Neural Processing Letters (2021) 53:4693–4710
https://doi.org/10.1007/s11063-021-10562-2

A Survey of Encoding Techniques for Signal Processing in Spiking Neural Networks

Daniel Auge[1] · Julian Hille[1,2] · Etienne Mueller[1] · Alois Knoll[1]

Accepted: 9 June 2021 / Published online: 22 July 2021

Abstract
Biologically inspired spiking neural networks are increasingly popular in the field of artificial intelligence due to their ability to solve complex problems while being power efficient. They do so by leveraging the timing of discrete spikes as main information carrier. Though, industrial applications are still lacking, partially because the question of how to encode incoming data into discrete spike events cannot be uniformly answered. In this paper, we summarise the signal encoding schemes presented in the literature and propose a uniform nomenclature to prevent the vague usage of ambiguous definitions. Therefore we survey both, the theoretical foundations as well as applications of the encoding schemes. This work provides a foundation in spiking signal encoding and gives an overview over different application-oriented implementations which utilise the schemes.

Keywords Spiking neural networks · Neural coding · Neuromorphic computing · Rate coding · Temporal coding

1 Introduction

Spiking Neural Networks (SNNs) use short “all-or-nothing” pulses to encode and transmit information. Such networks consist of neurons which describe the action potential generation as mathematical non-differential equation to approximate the observed behaviour of biological systems. This third generation of neural networks [54] promises a better processing performance than classical networks based on activation functions [53]. Additionally, since spikes are only exchanged when information is processed, the energy consumption of SNNs can be a fraction of comparable networks of the earlier generations on specalised hardware [8]. To make use of these principles, still many questions regarding data encoding, network

✉ Daniel Auge
daniel.auge@tum.de

Alois Knoll
knoll@in.tum.de

[1] Institut für Informatik VI, Technische Universität München, Boltzmannstr. 3, 85748 Garching bei München, Germany

[2] Infineon Technologies AG, Am Campeon 1-15, 85579 Neubiberg, Germany

architectures, training, and hardware realisations have to be answered. In this study, we focus on the question of how to convert analog and digital data into spikes.

The human brain as the most complex and efficient spike processing computing device might give some insights into the biological answer to this question. It uses various encoding schemes to represent visual, acoustic, or somatic data from the different senses. Signals containing motor commands executed by muscles again make use of further encodings. This biological model suggests, that coding schemes exist which are better suited for particular data forms than others, and that there are multiple schemes to choose from.

The implementations of artificial SNNs have shown a variety of different encoding schemes. Comparable with the biological findings, two main coding approaches can be differentiated: rate coding and temporal coding [29]. Rate codes embed the information in the instantaneous or averaged rate of spike generation of a single or group of neurons. This leads to a value which describes the activity of a neuron, which is comparable to the activation value of ordinary non-spiking artificial neurons. For temporal coding techniques, the precise timing of and between spikes is used to encode information. This includes the absolute timing in relation to a global reference, the relative timing of spikes emitted by different neurons or simply the order in which a population of neurons generate specific spikes.

As in the biological case, specific coding techniques are better suited than others depending on the type of data to be dealt with and the structure of the network. Networks analysing frame-based two-dimensional images will need a different approach than architectures dealing with audio streams. Systems, which rely on high processing speeds and fast responses on stimulus onset will not use codes which are based on temporal averaging. And neurons driving actuators will have to use different coding schemes than sensors retrieving a representation of the environment.

In addition to the wide variety of coding techniques available, often the nomenclature and categorisation used in the literature is not uniform amongst the publications. Some reports use the same designation for fundamentally different schemes. Others categorise existing techniques in many different ways. So far, there has been no publication which summarises and standardises the different approaches of the research community.

In this work, we will give an overview of existing coding techniques and establish a uniform nomenclature and categorisation. We focus on the abstract description of the different coding schemes to ensure generality. We thus do not closely consider distinct biological realisations consisting of specialised input neurons. However, we include biological examples to motivate the biological relations. Subsequently, we give an overview of exemplary implementations and applications of these schemes given in the literature. The network architectures used in those applications range from reservoirs over deep feedforward structures to convolutional implementations showing the large variety of available layer types and encoding scheme combinations. In the discussion we showcase differences and trade-offs between the encoding techniques.

2 Biological Background

Early research of biological neural systems suggested that rate coding is the predominant technique to transmit information within nervous systems [2]. Later publications, in contrast, showed that all sensory organs rather embed their perceptions into precise timings of action potentials. While Thorpe proposed the idea of exact spike timings as coding scheme [97], it took several years to find experimental evidence for this theory. It has been shown that

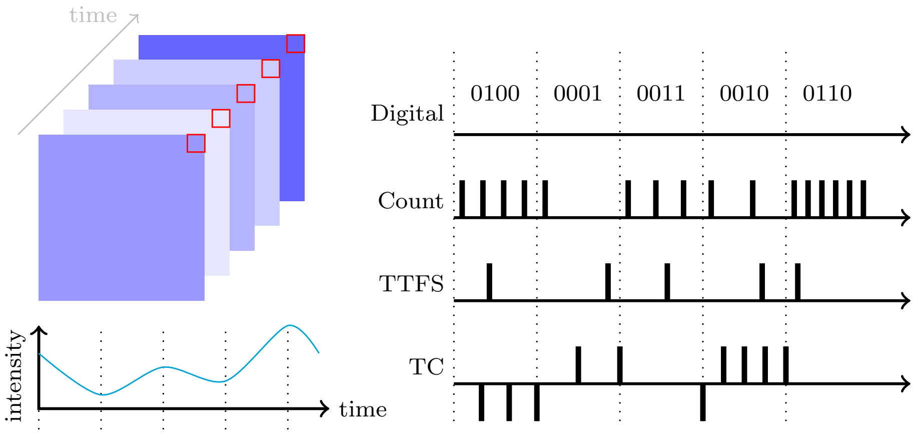


Fig. 1 Exemplary coding schemes for a sequence of images over time. The intensity-time plot indicates the changes of the pixel value in the red square as a continuous function. The dashed lines indicate the time instances at which the images have reached the colour value. Digital, count, and TTFS spikes in correlation to the local minima and maxima in the intensity curve. The TC emits spikes if the continuous intensity change exceeds a certain threshold

the human visual system needs 150 ms to process object recognition tasks, which supports this suggestion since rate codes would be too slow to explain these fast responses [95]. Further research supports these findings for visual [27,30,57,72], audio [25], tactile [39], and olfactory systems [1,55]. Additionally, the experiments show the trade-off between fast responses after stimulus onset and accuracy of the result. Mice, for example, can discriminate between simple odors within 200 ms. If they are similar, the discrimination can take 100 ms longer, suggesting an integration over time [39].

3 Encoding

To illustrate the applicability of different encoding schemes on the same problem, consider a video camera, which encodes visual information into spikes as depicted in Fig. 1. Digital cameras expose the image sensor in a fixed rate for a short period of time and encode the measured intensity of light at each pixel into an integer number. This frame-based approach can also be applied to generate a spike representation of the video stream. The light intensity can directly be translated into spike times, where a highly exposed pixel corresponds to a fast spike time or vice versa (time-to-first-spike (TTFS)). Alternatively, the intensity can be converted into the number of spikes generated within one frame (count). Here, a large number of emitted spikes can correspond to a high intensity at the associated pixel. As a third example, we could discard the approach of using frame-based measurements (temporal contrast (TC)). Instead, we observe the light intensity at a certain pixel and emit a spike as soon as the intensity change surpasses a distinct threshold.

