neurons N is equal to N_{RF}. The number of output neurons depends on the size of data sets and the readout strategy. Intuitively, for large database with a large number of classes and complex target patterns with more spikes, more output neurons are required to perform the learning task. A two layer spiking neural network with 1024 input neurons and a single output neuron is used to illustrate the recognition capability of this model.

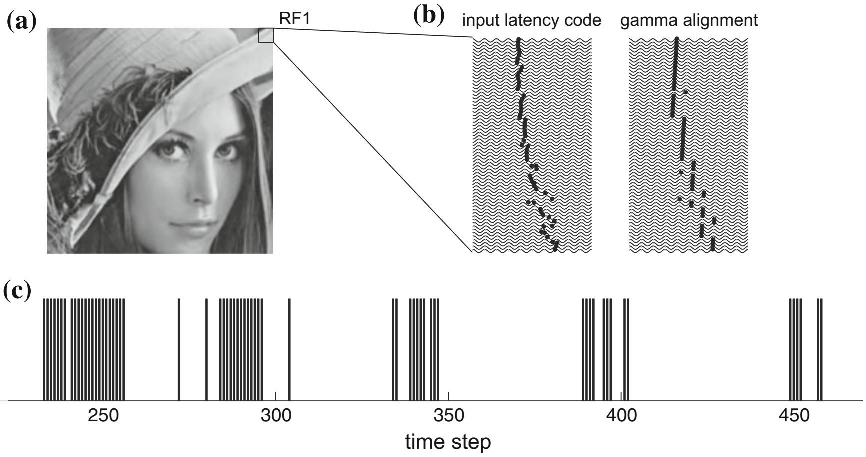

Fig. 3.4 The latency-phase encoding. The original image (256 × 256 pixels) in (**a**) is partitioned into 1024 RFs with the size of 8 × 8. The left pattern in (**b**) is the spike pattern of RF1 after latency encoding and the right one is the pattern further processed by the alignment operation (spikes are denoted by the dot markers). The compressed spike train of RF1 is given in (**c**). For better visualization, only part of the encoded spatiotemporal pattern is illustrated

Here, grayscale images with the size of 256 × 256 pixels are used as the external stimulation. Each pixel value is regarded as the intensity of the visual stimulation received by the photoreceptor cell in the retina. Thus there are 1024 RFs with the size of 8 × 8 pixels as shown in Fig. 3.4a. After the alignment as shown in Fig. 3.4b, each ganglion cell receives 64 spikes from 64 photoreceptor cells in its receptive field and compresses them into one spike train as shown in Fig. 3.4c. Therefore, information of the 256 × 256 pixel image is encoded into 1024 spike trains and each spike train contains 64 spikes. As the encoding method converts the intensity values into firing times of spikes, the visual information is preserved by the temporal configuration of the spike trains.

3.3.2 Learning Performance

To recognize images, we predefine different target spike patterns for input patterns. For simplicity, each target pattern is defined as a sequence of three spikes (each target pattern is denoted by a different marker type, as shown in Fig. 3.5a). After sensory encoding, three spatiotemporal patterns of length 640 ms are repetitively presented to the network in a random sequence. The number of epoch is increased when one pattern has been presented to the network, while the number of iteration is increased when all patterns have been presented to the network once. The responses of the output neuron for different input patterns are shown in Fig. 3.5a. To quantitively evaluate the learning performance, a correlation-based measure of spike timing [48] is adopted to measure the distance between the output pattern and the target pattern. The correlation C is close to unity when the output pattern matches the target pattern and equals to zero when the two patterns are unrelated. The spike trains (S_o and S_d) are convolved with a low pass Gaussian filter of a given width $\sigma = 2$ ms. If the filtered spike trains are $\overrightarrow{s_1}$ and $\overrightarrow{s_2}$, the correlation measure is

$$C = \frac{\overrightarrow{s_1} \cdot \overrightarrow{s_2}}{|\overrightarrow{s_1}||\overrightarrow{s_2}|} \tag{3.8}$$

The typical results of the training are shown in Fig. 3.5. Within 20 presentations of each input pattern, the output neuron is able to reproduce the target pattern as shown in Fig. 3.5. At first, the output neuron fires at random times. After several iterations, extra spikes firing at undesired times disappear, and the actual output patterns approach to the corresponding target patterns. When successful learning is achieved, the output neuron is able to reproduce different target patterns when different input patterns are given. We repeated the training for dozens of times and observed that the spiking neuron is able to learn the training pairs successfully.

