SpikeSMOKE: Spiking Neural Networks for Monocular 3D Object Detection with Cross-Scale Gated Coding

Xuemei Chen, Huamin Wang, Hangchi Shen, Shukai Duan *Member, IEEE,* Shiping Wen, *Senior Member, IEEE,*and Tingwen Huang, *Fellow, IEEE*

***Abstract*—Low energy consumption for 3D object detection is an important research area because of the increasing energy consumption with their wide application in fields such as autonomous driving. The spiking neural networks (SNNs) with low-power consumption characteristics can provide a novel solution for this research. Therefore, we apply SNNs to monocular 3D object detection and propose the SpikeSMOKE architecture in this paper, which is a new attempt for low-power monocular 3D object detection. As we all know, discrete signals of SNNs will generate information loss and limit their feature expression ability compared with the artificial neural networks (ANNs). In order to address this issue, inspired by the filtering mechanism of biological neuronal synapses, we propose a cross-scale gated coding mechanism(CSGC), which can enhance feature representation by combining cross-scale fusion of attentional methods and gated filtering mechanisms. In addition, to reduce the computation and increase the speed of training, we present a novel light-weight residual block that can maintain spiking computing paradigm and the highest possible detection performance. Compared to the baseline SpikeSMOKE under the 3D Object Detection, the proposed SpikeSMOKE with CSGC can achieve 11.78 (+2.82, Easy), 10.69 (+3.2, Moderate), and 10.48 (+3.17, Hard) on the KITTI autonomous driving dataset by $AP|_{R_{11}}$ at 0.7 IoU threshold, respectively. It is important to note that the results of SpikeSMOKE can significantly reduce energy consumption compared to the results on SMOKE. For example, the energy consumption can be reduced by 72.2% on the hard category, while the detection performance is reduced by only 4%. SpikeSMOKE-L (lightweight) can further reduce the amount of parameters by 3 times and computation by 10 times compared to SMOKE.**

***Index Terms*—Spiking Neural Networks, Monocular 3D Object Detection, Gating Coding, Lightweighting.**

I. Introduction

Although ANNs have shown excellent performance in various fields including computer vision (CV) [1], speech recognition (SR) [2], natural language processing (NLP) [3], etc, they are facing with skyrocketing problems of energy consumption, with the increasingly complex models and progressively large volumes of data. To solve this problem, some researchers are turning to brain-inspired SNNs, because they have excellent characteristics of event-driven computation [4], biologically interpretable [5], and asynchronous temporal processing.

At present, there has been a great deal of excellent research on SNNs applied to computer vision (CV), natural language processing (NLP) and other fields [6] [7] [8] [9] [10] [11] [5] [12]. For instance, advancements from Spiking ResNet [13] to EMS-ResNet [11] and then to MS-ResNet [12] introduced increasingly efficient event-driven network architectures. Spikformer [14], which is the initial utilization of Transformer in SNNs, followed by ongoing enhancements, led to the introduction of Spike-Driven Transformer v1/v2 [15] [16]. This innovation offers robust sequence modeling capabilities and heightened parallel processing efficiency for intricate tasks. For object detection, current research focuses on 2D object detection using 2D image data or 3D object detection using 3D data. For example, in Spiking-PointNet [17], point clouds—which represent 3D spatial information—were utilized for the purpose of detecting 3D objects, marking the first successful application of point clouds within SNNs. In [18], a spike-driven object detection framework SpikeYOLO was proposed for 2D image data. However, due to the lack of depth information, monocular 3D object detection using 2D image data for 3D object detection has not yet been explored through SNNs, which is unable to utilize the low-power characteristics of SNNs.

Monocular 3D object detection combines 2D image information and depth estimation to achieve detection and localization of 3D objects, which has been widely used in the field of autonomous driving [19] [20] [21] [22] [23]. Currently, there are many excellent models in the field of ANNs. For example, MonoWAD [24] introduced an innovative weather-adaptive diffusion model for monocular 3D object detection, LabelDistill [25] presented label-guided cross-modal knowledge distillation for camera-based 3D object detection, Mono3DVG [26] offered a fresh perspective by detecting 3D objects through detailed descriptions of visual appearance, and MonoCD [27], a network architecture based on auxiliary information. However, when applying monocular 3D object detection to resource-constrained edge devices like autonomous driving [28], its practical deployment will be restricted. So it is very important to implement a network model with low-power characteristics for monocular 3D object detection. Naturally, brain-like SNNs model with low-power characteristics, can offer us a novel insight to solve this problem.

Based on the above analysis, we apply SNNs to monocular 3D object detection to construct a novel architecture SpikeSMOKE. In order to shrink the gap between the discrete signals of SNNs and the continuous signals of ANNs for feature expressiveness, we propose a cross-scale gated coding mechanism (CSGC) inspired by the filtering mechanism of biological neuronal synapses, which can reduce information loss and improve the detection performance of the model by fusing multi-scale attention, as shown in Figure 1. To further reduce the model's power consumption while maintaining the spike computation paradigm, we propose innovative light-weight residual blocks. Numerous experiments have demonstrated on the autonomous driving KITTI dataset and the CIFAR10/100 datasets. The contributions of this paper can be summarized as follows.