In general, all encoding techniques can be divided into two main categories: rate and temporal coding. All specialised encoding schemes can be separated into these two by answering the question whether the exact timing and order of spikes is crucial for the information to be submitted. The resulting taxonomy is depicted in Fig. 2. Population codes, which are often referred to as third main category only add the information whether a single or multiple neurons are used in the particular coding scheme. Though, this can be the case in both temporal and rate codes and does not provide a unique differentiation.

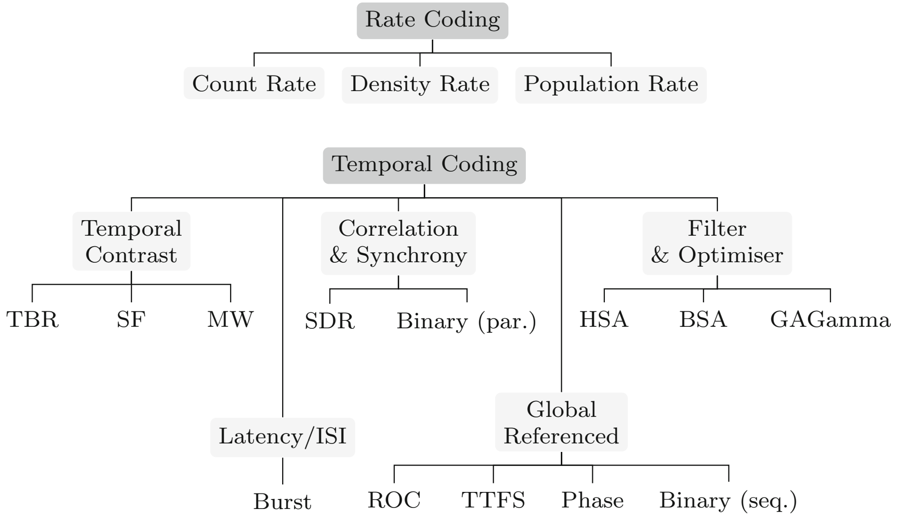
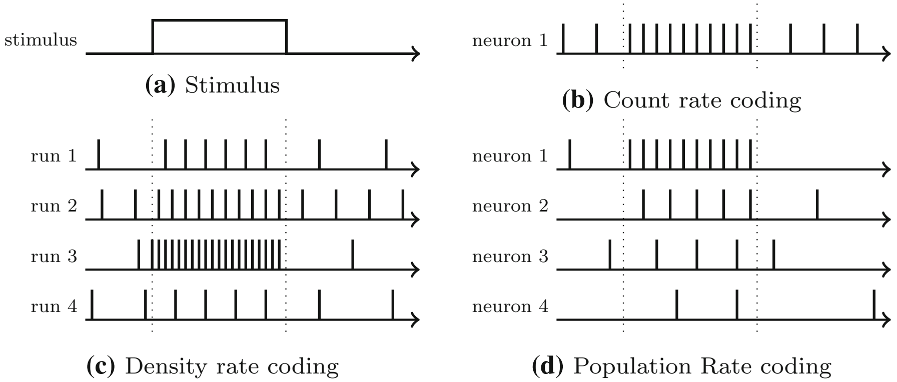


Fig. 2 Taxonomy of rate and temporal coding techniques



(a) Stimulus

(b) Count rate coding

(c) Density rate coding

(d) Population Rate coding

Fig. 3 Visualisation of rate coding techniques with a wide pulse stimulus. The dashed line indicates the rising and falling edge of the stimulus

3.1 Rate Coding

Rate codes can be further divided into three subcategories: count, density and population rate codes. Figure 3 shows an example visualisation of rate coding of an arbitrary input stimulus. For information exceeding the next sections, we refer to [28,29].

Count rate (*average over time*) is the most common rate coding scheme. It is defined by the mean firing rate

$$v = \frac{N_{\text{spike}}}{T}, \tag{1}$$

with the spike count N_{spike} and the time window T. This scheme is also referred to as frequency coding. In vivo, Adrian and Zotterman observed that stretching a frog muscle with different weights affects the frequency of the firing rate [2]. In artificial applications, firing rates can describe any slowly varying analog value, from pixel intensities to gas concentrations.

In a count rate code, the spike times can either be exact or random. The latter case is often modelled by a Poisson distribution. When encoding an analog number, the remaining reconstruction error due to the discretised number of spikes during a given time interval decreases by the number of spikes $1/N_{\text{spikes}}$. Due to the variations given in Poisson distributed spike trains, the error decreases only by $1/\sqrt{N_{\text{spikes}}}$ [18].

Density rate (*average over several runs*) The neural activity is measured over different simulations and the results of the neural responses are presented in a peri-stimulus-time histogram to visualise the spike activity. The spike density is defined by

$$p(t) = \frac{1}{\Delta t} \frac{N_{\text{spike}}(t; t + \Delta t)}{K}. \tag{2}$$

The number of spikes N_{spikes} in a time interval $[t; t + \Delta t]$ averaged over all iterations divided by the total number of iterations K and the duration Δt, specifies the spike density $p(t)$. This scheme is not a biologically plausible encoding method. One imagines a frog which tries to catch a fly by averaging over multiple computations over the exact same trajectory of the fly [29]. In an artificial SNN however, it can be beneficial to average over multiple simulation runs with the exact same inputs.

Population rate (*average over several neurons*) is based on similar properties of neurons in a population. The firing rate is defined by

$$A(t) = \frac{1}{\Delta t} \frac{N_{\text{spike}}(t; t + \Delta t)}{N}. \tag{3}$$

The number of spikes N_{spikes} in the total population are summed together for the time interval $[t; t + \Delta t]$ and divided by the duration Δt and the total number of neurons N.

A population of neurons does not necessarily have to be uniform in the spike response of neurons for a given input. If each neuron has a different (known) tuning curve describing the spike count rate at any input current, the superposition in a large population can encode single numbers, vectors or even function fields [21].

3.2 Temporal Coding

As depicted in Fig. 2, temporal codes can be divided into multiple subcategories. While temporal contrast (TC) schemes focus on the signal's derivative, globally referenced encodings process the input in packets in reference to a periodical signal or oscillation. Inter-spike-interval (ISI) codes interpret the relative timing between grouped blocks of spikes in contrast to correlation codes, which rely on the simultaneous activity of several neurons. Filter and optimiser based approaches base their spike patterns on the comparison of input and kernel functions. Figure 4 demonstrates the temporal encoding schemes in relation to a stimulus. Note that binary codes, Ben's spiker algorithm (BSA), and TC use a different stimulus in the illustration.

Global Referenced

The most basic temporal coding scheme is **TTFS**, which encodes information by the time difference Δt between stimulus onset and the first spike of a neuron. In the simplest case, the firing time can be the inverse of the stimulus amplitude $\Delta t = 1/a$ or a linear relation $\Delta t = 1 - a$, with a being the normalised signal amplitude. In both cases, a large amplitude leads to an early firing time whereas low amplitudes lead to a large interval or no spike at