3.3.3 Generalization Capability

The integrated model recognizes each image as a certain spatiotemporal pattern, in which the intensities of individual pixels are encoded into precisely timed spikes. Therefore, the generalization of the system is expected to be related to the pixel-level features of the input images. To study the generalization capability of the model, we add different levels of Gaussian, speckle and salt-and-pepper noise to the input images during the testing phase. The Gaussian noise is specified by its mean m and variance v, the speckle noise is specified by its variance v, and the salt-and-pepper noise is specified by the noise density d. For each kind of the noise with different intensities, we test the trained network with one hundred noisy images. The test

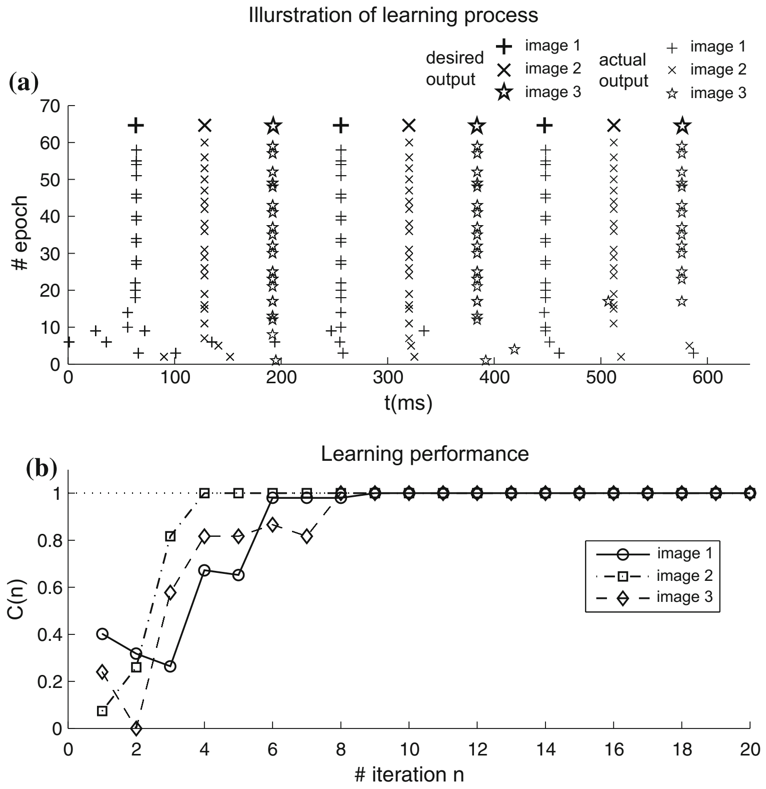

Fig. 3.5 Illustration of the learning process and performance. (**a**) Raster plot of the output spikes. When presented with different input patterns, the output patterns converge to the corresponding target patterns. Given different input patterns, spikes generated by the output neuron are denoted by different marker types. (**b**) The correlations C between output spike trains and the target spike trains against learning iterations. At first, the output neuron fires at random times. After several iterations, the output patterns begin to approach to the target patterns and the learning is converged within twenty iterations

results are shown in Fig. 3.6b. By analyzing the learning process, we can see that the pixel-feature dependent generalization is related to temporally local learning algorithm. During the learning process, only the synaptic weights associated with input spikes evoking the postsynaptic spikes within the learning window are updated. The decaying learning window makes the optimization process to be focused on a limited number of synapses, which affects the firing time of the nearest postsynaptic neuron. At the same time, noise added to input images shifts part of the firing times of the encoded spatiotemporal pattern. Therefore, the spiking neuron should be able to reproduce target spikes with a small temporal error in response to the input images with pixel noise, but fail to recognize images in the presence of other type of noises. As expected, the test results in Fig. 3.6b show that the system is more resistant to salt-and-pepper noise than speckle noise or Gaussian noise.

We also add the different type of noises to the input images during the training phase. For each type of noise, 100 × 3 noisy images are used as the training set. After

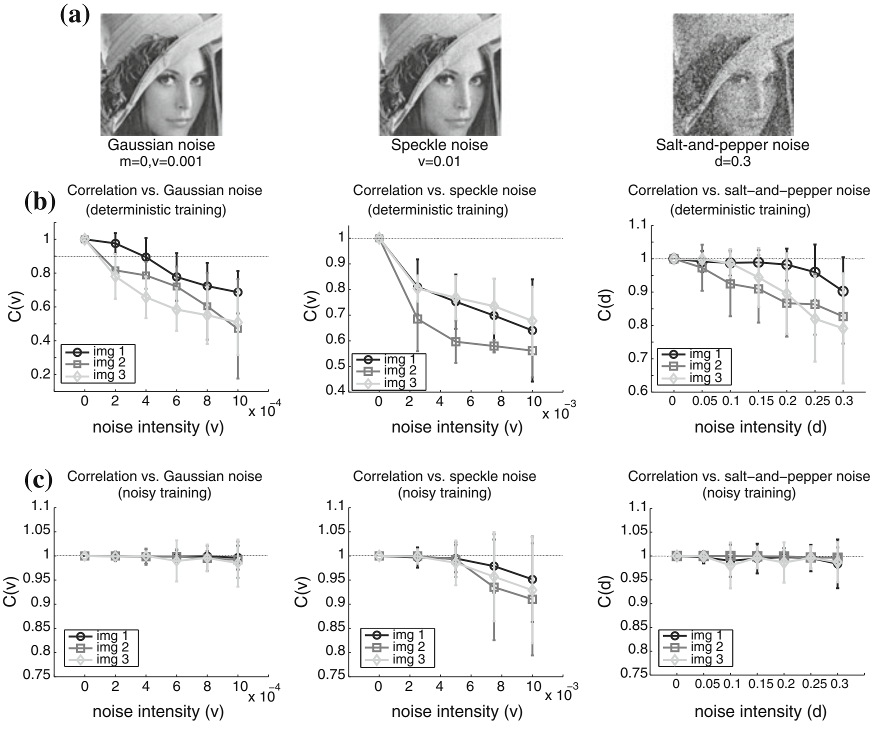

Fig. 3.6 The test results with different type of noises added to the input images. (**a**) Examples of images with different type of noises, such as Gaussian, speckle and salt-and-pepper noise. The correlation C between the output spike pattern and the target pattern is used to evaluate the precision of the neural responses. (**b**) Reliable responses can be reproduced by the spiking neural networks for noisy images (e.g., deterministic training). (**c**) The robustness to noise is improved when the noise information is included during the training phase (e.g., noisy training)

training, another 100 × 3 images with noise of the same type and intensity level are used to examine the reliability of the neural responses after noisy training. As shown in Fig. 3.6c, when the noise information is learned by the classifier during training phase, the robustness of the system due to the effect of noise has been improved. It can also be observed that the maximum level of salt-and-pepper noise that the system can tolerate is much higher than that of the other two type of noises, which is consistent with our analysis.

3.3.4 Parameters Evaluation

To examine the influence of parameter variations in the encoded patterns, 100 images (256 × 256 pixel, 8-bit grayscale) from the Urban and Natural Scene Categories of the LabelMe database are encoded with various parameter configurations. The images from LabelMe data set are used here to study the properties of the integrated model due to their distributed intensity values and their closeness to real-world stimulation. A few sample images from the data set are given in Fig. 3.7.

The size of receptive field, encoding cycles and phase shift constant are important parameters for the encoding method. Since photoreceptor cells of the same RF convey visual information to the corresponding retinal ganglion cell, the number of photoreceptor cells in each RF affects the number of spikes in the compressed spike train. If the length of encoding window is fixed, increasing the RF size would result in a higher average firing rate of the compressed spike trains.

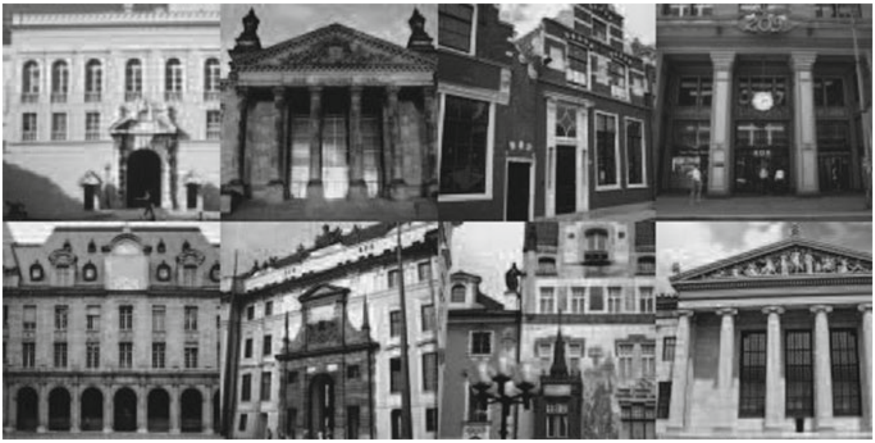

Fig. 3.7 Sample images of "buildings inside city" category from the LabelMe database. The original 256 × 256 color images are converted into 8-bit grayscale images