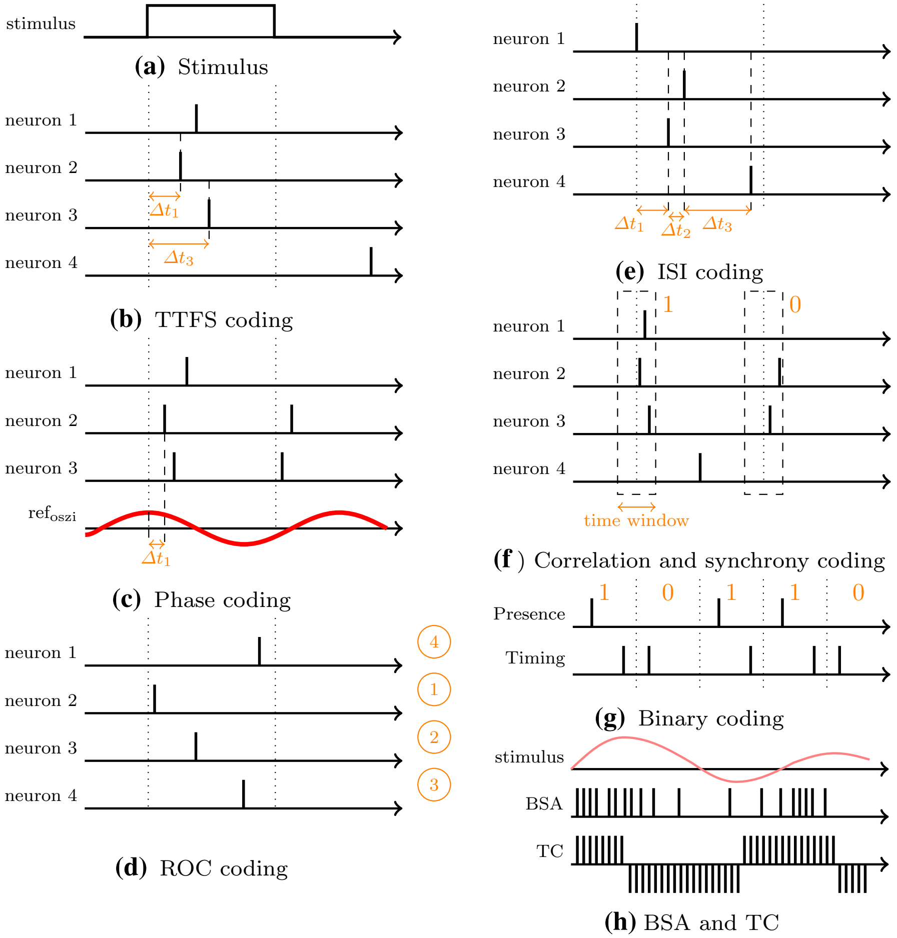


Fig. 4 Visualisation of temporal coding techniques with dashed line indicating the rising and falling edge of the stimulus. Δt describes the latency between the reference point and the spike. In (d) the order of spikes is numbered on the right

all. As a biological example, Johansson and Birznieks discovered that the relative time of the first spike in regard to a discrete mechanical fingertip event contains direction and force information [39]. Gollisch and Meister observed TTFS in the retinal pathway and found invariant relation to stimulus contrast and robustness to noise variations [30]. Though, they called the coding scheme “latency coding”, which can be mistaken as ISI coding due to the unclear definition of latency between spikes and a global reference or between multiple spikes.

Instead of a single reference point, **phase** coding encodes information in the relative time difference between spikes and a reference oscillation [36,42]. The phase pattern repeats periodically if no changes between the cycles appeared. Each single neuron fires in respect to the reference signal and encodes the data similar to TTFS. Such a behaviour was detected by Gray, König, Engel, and Singer [31]. They analysed the firing probability of neurons in the cat visual cortex and identified a relation between the firing pattern and a reference oscillation.

ROC (rank-order coding) is based on the firing order of a population of neurons in relation to a global reference [26,96]. In contrast to TTFS, ROC encodes the information without considering the precise timing of the spikes. It functions as a discrete normalisation filter with the loss of the absolute amplitude information. As a consequence, it is not possible to reconstruct the absolute signal amplitude or an exactly constant signal. The scheme is further limited by the distinction of the spikes and jitter due to a huge effect for small ISI. In the basic version of ROC the precise spike time is not relevant but there are modified versions which use the ISI to encode additional information.

A further subcategory of globally referenced schemes are **(sequential) binary** codes. Here, each spike corresponds to a ”1” or ”0” in a bit stream. In relation to a fixed reference clock, two schemes to encode the bits are possible: the presence or absence of a spike within a given interval [110], or the timing of the spike within the interval [33]. In the former case, a logical ”1” corresponds to a spike being present during one clock cycle. In the latter case, the clock cycle is divided into two sub-intervals. If a spike is present in the first half, a ”0” is encoded, if it is the case in the second half, a ”1” is present or vice versa. This ensures the constant presence of spikes independent of the bit pattern to be encoded.

Often, the first spike of all global referenced coding schemes represents the most significant element of the pattern, comparable to binary representations. This leads to an interesting behaviour in the network parameter selection because the threshold of the output neurons can be adjusted in regard to the speed-accuracy trade-off. This means the network can already predict the output pattern before the whole input pattern has been processed [96].

ISI Coding

In **ISI** or **latency** coding the information is embedded into the relative time difference (latency) between the spikes of a neuron group [71]. The dependency of the ISI with the stimulus intensity was observed in pyramidal cells [64]. Li and Tsien [48] state that rare events such as longer silence periods contain more information than periods of higher spike activity.

A sub-category of ISI coding is **burst** coding which converts the input into various inter-spike latencies. A burst is a group of spikes with a very small ISI [67]. If a spike is a part of the burst depends on the ISI threshold and the expected number of spikes [99,108].

Correlation and Synchrony

Correlation and synchrony coding uses the temporal reference to other spiking neurons. The input pattern is converted into a spatio-temporal spike representation. There, spike groups with a relative short ISI represent specific input patterns [29]. Information is encoded by the distinction of which neurons fire at the same time. **Sparse distributed representations (SDRs)** [4,62] also fall into this category. Here, a subset of neurons inside a population is active at any given point of time. This enables to represent a virtually infinite number of patterns without significant errors [34]. In the extreme case, only one single neuron is active at any given time. Then, every neuron is allocated to a specific input value. A spike is generated as soon as this value is crossed. This scheme can be referred to as **amplitude** coding, since the signal strength is directly encoded in the activity of one neuron at a time.

In vivo, the general synchronous coding scheme has been observed in the somatosensory cortex of monkeys [88] or the visual cortex of cats [31,32]. There, the authors hypothesise that synchrony can give evidence about the significance of the incoming stimulus. A further biological example are grid and place cells [58,61] which encode spatial representations into the synchronous firing of a specific subset of a population.

Next to the already introduced sequential binary codes, the synchronous firing of neurons can also be interpreted as the ones and zeros of a binary number. In these **(parallel) binary** codes, each neuron encodes a specific bit within a larger word in contrast to the sequential codes, where a single neuron encodes the information into the precise timing within a stream.

Filter and Optimizer-based Approaches

In both neuroscience and control theory, an often utilised method to find a description of a system is to feed a known signal into it and to measure its output. In the neurological case, the input is an arbitrary analog signal, the system is a single neuron or population, and the output is a spike train. **BSA** [81] and its predecessor **Hough spiker algorithm (HSA)** [37] reverse this idea and use a known filter to compute a spike train for a corresponding input signal. A spike is generated as soon as the convolution of signal and filter exceeds a certain threshold. Since this method can only process inputs of a specific range, the incoming signal has to be normalised prior to conversion.

Sengupta, Scott, and Kasabov interpret the encoding process as a data compression problem with background knowledge [83] and introduce the GaGamma scheme. Thereby, information has to be maximised while minimising the spike density. By leveraging prior knowledge of the signal to be encoded, specific optimal solutions can be found while solving the mixed-integer optimisation problem.

Temporal Contrast

The last subcategory of temporal codes is TC coding. It converts an analog signal to a spike train by observing the changes in the signal intensity [40]. It is separated into three different algorithms: **threshold-based representation (TBR)**, **step-forward (SF)**, and **moving-window (MW)**. TBR compares the absolute signal change of an input signal with a threshold and emits positive or negative spikes accordingly. The threshold depends on the summation of the mean derivative with the multiplication of a factor and the derivative standard deviation. In contrast to TBR, SF just uses the next available signal value and checks if the previous value and an additional threshold is exceeded. It sends out appropriate spikes depending on the polarity of the signal difference. MW uses a base which is defined by the mean of the previous signal in a time window. Again positive or negative spikes are emitted if the current signal value exceeds the base and threshold. For further information and implementations we refer to [70].