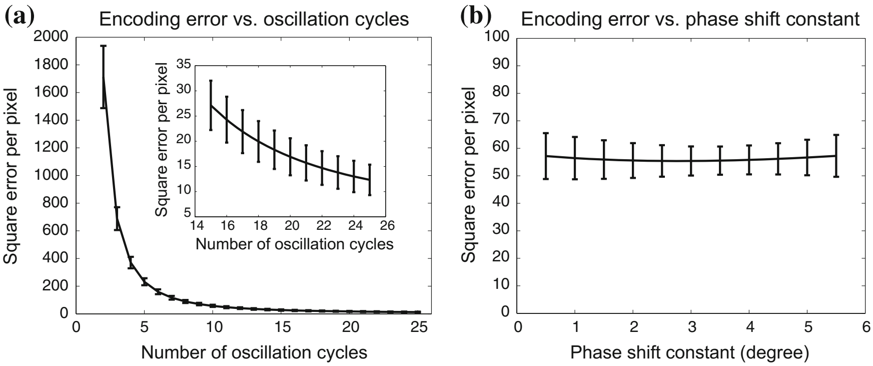

Fig. 3.8 The encoding error with different encoding cycles and phase shift constants on natural images from the LabelMe database. The average square error per pixel (vertical axis) is employed to estimate the encoding accuracy of the test images. (**a**) The encoding error drops when the number of oscillation cycles increases. With more subthreshold membrane oscillation cycles, more oscillation peaks provide more sampling points to encode input intensities (the tail of the curve is enlarged in the inset). (**b**) The phase shift constant $\Delta\phi$ slightly affects the encoding accuracy

Considering the accuracy of encoding process, no error is introduced by the latency encoding scheme. The distortion of information is resulted from the alignment operation. As the alignment operation moves spikes to the peaks of the subthreshold oscillations, the encoding accuracy is affected by the number of oscillation cycles within the encoding period as shown in Fig. 3.8a. To estimate the accuracy of encoding, we compare the reconstructed images with the original images using the average square of error per pixel,

$$e = \frac{\sum_{i=1}^{n} (s_i - s_i')^2}{n} \tag{3.9}$$

where s_i and s_i' are the intensities of the ith pixel in the original image and the reconstructed image, respectively.

Since the intensity information is carried by the temporal spikes, the distribution of the original images as well as the encoding parameters such as phase shift resolution $\Delta\phi$ may affect the temporal distribution of the encoded spatiotemporal patterns. The experiment results illustrate that the phase shift constant hardly affects the encoding accuracy as shown in Fig. 3.8b. However, it will determine the spike distribution of the compressed spike train as shown in Fig. 3.9. The encoded spikes concentrate in the time domain with a small shift constant as shown in Fig. 3.9a and spread out with a large shift constant as shown in Fig. 3.9b.

Therefore, the choice of encoding cycles depends on the precision requirement for a specific application. Since the phase shift resolution $\Delta\phi$ affects the distribution of encoded spatiotemporal patterns, it should be tuned according to the learning algorithm adopted in the posterior neural network.

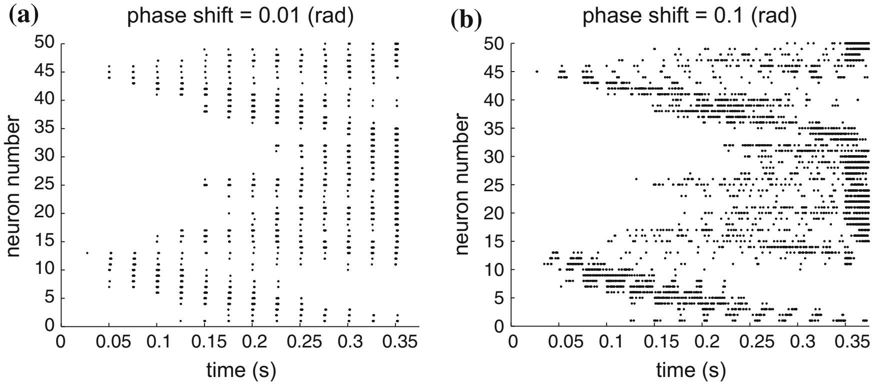

Fig. 3.9 The encoded patterns with a different phase shift constant. The phase shift constant is the phase difference between nearby photoreceptor cells in the same receptive field and affects the firing times within each receptive field. With a small phase shift constant, neurons within the same receptive field tend to fire simultaneously as shown in (**a**). With a large phase shift constant, the temporal distribution of spikes is scattered as shown in (**b**)

Since the postsynaptic depolarization is determined by the integration of presynaptic input spikes, temporal distribution of input spatiotemporal patterns and the complexity of target patterns will affect the learning performance. On one hand, because a target spike requires one or more preceding input spikes to excite the output neuron to fire at the desired time, enough presynaptic input spikes are needed for the generation of spikes. On the other hand, increasing the number of target spikes will result in competition for limited available synapses between the target spikes firing at different times and impose restriction on the behavior of the output neuron. We tested the system on 100 images (128 × 128 8-bit grayscale images from Urban and Natural Scene Categories of LabelMe database) to examine the influence of target patterns on the learning performance. For each number of target spikes, the network was trained with one randomly generated target pattern. It is observed that the spiking neuron needs more iterations to achieve a successful learning for a more complex target patterns as discussed in our analysis.

3.3.5 Capacity of the Integrated System

The spiking neural network with the same settings in previous experiments is used to explore the memory capacity of the integrated system. From a computational point of view, precisely timed spikes have a remarkable encoding capacity, i.e., the memory capacity of the system is often limited by the learning scheme employed. Since most of the information is reserved by the temporal code, the design of target patterns plays a pivot role in exploiting the information carried by the encoded spatiotemporal

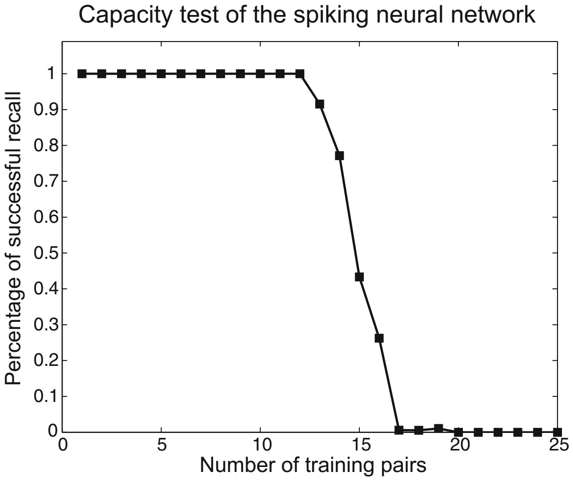

Fig. 3.10 Memory (or recognition) capacity of the integrated model. The average percentage of successful recall of patterns is plotted as a function of training pairs. The successful recall percentage drops dramatically after the number of training pairs is larger than 11

patterns. We train the network with different number of input patterns and define the percentage of successful recall of trained pairs as an evaluation of the memory capacity. A successful recall of one trained pattern is achieved when the distance between the output pattern of the trained network and the target pattern is close enough, i.e., $C > 0.95$ as the threshold. To simplify the problem for a classification task, we randomly generated one target spike train containing ten spikes for all input images every time and repeat the experiment for 20 times.

As shown in Fig. 3.10, for the 1024-1 spiking neural network with ten spikes in the target patterns and the selected parameter settings, around 11 training pairs can be successfully stored and recalled with a slight time shift. The percentage of successful recall decreases quickly when the number of training pairs is increased. Apparently, it can be inferred that decreasing the number of target spikes (complexity) or increasing the free tunable parameters will lead to a larger amount of information capacity. However, this would also allow less information of the spatiotemporal patterns to be learned. Although it is not mathematically analyzed, the presented simulation results for the specific case provide some insight into the information capacity of the system.

To summarize it from a system level, temporally distributed input spatiotemporal patterns and simple target patterns are preferred for better generalization capabilities and memory capacity of the integrated model. The scattered distribution of input patterns enables the output neuron to generate spikes at arbitrary times. Although the network can learn more about the original images with more complex target patterns, the computational efforts will also be increased and the information capacity will be limited. Therefore, the tradeoff between the learning level of input patterns and the computational efforts as well as memory capacity should be considered for any specific applications.

3.4 Related Works

Spiking neural networks have been applied to solve different classification tasks [31, 49–52]. Hopfield and Brody [30] proposed a computational model for pattern recognition, in which analog signal is employed as neural representation of sensory stimuli. The transient synchronization of decaying delay activity of a specific subset of input neurons are used for recognition. Although it has been successfully applied to speech recognition [31] and olfactory recognition [49], the unknown mechanism of encoding input stimulation into decay firing activities makes the model questionable. Bohte et al. [50] proposed a temporal version of error-backpropagation, SpikeProp. The SpikeProp was demonstrated to be able to classify images with a three-layer spiking neural network. However, the adaptive learning can only be applied to analytically tractable neuron models, and the weights with mixed signs are suspected to cause failures of training [53]. Gütig and Sompolinsky [51] proposed a supervised learning algorithm, temptron, to classify spatiotemporal patterns by generating at least one spike or staying quiescent.

Brader, Senn and Fusi [52] proposed an alternative approach, in which a spike-driven model is able to perform binary image classification with spiking neurons using rate codes. In this approach, grayscale value of each pixel of input images is normalized to a binary value such that the largest element is unity. Then each element was encoded by Poisson spike trains at different frequencies. After learning, images from different classes can be distinguished by the firing rates of output neurons. However, the spike-driven model only focuses on the learning part and pay little attention to the sensory encoding. By transforming 8-bit grayscale images into binary images, a large amount of the images have been discarded. Therefore, the actual information carried by the input patterns are far less than that of the original images. Moreover, the spike-driven learning relies on a stochastic process, which makes the learning algorithm less efficient and computational demanding.