4 Applications

As shown in the introductory example, one single type of input data for a given problem can be translated into spikes in several ways. The following implementations demonstrate this further and should give an overview of the variety of problems which can be solved with SNNs. Additionally, it further indicates that a universal answer to neural coding has not been found yet.

4.1 Rate Coding

Early work on SNNs is mainly based on rate coding. Until 2012 multiple authors presented fully connected feedforward networks which achieved up to 94% for digit recognition on the MNIST handwritten digits dataset [11,22,47,91]. Unsupervised spike-timing-dependent plasticity (STDP)-based models improved the accuracy to 95% in 2015 [19] and over 97% in 2019 [92]. A similar approach could classify the iris dataset with an accuracy of 97% and the Wisconsin breast cancer dataset with 94% [79].

Interestingly, the best results were achieved by training non-spiking artificial neural networks (ANNs) and subsequently converting them into the spiking domain. Whereas using sigmoid as an activation function turned out to be suboptimal for translating it into spiking neurons [73]. Today's default activation function ReLU can almost directly be translated into the spiking rate, by only normalizing the weight for a near-lossless accuracy conversion [20]. Through this approach the accuracy on MNIST could be leveraged to over 98% [20,38,59] and achieving the best performance of 99.42% by Esser, Appuswamy, Merolla, Arthur, and Modha [23].

Similarly for convolutional neural networks (CNNs), whereas earlier work relied on STDP achieving over 98% [43,92,93], conversion-based approaches could easily attain more than 99% [20,76]. A significantly larger discrepancy can be found by training on the more challenging CIFAR-10 dataset [46], which could only achieve 75.42% [65] without, but 90.85% with conversion methods [76].

Some research also focuses on how to obtain those rate-coded signals. Besides the highly popular applications in image processing, Liu, Schaik, Mincti, and Delbruck proposed an event-based cochlea, which encodes the amplitude of specific frequencies within a signal into a rate code [51]. These “pulse-frequency modulators” emit a higher event rate the larger the corresponding frequency component is.

Besides classification tasks, rate-coded information are often used in robotic applications [7]. Most implementations make use of Poisson-distributed spike trains to closely emulate the properties of real neurons.

To overcome the limitation of rate-based networks of producing large amounts of spikes, Zambrano and Bohte [107] presented a method for adapting the firing rate, resulting in a significant reduction of spike events. A different approach uses a global referenced binary coding to reduce the number of spikes. Together with neuron models with exponential input characteristics, the same activation of the neuron can be reached as with a count rate code but with far less spikes [110].

4.2 Temporal Coding

Global Referenced

The idea of converting ANNs was historically based on rate coding schemes, but there are also temporal based methods. Rueckauer and Liu used the coding scheme TTFS for the classification of the MNIST dataset with less operations and an error rate within 2%. The SNN implementation decreases the computational cost by factor 7 for the LeNet5 architecture on MNIST [75]. Zhang, Zhou, Zhi, Du, and Chen [109] also utilise TTFS encoding on a converted network; but in contrast, they apply the scheme reversed. Here, the first spike encodes the weakest feature whereas the last spike has the largest influence. A further approach uses phase coding to represent information inside a converted ANN [44]. The authors show that this reduces the overall number of spikes and the inference latency while preserving the accuracy of the image recognition tasks.

The artificial microelectronic nose by Chen, Ng, Bermak, Law, and Martinez uses ROC and TTFS to detect gases like ethanol, carbon monoxide and hydrogen [13]. The sensor output is sampled and converted to a spike train in a microcontroller. The data are then inserted to an SNN which identifies the gas type. Encoding the samples with rank order coding achieved an classification accuracy of 95.2% and with TTFS 100%. This difference arises from the fact that in ROC the spikes are really close together and a small spike jitter has a large effect on the classification accuracy.

In [9], the authors implemented an unsupervised network which can compute and learn clusters from realistic high-dimensional data. They used a sparse temporal coding scheme which they called population coding. We would define this coding as sparse TTFS coding, because the relative time difference between the spikes and the stimuli contains the crucial information. The input neurons cover the whole data-range and use Gaussian receptive fields to map the continuous input values to specific delay times. Significant data will have small delays and non-relevant data will not emit an action potential in the defined time interval which introduces sparsity. A similar coding was implemented with a deep SNN for image classification with data-sets like Caltech 101, ETH-80, and MNIST [43]. The first network layer detects the contrasts in the input image with a Gaussian filter and encodes the contrast into spike latencies. Higher contrast has shorter delay and too low contrast will be neglected. This convolutional SNN achieved an accuracy of 98.4% in the MNIST data-set. Similar encoding idea was implemented on the iris data-set with an accuracy of 92.55% in [105,106]. It was extended by observing two different input connection schemes. First by connecting each receptive field row with a neuron and the second with sparse random connections between the receptive fields and the input neurons. During learning the random connection achieves faster and higher accuracy rates compared to the structured connections. [79] also implemented the temporal coding based on the Gaussian receptive field and accomplished an accuracy of 99% for the iris data-set and 90% for the Wisconsin breast cancer data-set with a single layer (comparable to the state-of-the-art). During the learning process the network tries to memorise patterns for future feature predictions.

Delorme and Thorpe propose a network for image recognition which operates entirely in the spiking domain [16,17]. The input layer of the network consists of pairs of ON and OFF center cells which indicate the intensity difference across the cells. Based on this activity, the spike code is generated. This process resembles the operation of biological eyes. The second and third layer of the network consist of neurons selective on edges of different orientations and the final class label, respectively. A comparable approach has later been described by Wysoski, Benuskova, and Kasabov [104] where face recognition of stream data is performed by accumulating different opinions over several views.

A further interesting approach is presented by Liu and Yue. The authors combine the feature extraction capabilities of classical neural networks with the fast unsupervised learning of spiking neural networks [50]. In their proposed network, features are extracted from image data using a simplified convolutional hierarchical max-pooling model [85], and encoded into spikes using the ROC scheme. Subsequently, the spikes are fed into the second (spiking) half of the network which utilises the unsupervised STDP learning method to identify different classes.

The network implementations dealing with audio input for speaker identification or speech recognition utilise the frequency domain representation of the incoming audio signal [52,102, 103]. The transform between time and frequency domain is realised using general filter banks or mel-cepstral coefficients. The resulting feature vectors represent the frequencies present during a fixed measurement time encoded with ROC. In each frame, the amplitudes of each frequency are then encoded into spike latencies. In most cases, two succeeding measurement frames have an overlap of 50 %.

ISI Coding

Implementations of pure ISI coding schemes are not widely used. Sharma and Srinivasan implemented a time series forecasting network by encoding the data into the latency between consecutive spikes [87]. The network achieved higher accuracy than traditional networks with a smaller architecture size, leveraging ISI coding and an evolutionary learning algorithm.

The subcategory of burst coding indicates to be a fast and energy-efficient information coding technique. This was shown on the MNIST and CIFAR classification problems with a deep SNN architecture [67]. Furthermore, Chen and Qiu implemented burst coding for real-time anomaly detection on the IBM TrueNorth processor [14]. The input consists of a continuous stream from the intrusion detection DARPA dataset. They observed that burst coding increases the detection accuracy while decreasing the hardware complexity compared to rate coding.