Due to the use of different encoding scheme and learning strategy, the proposed integrated model has several advantages over existing approaches. First, we look at the pattern recognition process at a system level. Rather than considering sensory encoding and learning as isolated processes, we integrated biological plausible encoding and learning processes using consistent neural codes. The latency-phase encoding scheme retains almost all information of the input images with high precision and links up the sensory encoding with learning process. Second, in the integrated spike-based model, we demonstrated that input patterns can be classified by precisely timed spike trains rather than the mean firing rates or single spike code. With the rich capacity of temporal codes, detailed information of the inputs can be exploited by designing the target pattern and precisely timed spikes can be generated. Furthermore, the supervised spike-timing based learning allows an efficient computation and fast convergence, such that the system can be applied to real-life tasks, such as movement control [54] and neuroprostheses control [55].

The input neurons are supposed to fire more than once in our model, which makes better use of the synaptic weights and generalization performance. Although the temporal codes provide a large amount of information, multi-spike signal results in the competition among target spikes firing at different times for the available resources. This leads to limited memory capacity and slow convergence as shown in the simulation results. Therefore the removal of the conflicts among the target spikes remains a challenging but interesting issue for the spike-timing based learning algorithm. One approach is to employ multiple layer and recurrent neural structures, such as liquid state machine [56], so as to increase the computational capability of the system and to absorb the influence of multiple spikes.

There are a few limitations in our current model. The encoding scheme in the model does not incorporate any information extraction to preprocess the input patterns, which is viewed as a necessary procedure in traditional pattern recognition models. By using filtering techniques as proposed in HMAX model [57] or local edge detectors [58], it is believed that the performance and memory capacity in the proposed model will be improved with an efficient neural code in a more concise and abstract manner.

3.5 Conclusions

In this chapter, an integrated computational model with latency-phase encoding method and supervised spiking-timing based learning algorithm has been proposed. Stimuli were first encoded into spatiotemporal patterns with latency-phase scheme, which builds up a bridge between real-world stimuli to neural signals in a biological plausible way. Then the patterns were learned by spiking neurons using a spike-timing based supervised method with millisecond time precision. As shown in the simulation results, the spike-timing based neural networks with temporal codes are capable of solving pattern recognition task by computing with action potentials.

Although the current model has limitations in the recognition capacity, our study exploits the computational mechanisms employed by neural systems in two respects: First, our model was built at a system level emphasizing both the sensory encoding and learning process. It is an integrated system based on a unified temporal coding scheme and consistent with the known neurobiological mechanisms. Second, we have demonstrated the classification capability of the system that computes precisely timed spikes and realistic stimuli, analogously to cognitive computation in human brain. The approaches based on cognitive computation will play a leading role in many applications spanning across signal processing, autonomous systems and robotics [59–61].

References

1. Du Bois-Reymond, E.: Untersuchungen uer thierische elektricita. G. Reimer (1848)
2. Panzeri, S., Brunel, N., Logothetis, N.K., Kayser, C.: Sensory neural codes using multiplexed temporal scales. Trends Neurosci. **33**(3), 111–120 (2010)
3. Softky, W.R.: Simple codes versus efficient codes. Curr. Opin. Neurobiol. **5**(2), 239–247 (1995)
4. Rullen, R.V., Thorpe, S.J.: Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. Neural Comput. **13**(6), 1255–1283 (2001)
5. Adrian, E.: The Basis of Sensation: The Action of the Sense Organs. W. W. Norton, New York (1928)
6. Shadlen, M.N., Newsome, W.T.: Noise, neural codes and cortical organization. Curr. Opin. Neurobiol. **4**(4), 569–579 (1994)
7. Litvak, V., Sompolinsky, H., Segev, I., Abeles, M.: On the transmission of rate code in long feedforward networks with excitatory-inhibitory balance. J. Neurosci. **23**(7), 3006–3015 (2003)
8. Seung, H.S.: Learning in spiking neural networks by reinforcement of stochastic synaptic transmission. Neuron **40**(6), 1063–1073 (2003)
9. Barak, O., Tsodyks, M.: Recognition by variance: learning rules for spatiotemporal patterns. Neural Comput. **18**, 2343–2358 (2006)
10. Bialek, W., Rieke, F., de Ruyter van Steveninck, R., Warland, D.: Reading a neural code. Science **252**(5014), 1854–1857 (1991)
11. Victor, J.D.: How the brain uses time to represent and process visual information. Brain Res. **886**(1–2), 33–46 (2000)
12. Carr, C.E.: Processing of temporal information in the brain. Annu. Rev. Neurosci. **16**(1), 223–243 (1993)
13. Singer, W., Gray, C.M.: Visual feature integration and the temporal correlation hypothesis. Annu. Rev. Neurosci. **18**(1), 555–586 (1995)
14. Kayser, C., Montemurro, M.A., Logothetis, N.K., Panzeri, S.: Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns. Neuron **61**(4), 597–608 (2009)
15. Meister, M., II, M.J.B.: The neural code of the retina. Neuron **22**(3), 435–450 (1999)
16. Gollisch, T., Meister, M.: Rapid neural coding in the retina with relative spike latencies. Science **319**(5866), 1108–1111 (2008)
17. Keat, J., Reinagel, P., Reid, R., Meister, M.: Predicting every spike: A model for the responses of visual neurons. Neuron **30**(3), 803–817 (2001)
18. Llinas, R.R., Grace, A.A., Yarom, Y.: In vitro neurons in mammalian cortical layer 4 exhibit intrinsic oscillatory activity in the 10-to 50-Hz frequency range. Proc. Natl. Acad. Sci. **88**(3), 897–901 (1991)
19. Koepsell, K., Wang, X., Vaingankar, V., Wei, Y., Wang, Q., Rathbun, D.L., Usrey, W.M., Hirsch, J.A., Sommer, F.T.: Retinal oscillations carry visual information to cortex. Front. Syst. Neurosci. **3**, 4 (2009)
20. Heiligenberg, W.: Neural Nets in Electric Fish. MIT Press, Cambridge (1991)
21. Chrobak, J.J., Buzsáki, G.: Gamma oscillations in the entorhinal cortex of the freely behaving rat. J. Neurosci. **18**(1), 388–398 (1998)
22. O'Keefe, J., Burgess, N.: Dual phase and rate coding in hippocampal place cells: theoretical significance and relationship to entorhinal grid cells. Hippocampus **15**(7), 853–866 (2005)
23. Tsodyks, M.V., Skaggs, W.E., Sejnowski, T.J., McNaughton, B.L.: Population dynamics and theta rhythm phase precession of hippocampal place cell firing: a spiking neuron model. Hippocampus **6**(3), 271–280 (1996)
24. Jensen, O.: Information transfer between rhythmically coupled networks: reading the hippocampal phase code. Neural Comput. **13**(12), 2743–2761 (2001)
25. Blumenfeld, B., Preminger, S., Sagi, D., Tsodyks, M.: Dynamics of memory representations in networks with novelty-facilitated synaptic plasticity. Neuron **52**(2), 383–394 (2006)
26. Tang, H., Li, H., Yan, R.: Memory dynamics in attractor networks with saliency weights. Neural Comput. **22**(7), 1899–1926 (2010)