Correlation and Synchrony

Sparse representations as one subcategory of synchrony coding are implemented in the hierarchical temporal memory (HTM) model [35]. The goal of this model is to understand and mimic the human neocortex and utilise it in several scientific and industrial applications. The implementation of HTM by Numenta is a clocked system consisting of a spatial pooler learning sparse representations of input neurons which fire together, and a temporal memory where temporal pattern sequences are determined. The system is well suited for applications dealing with anomaly detection or prediction of recurring sequences. The developers show this at examples from different domains like GPS surveillance or monitoring the CPU utilisation in computer centres [3].

An application of amplitude coding is given in [5]. There, the authors encode images by sequentially iterating over all pixels and converting each pixel's grey-value to a spike event of the neuron associated with the same intensity threshold.

Filter and Optimizer-based Approaches

Filter and optimizer-based approaches are primarily used to encode data streams. Examples are the utilisation of BSA for electroencephalography (EEG) classification [60] or speech recognition [80]. Additionally, BSA is implemented in the NeuCube simulator as one of the proposed encoding schemes [40]. GAGamma encodes functional magnetic resonance imaging (fMRI) data using an optimizer-based approach by leveraging the prior knowledge of the signal properties [82].

Temporal Contrast Coding

A prominent example of temporal contrast coding in hardware applications are event-based cameras. Lichtsteiner, Posch, and Delbruck implemented the first asynchronous event based camera which can detect changes in light intensity with a high dynamic range [15,49]. For each pixel of the camera sensor, a positive or negative spike event is emitted as soon as the

Table 1 Overview and comparison between different MNIST classification implementations with different coding schemes which are capable to handle frame based inputs

Coding group	Coding scheme	Technique	Accuracy	References
Rate	Count rate	Conversion based	99.42%	[23]
Temporal	TTFS	Conversion based	98.06%	[75]
Temporal	Phase	Spiking CNN	99.2%	[44]
Temporal	Burst	Spiking CNN	99.25%	[67]
Temporal	ROC	Spiking CNN	98.4%	[43]
Temporal	SDR	HTM	95%	[98]

relative change surpasses a threshold. Because the relative change is evaluated per pixel even scenes with uneven lighting conditions can be perceived with high detail. These biologically inspired cameras send out data packets containing the coordinates of the respective pixel and the time stamp of the event. Accordingly, in contrast to classical image-based cameras, only pixels which are subject to intensity changes transmit information. These type of optical sensors are often used in robotics [6,56] or classification tasks. Datasets for classification applications containing event camera-based recordings of MNIST, Caltech101, poker cards, or human postures are readily available [12,63,73,84].

A CNN-based evaluation of the different datasets is given in [90]. Paulun, Wendt, and Kasabov present the processing of spike trains generated by event cameras using the NeuCube simulator [68]. The simulator additionally implements the temporal contrast schemes for other types of input data. Kasabov, Scott, Tu, et al., for example, used TBR to encode real valued weather data to predict the population of a species in relation to weather and climate factors and achieved a state-of-the-art accuracy [40]. Many further applications and methodical background information in close relation to the NeuCube simluator can be found in [41].

5 Discussion

After presenting the concepts and applications, the remaining question is which encoding scheme to use for a specific application. Many publications discuss this question and compare different sub-sets of the presented coding schemes [45,69,77,79,86,89,94,100]. Most of them report a comparison of rate and temporal codes. In general, the coding schemes differ in accuracy, dynamics, latency, noise vulnerability, energy consumption, hardware requirements, and many more.

One approach to quantify the differences of coding schemes is by applying information theory on the topic of neural coding. Here, it has been tried to compare coding schemes with respect to the number of bits which can be encoded by a specific number of neurons or spikes [10,66,101]. Count rate codes for example encode $log_2(N_{\text{spikes}} + 1)$ bit of information into N_{spikes} spikes [77]. ROC-coded signals encode $log_2(N_{\text{spikes}}!)$ bit [96] since the order of the respective spikes carries the information. Reducing the coding schemes to a single number of bits enables a quantitative comparison but lacks the consideration of many other aspects which influence the efficiency of a code. Foremost, the developed processing architecture must match the chosen coding scheme. Even though having a highly efficient coding scheme which can encode data with a low number of spikes accurately does not necessarily lead to an efficient system. Hence, we must rely on qualitative analyses of the coding schemes or comparisons of whole systems. Some publications provide these quantitative system comparisons [13,79] by evaluating classification accuracy or energy consumption at specific tasks. Table 1 provides an overview of MNIST classification accuracies for different coding techniques. Though the differences are not only linked to the encoding scheme since the publications describe various learning methods and network architectures. Consequently, the accuracies provide information on the general system performance but not on the coding schemes themselves. In the next few paragraphs we try to summarise some of the important qualitative differences between the schemes.

While rate coding was seen as the only meaningful code in populations [86], current research focuses more on those coding schemes which are based on precise spike times. Though, rate codes are also utilised in different applications. Count rate codes are often used in applications which convert ANNs into SNNs due to their equivalence to activation values. Researchers show a lossless conversion while reducing the power consumption of the network by decades, given optimised neuromorphic hardware is used [76]. Another strength of rate codes is their robustness and their behaviour towards noise [21].

Temporal codes, however, have been shown to offer a higher information capacity compared to rate codes [78], faster reaction times, and higher transmission speeds. Furthermore, they favour the utilisation of local learning rules like STDP. Rullen and Thorpe state that ROC is biologically more realistic than TTFS due to the fact that the brain cannot know the exact start of a stimulus [77]. The same argument is used by Rolls, Franco, Aggelopoulos, and Jerez against both ROC and TTFS [74]. The authors analysed the information content of spikes in the inferior temporal visual cortex and propose that count rate is fast in short time windows and transports more information than TTFS or ROC from a biological perspective due to the effect of spontaneous neuronal firing. Li and Tsien argue that this spontaneous spike activity is related to the ISI which carries more information than expected and should not be ignored [48]. This shows that there are still different opinions on the encoding techniques.

Rate and Temporal coding provide different benefits and combining these schemes could have a huge impact in the system performance. The fast temporal coding can be used for fast systems and the rate coding for methods with less strict time constraints [39,78]. Fairhall, Lewen, Bialek, and de Ruyter van Steveninck suggest a multi-layer coding scheme where spike trains represent information in different channels of various encoding schemes depending on the timescale [24]. Similar ideas are called hybrid coding where the neural encoding scheme varies between network layers [67] or the neuron switches between coding techniques [77]. The topic of hybrid neural code is not yet clearly defined and needs further investigations.

6 Conclusion

In biological systems there exist several techniques to encode sensory information into spike trains. Probably many more yet to explore. In this work, we summarised those schemes together with less biologically plausible encoding schemes for the utilisation in applications based on artificial SNNs. In summary, there are two main categories of encoding schemes. Rate-based schemes average the spike activity over time, populations, or several runs and do not rely on the precise timing of every single spike event. They convince through their robustness against fluctuations and noise as well as their simplicity due to the equivalence to the activation value of current ANNs. Temporal encoding schemes on the other hand rely on the precise timing of every single spike and can thus achieve higher information densities and efficiencies. However they involve more complex architectures and lacking training methods.

It is expected that more applications for SNNs will arise with the perspective of more advanced architectures, better learning algorithms and the development of energy-efficient neuromorphic hardware. To assist this growing field, further investigations on neural coding techniques in system contexts need to be made.

Funding Open Access funding enabled and organized by Projekt DEAL.

References

1. Abraham NM, Spors H, Carleton A, Margrie TW, Kuner T, Schaefer AT (2004) Maintaining accuracy at the expense of speed: stimulus similarity defines odor discrimination time in mice. Neuron 44(5):865–876. https://doi.org/10.1016/j.neuron.2004.11.017
2. Adrian ED, Zotterman Y (1926) The impulses produced by sensory nerve endings: part 3 impulses set up by touch and pressure. J Physiol 61(4):465–483
3. Ahmad S, Lavin A, Purdy S, Agha Z (2017) Unsupervised real-time anomaly detection for streaming data. Neurocomputing 262:134–147. https://doi.org/10.1016/j.neucom.2017.04.070
4. Ahmad S, Scheinkman L (2019) How Can We Be So Dense? The benefits of using highly sparse representations. arXiv preprint arXiv:1903.11257
5. Bellec G, Salaj D, Subramoney A, Legenstein R, Maass W (2018) Long short-term memory and learning-to-learn in networks of spiking neurons. In: Advances in neural information processing systems, pp. 787–797
6. Bing Z, Meschede C, Huang K, Chen G, Rohrbein F, Akl M, Knoll A (2018) End to end learning of spiking neural network based on r-stdp for a lane keeping vehicle. In: 2018 IEEE international conference on robotics and automation (ICRA), pp. 4725–4732. IEEE
7. Bing Z, Meschede C, Röhrbein F, Huang K, Knoll AC (2018) A survey of robotics control based on learning-inspired spiking neural networks. Front Neurorobot 12:35 Publisher: Frontiers
8. Blouw P, Choo X, Hunsberger E, Eliasmith C (2019) Benchmarking keyword spotting efficiency on neuromorphic hardware. In: Proceedings of the 7th annual neuro-inspired computational elements workshop, pp. 1–8
9. Bohte SM, La Poutré H, Kok JN (2002) Unsupervised clustering with spiking neurons by sparse temporal coding and multilayer RBF networks. IEEE Trans Neural Netw 13(2):426–435
10. Borst A, Theunissen FE (1999) Information theory and neural coding. Nat Neurosci 2(11):947–957. https://doi.org/10.1038/14731
11. Brader JM, Senn W, Fusi S (2007) Learning real-world stimuli in a neural network with spike-driven synaptic dynamics. Neural Comput 19(11):2881–2912
12. Calabrese E, Taverni G, Awai Easthope C, Skriabine S, Corradi F, Longinotti L, Eng K, Delbruck T (2019) Dhp19: Dynamic vision sensor 3d human pose dataset. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops
13. Chen HT, Ng KT, Bermak A, Law MK, Martinez D (2011) Spike latency coding in biologically inspired microelectronic nose. IEEE Trans Biomed Circuit Syst 5(2):160–168
14. Chen, Q, Qiu Q (2017) Real-time anomaly detection for streaming data using burst code on a neurosynaptic processor. In: Design, automation & test in europe conference & exhibition (DATE), 2017, pp. 205–207. IEEE https://doi.org/10.23919/DATE.2017.7926983. Event-place: Lausanne, Switzerland
15. Delbruck T, Lichtsteiner P (2007) Fast sensory motor control based on event-based hybrid neuromorphic-procedural system. In: 2007 IEEE international symposium on circuits and systems, pp. 845–848. IEEE . https://doi.org/10.1109/ISCAS.2007.378038. Event-place: New Orleans, LA, USA

16. Delorme A, Perrinet L, Thorpe SJ (2001) Networks of integrate-and-fire neurons using rank order coding B: Spike timing dependent plasticity and emergence of orientation selectivity. Neurocomputing 38—-40:539–545. https://doi.org/10.1016/S0925-2312(01)00403-9
17. Delorme A, Thorpe SJ (2001) Face identification using one spike per neuron: resistance to image degradations. Neural Netw 14(6–7):795–803
18. Denéve S, Machens CK (2016) Efficient codes and balanced networks. Nature Neurosci 19(3):375
19. Diehl PU, Cook M (2015) Unsupervised learning of digit recognition using spike-timing-dependent plasticity. Frontiers Comput Neurosci 9:99
20. Diehl PU, Neil D, Binas J, Cook M, Liu SC, Pfeiffer M (2015) Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. In: Neural networks (IJCNN), 2015 international joint conference on, pp. 1–8. IEEE
21. Eliasmith C, Anderson CH (2004) Neural engineering: computation, representation, and dynamics in neurobiological systems. MIT Press, Cambridge
22. Eliasmith C, Stewart TC, Choo X, Bekolay T, DeWolf T, Tang Y, Rasmussen D (2012) A large-scale model of the functioning brain. Science 338(6111):1202–1205. https://doi.org/10.1126/science.1225266
23. Esser SK, Appuswamy R, Merolla P, Arthur JV, Modha DS (2015) Backpropagation for energy-efficient neuromorphic computing. In: Advances in neural information processing systems, pp. 1117–1125
24. Fairhall AL, Lewen GD, Bialek W, de Ruyter van Steveninck RR (2001) Efficiency and ambiguity in an adaptive neural code. Nature 412(6849):787–792. https://doi.org/10.1038/35090500
25. Galambos R, Davis H (1943) The response of single auditory-nerve fibers to acoustic stimulation. J Neurophysiol 6(1):39–57
26. Gautrais J, Thorpe S (1998) Rate coding versus temporal order coding: a theoretical approach. Biosystems 48(1–3):57–65
27. Gawne TJ, Kjaer TW, Richmond BJ (1996) Latency: another potential code for feature binding in striate cortex. J Neurophys 76(2):1356–1360
28. Gerstner W, Kistler WM (2002) Spiking neuron models: single neurons, populations, plasticity. Cambridge University Press, Cambridge
29. Gerstner W, Kistler WM, Naud R, Paninski L (2014) Neuronal dynamics: from single neurons to networks and models of cognition. Cambridge University Press, Cambridge
30. Gollisch T, Meister M (2008) Rapid neural coding in the retina with relative spike latencies. Science 319(5866):1108–1111. https://doi.org/10.1126/science.1149639
31. Gray CM, König P, Engel AK, Singer W (1989) Oscillatory responses in cat visual cortex exhibit inter-columnar synchronization which reflects global stimulus properties. Nature 338(6213):334
32. Gray CM, Singer W (1989) Stimulus-specific neuronal oscillations in orientation columns of cat visual cortex. Proc Nat Acad Sci 86(5):1698–1702
33. Hamanaka H, Torikai H, Saito T (2006) Quantized spiking neuron with A/D conversion functions. IEEE Trans Circuit Syst II: Express Briefs 53(10):1049–1053
34. Hawkins J, Ahmad S (2016) Why neurons have thousands of synapses, a theory of sequence memory in neocortex. Frontiers in neural circuits 10. https://doi.org/10.3389/fncir.2016.00023
35. Hawkins J, Blakeslee S (2004) On intelligence: how a new understanding of the brain will lead to the creation of truly intelligent machines. Macmillan, New York
36. Hopfield JJ (1995) Pattern recognition computation using action potential timing for stimulus representation. Nature 376(6535):33
37. Hough M, De Garis H, Korkin M, Gers F, Nawa NE (1999) SPIKER: Analog waveform to digital spike-train conversion in ATR's artificial brain (cam-brain) project. In: International conference on robotics and artificial life. Citeseer
38. Hunsberger E, Eliasmith C (2015) Spiking deep networks with LIF neurons. arXiv preprint arXiv:1510.08829
39. Johansson RS, Birznieks I (2004) First spikes in ensembles of human tactile afferents code complex spatial fingertip events. Nat Neurosci 7(2):170
40. Kasabov N, Scott NM, Tu E, Marks S, Sengupta N, Capecci E, Othman M, Doborjeh MG, Murli N, Hartono R, Espinosa-Ramos JI, Zhou L, Alvi FB, Wang G, Taylor D, Feigin V, Gulyaev S, Mahmoud M, Hou ZG, Yang J (2016) Evolving spatio-temporal data machines based on the NeuCube neuromorphic framework: design methodology and selected applications. Neural Netw 78:1–14. https://doi.org/10.1016/j.neunet.2015.09.011
41. Kasabov NK (2019) Time-space, spiking neural networks and brain-inspired artificial intelligence. Springer, Berlin
42. Kayser C, Montemurro MA, Logothetis NK, Panzeri S (2009) Spike-phase coding boosts and stabilizes information carried by spatial and temporal spike patterns. Neuron 61(4):597–608. https://doi.org/10.1016/j.neuron.2009.01.008