27. Mainen, Z., Sejnowski, T.: Reliability of spike timing in neocortical neurons. Science **268**(5216), 1503–1506 (1995)
28. Bi, G.Q., Poo, M.M.: Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J. Neurosci. **18**(24), 10464–10472 (1998)
29. Bi, G.Q., Poo, M.M.: Synaptic modification by correlated activity: Hebb's postulate revisited. Annu. Rev. Neurosci. **24**, 139–166 (2001)
30. Hopfield, J.J., Brody, C.D.: What is a moment? "cortical" sensory integration over a brief interval. Proc. Natl. Acad. Sci. **97**(25), 13919–13924 (2000)
31. Hopfield, J.J., Brody, C.D.: What is a moment? transient synchrony as a collective mechanism for spatiotemporal integration. Proc. Natl. Acad. Sci. **98**(3), 1282–1287 (2001)
32. Ito, M.: Mechanisms of motor learning in the cerebellum. Brain Res. **886**(1–2), 237–245 (2000)
33. Montgomery, J., Carton, G., Bodznick, D.: Error-driven motor learning in fish. Biol. Bull. **203**(2), 238–239 (2002)
34. Knudsen, E.I.: Supervised learning in the brain. J. Neurosci. **14**(7), 3985–3997 (1994)
35. Ito, M.: Control of mental activities by internal models in the cerebellum. Nat. Rev. Neurosci. **9**(4), 304–313 (2008)
36. Ponulak, F., Kasinski, A.: Supervised learning in spiking neural networks with resume: sequence learning, classification, and spike shifting. Neural Comput. **22**(2), 467–510 (2010)
37. Gerstner, W., Kistler, W.M.: Spiking Neuron Models: Single Neurons, Populations, Plasticity. Cambridge University Press, Cambridge (2002)
38. Nadasdy, Z.: Information encoding and reconstruction from the phase of action potentials. Front. Syst. Neurosci. **3**, 6 (2009)
39. Gawne, T.J., Kjaer, T.W., Richmond, B.J.: Latency: another potential code for feature binding in striate cortex. J. Neurophysiol. **76**(2), 1356–1360 (1996)
40. Reich, D.S., Mechler, F., Victor, J.D.: Temporal coding of contrast in primary visual cortex: when, what, and why. J. Neurophysiol. **85**(3), 1039–1050 (2001)
41. Greschner, M., Thiel, A., Kretzberg, J., Ammermüller, J.: Complex spike-event pattern of transient on-off retinal ganglion cells. J. Neurophysiol. **96**(6), 2845–2856 (2006)
42. Hopfield, J.J.: Pattern recognition computation using action potential timing for stimulus representation. Nature **376**(6535), 33–36 (1995)
43. Arnett, D.: Statistical dependence between neighboring retinal ganglion cells in goldfish. Exp. Brain. Res. **32**(1) (1978)
44. DeVries, S.H.: Correlated firing in rabbit retinal ganglion cells. J. Neurophysiol. **81**(2), 908–920 (1999)
45. Meister, M., Lagnado, L., Baylor, D.A.: Concerted signaling by retinal ganglion cells. Science **270**(5239), 1207–1210 (1995)
46. Widrow, B., Hoff, M.E., et al.: Adaptive switching circuits (1960)
47. Russell, B.C., Torralba, A., Murphy, K.P., Freeman, W.T.: Labelme: a database and web-based tool for image annotation. Int. J. Comput. Vis. **77**(1–3), 157–173 (2008)
48. Schreiber, S., Fellous, J., Whitmer, D., Tiesinga, P., Sejnowski, T.: A new correlation-based measure of spike timing reliability. Neurocomputing **52–54**, 925–931 (2003)
49. Brody, C.D., Hopfield, J.: Simple networks for spike-timing-based computation, with application to olfactory processing. Neuron **37**(5), 843–852 (2003)
50. Bohte, S.M., Bohte, E.M., Poutr, H.L., Kok, J.N.: Unsupervised clustering with spiking neurons by sparse temporal coding and multi-layer RBF networks. IEEE Trans. Neural Netw. **13**, 426–435 (2002)
51. Gütig, R., Sompolinsky, H.: The tempotron: a neuron that learns spike timing-based decisions. Nat. Neurosci. **9**(3), 420–428 (2006)
52. Brader, J.M., Senn, W., Fusi, S.: Learning real-world stimuli in a neural network with spike-driven synaptic dynamics. Neural Comput. **19**(11), 2881–2912 (2007)
53. Haruhiko, T., Masaru, F., Hiroharu, K., Shinji, T., Hidehiko, K., Terumine, H.: Obstacle to training spikeprop networks: cause of surges in training process. In: Proceedings of the 2009 International Joint Conference on Neural Networks, pp. 1225–1229. IEEE Press, Piscataway (2009)

54. Manette, O., Maier, M.: Temporal processing in primate motor control: relation between cortical and EMG activity. IEEE Trans. Neural Netw. **15**(5), 1260–1267 (2004)
55. Müller-Putz, G.R., Scherer, R., Pfurtscheller, G., Neuper, C.: Temporal coding of brain patterns for direct limb control in humans. Front. Neurosci. **4** (2010)
56. Maass, W., Natschläger, T., Markram, H.: Real-time computing without stable states: a new framework for neural computation based on perturbations. Neural Comput. **14**(11), 2531–2560 (2002)
57. Riesenhuber, M., Poggio, T.: Hierarchical models of object recognition in cortex. Nature Nurosci. **2**(11), 1019–1025 (1999)
58. van Wyk, M., Taylor, W.R., Vaney, D.I.: Local edge detectors: a substrate for fine spatial vision at low temporal frequencies in rabbit retina. J. Neurosci. **26**(51), 13250–13263 (2006)
59. Perlovsky, L.: Computational intelligence applications for defense [research frontier]. Comput. Intell. Mag. IEEE **6**(1), 20–29 (2011)
60. Meng, Y., Zhang, Y., Jin, Y.: Autonomous self-reconfiguration of modular robots by evolving a hierarchical mechanochemical model. Comput. Intell. Mag. IEEE **6**(1), 43–54 (2011)
61. Yan, R., Tee, K.P., Chua, Y., Li, H., Tang, H.: Gesture recognition based on localist attractor networks with application to robot control [application notes]. Comput. Intell. Mag. IEEE **7**(1), 64–74 (2012)