43. Kheradpisheh SR, Ganjtabesh M, Thorpe SJ, Masquelier T (2018) STDP-based spiking deep convolutional neural networks for object recognition. Neural Netw 99:56–67
44. Kim J, Kim H, Huh S, Lee J, Choi K (2018) Deep neural networks with weighted spikes. Neurocomputing 311:373–386. https://doi.org/10.1016/j.neucom.2018.05.087 Publisher: Elsevier
45. Kiselev M (2016) Rate coding vs. temporal coding-is optimum between? In: 2016 international joint conference on neural networks (IJCNN), pp. 1355–1359. IEEE
46. Krizhevsky A (2009) Learning multiple layers of features from tiny images p. 60
47. LeCun Y (1998) The MNIST database of handwritten digits. http://yann.lecun.com/exdb/mnist/ (1998)
48. Li M, Tsien JZ (2017) Neural code-neural self-information theory on how cell-assembly code rises from spike time and neuronal variability. Front Cellular Neurosci 11:236. https://doi.org/10.3389/fncel.2017.00236
49. Lichtsteiner P, Posch C, Delbruck T (2008) A 128$*times*$128 120 dB 15 μs latency asynchronous temporal contrast vision sensor. IEEE J Solid-State Circuit 43(2):566–576. https://doi.org/10.1109/JSSC.2007.914337
50. Liu D, Yue S (2017) Fast unsupervised learning for visual pattern recognition using spike timing dependent plasticity. Neurocomputing 249:212–224
51. Liu S, van Schaik, A, Mincti BA, Delbruck T (2010) Event-Based 64-channel binaural silicon cochlea with Q enhancement mechanisms. In: Proceedings of 2010 IEEE international symposium on circuits and systems, pp. 2027–2030 . https://doi.org/10.1109/ISCAS.2010.5537164
52. Loiselle S, Rouat J, Pressnitzer D, Thorpe S (2005) Exploration of rank order coding with spiking neural networks for speech recognition. In: Proceedings. 2005 IEEE international joint conference on neural networks, 2005., vol. 4, pp. 2076–2080. IEEE
53. Maass W (1995) On the computational complexity of networks of spiking neurons. Advances in Neural Information Processing Systems 7, NIPS Conference, Denver, Colorado, USA, 1994
54. Maass W (1997) Networks of spiking neurons: the third generation of neural network models. Neural Netw 10(9):1659–1671. https://doi.org/10.1016/S0893-6080(97)00011-7
55. Margrie TW, Schaefer AT (2003) Theta oscillation coupled spike latencies yield computational vigour in a mammalian sensory system. J Physiol 546(2):363–374. https://doi.org/10.1113/jphysiol.2002.031245
56. Milde MB, Blum H, Dietmüller A, Sumislawska D, Conradt J, Indiveri G, Sandamirskaya Y (2017) Obstacle avoidance and target acquisition for robot navigation using a mixed signal analog/digital neuromorphic processing system. Front Neurorobot **11**, 28 . https://doi.org/10.3389/fnbot.2017.00028
57. Montemurro MA, Rasch MJ, Murayama Y, Logothetis NK, Panzeri S (2008) Phase-of-firing coding of natural visual stimuli in primary visual cortex. Current Biol 18(5):375–380. https://doi.org/10.1016/j.cub.2008.02.023 Publisher: Elsevier
58. Moser EI, Kropff E, Moser MB (2008) Place cells, grid cells, and the brain's spatial representation system. Ann Rev Neurosci 31:69–89 Publisher: Annual Reviews
59. Neil D, Liu SC (2016) Effective sensor fusion with event-based sensors and deep network architectures. In: 2016 IEEE international symposium on circuits and systems (ISCAS), pp. 2282–2285. IEEE, Montréal, QC, Canada . https://doi.org/10.1109/ISCAS.2016.7539039
60. Nuntalid N, Dhoble K, Kasabov N (2011) EEG classification with BSA spike encoding algorithm and evolving probabilistic spiking neural network. In: Lu BL, Zhang L, Kwok J (eds) Neural information processing, vol 7062. Springer, Berlin Heidelberg, Berlin, Heidelberg, pp 451–460. https://doi.org/10.1007/978-3-642-24955-6_54
61. O'Keefe J, Dostrovsky J (1971) The hippocampus as a spatial map: Preliminary evidence from unit activity in the freely-moving rat. Brain Res 34, 171–175. https://doi.org/10.1016/0006-8993(71)90358-1
62. Olshausen BA, Field DJ (2004) Sparse coding of sensory inputs. Current Opinion Neurobiol 14(4):481–487
63. Orchard G, Jayawant A, Cohen GK, Thakor N (2015) Converting static image datasets to spiking neuromorphic datasets using saccades. Front Neurosci 9:437 Publisher: Frontiers
64. Oswald AMM, Doiron B, Maler L (2007) Interval coding I burst interspike intervals as indicators of stimulus intensity. J Neurophysiol 97(4):2731–2743. https://doi.org/10.1152/jn.00987.2006
65. Panda P, Roy K (2016) Unsupervised regenerative learning of hierarchical features in spiking deep networks for object recognition. arXiv:1602.01510 [cs]
66. Panzeri S, Senatore R, Montemurro MA, Petersen RS (2007) Correcting for the sampling bias problem in spike train information measures. J Neurophysiol 98(3):1064–1072. https://doi.org/10.1152/jn.00559.2007
67. Park S, Kim S, Choe H, Yoon S (2019) Fast and efficient information transmission with burst spikes in deep spiking neural networks. In: 2019 56th ACM/IEEE design automation conference (DAC), pp. 1–6. IEEE (2019)

68. Paulun L, Wendt A, Kasabov N (2018) A retinotopic spiking neural network system for accurate recognition of moving objects using NeuCube and dynamic vision sensors. Front Comput Neurosci 12:42 Publisher: Frontiers
69. Perrinet L, Samuelides M, Thorpe S (2004) Coding static natural images using spiking event times: do neurons cooperate? IEEE Trans Neural Netw 15(5):1164–1175
70. Petro B, Kasabov N, Kiss RM (2020) Selection and optimization of temporal spike encoding methods for spiking neural networks. IEEE Trans Neural Netw Learn Syst 31(2):358–370. https://doi.org/10.1109/TNNLS.2019.2906158
71. Ponulak F, Kasinski A (2011) Introduction to spiking neural networks: Information processing, learning and applications. Acta Neurobiol Experiment 71(4):409–433
72. Portelli G, Barrett JM, Hilgen G, Masquelier T, Maccione A, Di Marco S, Berdondini L, Kornprobst P, Sernagor E (2016) Rank order coding: a retinal information decoding strategy revealed by large-scale multielectrode array retinal recordings. Eneuro 3(3) (2016). https://doi.org/10.1523/ENEURO.0134-15.2016
73. Pérez-Carrasco JA, Zhao B, Serrano C, Acha B, Serrano-Gotarredona T, Chen S, Linares-Barranco B (2013) Mapping from frame-driven to frame-free event-driven vision systems by low-rate rate coding and coincidence processing-application to feedforward convnets. IEEE Trans Pattern Anal Machi Intell 35(11):2706–2719
74. Rolls ET, Franco L, Aggelopoulos NC, Jerez JM (2006) Information in the first spike, the order of spikes, and the number of spikes provided by neurons in the inferior temporal visual cortex. Vis Res 46(25):4193–4205. https://doi.org/10.1016/j.visres.2006.07.026
75. Rueckauer B, Liu SC (2018) Conversion of analog to spiking neural networks using sparse temporal coding. In: 2018 IEEE international symposium on circuits and systems (ISCAS), pp. 1–5 . https://doi.org/10.1109/ISCAS.2018.8351295
76. Rueckauer B, Lungu IA, Hu Y, Pfeiffer M, Liu SC (2017) Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. Front Neurosci 11:682
77. Rullen RV, Thorpe SJ (2001) Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex. Neural Comput 13(6):1255–1283
78. Saal HP, Vijayakumar S, Johansson RS (2009) Information about complex fingertip parameters in individual human tactile afferent neurons. J Neurosci 29(25):8022–8031. https://doi.org/10.1523/JNEUROSCI.0665-09.2009
79. Sboev A, Serenko A, Rybka R, Vlasov D (2020) Solving a classification task by spiking neural network with stdp based on rate and temporal input encoding. mathematical methods in the applied sciences p. mma.6241 . https://doi.org/10.1002/mma.6241
80. Schrauwen B, D'Haene M, Verstraeten D, Van Campenhout J (2008) Compact hardware liquid state machines on FPGA for real-time speech recognition. Neural Netw 21(2–3):511–523
81. Schrauwen B, Van Campenhout J (2003) BSA, a fast and accurate spike train encoding scheme. In: Proceedings of the international joint conference on neural networks, vol. 4, pp. 2825–2830. IEEE Piscataway, NJ
82. Sengupta N, Kasabov N (2017) Spike-time encoding as a data compression technique for pattern recognition of temporal data. Inf Sci 406:133–145
83. Sengupta N, Scott N, Kasabov N (2003) Framework for knowledge driven optimisation based data encoding for brain data modelling using spiking neural network architecture. In: V. Ravi, B.K. Panigrahi, S. Das, P.N. Suganthan (eds.) Proceedings of the fifth international conference on fuzzy and neuro computing (FANCCO - 2015), Advances in intelligent systems and computing, pp. 109–118. Springer International Publishing (2015). https://doi.org/10.1007/978-3-319-27212-2_9. Event-place: Cham
84. Serrano-Gotarredona T, Linares-Barranco B (2015) Poker-DVS and MNIST-DVS. Their history, how they were made, and other details. Front Neurosci 9:481
85. Serre T, Wolf L, Bileschi S, Riesenhuber M, Poggio T (2007) Robust object recognition with cortex-like mechanisms. IEEE Trans Pattern Anal Mach Intell 29(3):411–426
86. Shadlen MN, Newsome WT (1994) Noise, neural codes and cortical organization. Current Opinion Neurobiol 4(4):569–579
87. Sharma V, Srinivasan D (2010) A spiking neural network based on temporal encoding for electricity price time series forecasting in deregulated markets. In: The 2010 international joint conference on neural networks (IJCNN), pp. 1–8. IEEE . https://doi.org/10.1109/IJCNN.2010.5596676. Event-place: Barcelona, Spain
88. Steinmetz PN, Roy A, Fitzgerald P, Hsiao S, Johnson K, Niebur E (2000) Attention modulates synchronized neuronal firing in primate somatosensory cortex. Nature 404(6774):187–190

89. Storchi R, Bale MR, Biella GEM, Petersen RS (2012) Comparison of latency and rate coding for the direction of whisker deflection in the subcortical somatosensory pathway. J Neurophysiol 108(7):1810–1821. https://doi.org/10.1152/jn.00921.2011
90. Stromatias E, Soto M, Serrano-Gotarredona T, Linares-Barranco B (2017) An event-driven classifier for spiking neural networks fed with synthetic or dynamic vision sensor data. Front Neurosci 11:350 Publisher: Frontiers
91. Tavanaei A, Ghodrati M, Kheradpisheh SR, Masquelier T, Maida A (2019) Deep learning in spiking neural networks. Neural Netw 111:47–63
92. Tavanaei A, Maida A (2019) BP-STDP: Approximating backpropagation using spike timing dependent plasticity. Neurocomputing 330:39–47. https://doi.org/10.1016/j.neucom.2018.11.014
93. Tavanaei A, Maida AS (2017) Multi-layer unsupervised learning in a spiking convolutional neural network. In: 2017 international joint conference on neural networks (IJCNN), pp. 2023–2030 . https://doi.org/10.1109/IJCNN.2017.7966099. ISSN: 2161-4407
94. Thorpe S, Delorme A, Van Rullen R (2001) Spike-based strategies for rapid processing. Neural Netw 14(6–7):715–725
95. Thorpe S, Fize D, Marlot C (1996) Speed of processing in the human visual system. Nature 381(6582):520
96. Thorpe S, Gautrais J (1998) Rank order coding. In: Computational neuroscience, pp. 113–118. Springer
97. Thorpe SJ (1990) Spike arrival times: A highly efficient coding scheme for neural networks. Parallel processing in neural systems pp. 91–94
98. Truong SN, Pham KV, Min KS (2018) Spatial-pooling memristor crossbar converting sensory information to sparse distributed representation of cortical neurons. IEEE Trans Nanotechnol 17(3):10
99. Turnbull L, Dian E, Gross G (2005) The string method of burst identification in neuronal spike trains. J Neurosci Methods 145(1–2):23–35. https://doi.org/10.1016/j.jneumeth.2004.11.020
100. VanRullen R, Guyonneau R, Thorpe SJ (2005) Spike times make sense. Trends Neurosci 28(1):1–4
101. Wu S, Si Amari, Nakahara H (2002) Population coding and decoding in a neural field: a computational study. Neural Comput 14(5):999–1026. https://doi.org/10.1162/089976602753633367
102. Wysoski SG, Benuskova L, Kasabov N (2007) Adaptive spiking neural networks for audiovisual pattern recognition. In: International conference on neural information processing, pp. 406–415. Springer (2007)
103. Wysoski SG, Benuskova L, Kasabov N (2007) Text-independent speaker authentication with spiking neural networks. In: International conference on artificial neural networks, pp. 758–767. Springer (2007)
104. Wysoski SG, Benuskova L, Kasabov N (2008) Fast and adaptive network of spiking neurons for multi-view visual pattern recognition. Neurocomputing 71(13):2563–2575. https://doi.org/10.1016/j.neucom.2007.12.038
105. Yu Q, Tang H, Tan KC, Li H (2013) Rapid feedforward computation by temporal encoding and learning with spiking neurons. IEEE Trans Neural Netw Learn Syst 24(10):1539–1552
106. Yu Q, Tang H, Tan KC, Yu H (2014) A brain-inspired spiking neural network model with temporal encoding and learning. Neurocomputing 138:3–13
107. Zambrano D, Bohte SM (2016) Fast and efficient asynchronous neural computation with adapting spiking neural networks. arXiv preprint arXiv:1609.02053
108. Zeldenrust F, Wadman WJ, Englitz B (2018) Neural coding with bursts - current state and future perspectives. Front Comput Neurosci 12:48. https://doi.org/10.3389/fncom.2018.00048
109. Zhang L, Zhou S, Zhi T, Du Z, Chen Y (2019) Tdsnn: From deep neural networks to deep spike neural networks with temporal-coding. Proc AAAI Conf Artif Intell 33:1319–1326
110. Zhang M, Zheng N, Ma D, Pan G, Gu Z (2018) Efficient spiking neural networks with logarithmic temporal coding. arXiv preprint arXiv:1811.04233

Terms and Conditions