(1) To enhance the energy efficiency of monocular 3D object detection, we draw inspiration from brain-like SNNs, known for their low-power properties, and leverage the simplicity of the SMOKE network to construct a novel architecture called SpikeSMOKE.

(2) In this SpikeSMOKE, we introduce the parallel cross-scale gated coding mechanism CSGC based on attention to improve the information representation capability of the model. At the same time, we also propose a light-weight residual block on the SpikeSMOKE model.

(3) The SpikeSMOKE can significantly decrease energy consumption compared to the results with SMOKE for 3D Object Detection. For instance, energy usage can be reduced by 72.2% in the hard category, with merely a 4% dip in detection performance.

(4)The SpikeSMOKE with CSGC can achieve better results on the KITTI dataset, in comparison to the baseline SpikeSMOKE. Moreover, in order to validate the generalizability of the proposed CSGC encoding strategy, we have also done experiments on the CIFAR-10/100 datasets to validate this.

This paper is organized as follows. In section II, we describe the kinetic formulation of LIF for spike neurons, the model architecture for monocular 3D object detection, the coding mechanism, and lightweighting methods. In section III, we describe the macroscopic architecture of SpikeSMOKE, the detailed implementation of cross-scale gated coding (CSGC) and lightweighted residual structures. In the IV section, we present the experimental results of our methods and the results of ablation experiments. In the V section, we summarize the paper.

II. Related works

A. Spike Neuron

Spike neurons simulate the behavior of biological neurons transmitting signal sequences through synapses, transmitting temporal signals and spatial information through the membrane voltage U. The commonly used neurons in spiking neural networks consist of integrate-and-fire (IF) [29] neurons and leaky integrate-and-fire (LIF) [30]neurons. We used LIF neurons and described their neuronal dynamics formulation as follows:

$$\begin{aligned} U_i^l[t] &= H_i^l[t-1] + \sum_j W_{ij} S_j^l[t] \\ S_i^l[t] &= Hea(U_i^l[t] - Uth) = \begin{cases} 1, U_i^l[t] > U_{th} \\ 0, U_i^l[t] < U_{th} \end{cases} \\ H_i^l[t] &= \tau U_i^l[t](1 - S_i^l[t]) + U_{reset} S_i^l[t] \end{aligned} \tag{1}$$

$U_i^l[t]$ denotes the membrane voltage of neuron i in layer l, W_{ij} indicates the connection weights of layer l and layer l-1, $S_j^l[t]$ represents the spike of the jth neuron in layer l-1, and Hea is the Heaviside function. At discrete time step T, when the membrane voltage of the neuron is greater than the spike neuron threshold, i. e. , $U_i^l[t] > U_{th}$, the neuron emit a spike [31]. τ denotes the decay coefficient. After issuing spike 1, the membrane voltage is usually reset to 0.

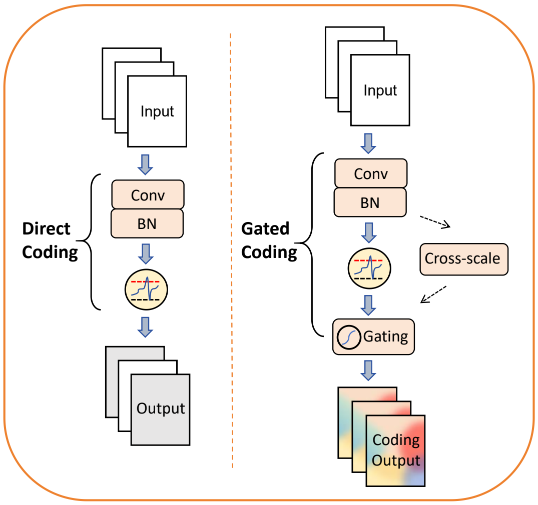

Fig. 1: Gated coding is different from direct coding. The proposed gated coding mechanism utilizes a cross-scale fusion module that incorporates a gating unit (CSGC) to mimic the filtering mechanism of biological neuronal synapses to produce a filtering effect on the input feature map.

B. Monocular 3D Object Detection

Common approaches for monocular 3D object detection include Two-stage methods (MonoLSS [32]), Single-stage anchor-based methods (GrooMeD-NMS [33]) and Single-stage anchor-free methods (SMOKE [34], MonoCD [27], MonoDGP [35]). Compared with the first method, which is prone to introducing 2D noise [34], and the second method, which usually requires complex data preprocessing and non-maximum suppression [36] [37], the third method has the advantages of a simple model and high computational efficiency. In this paper, we use the SMOKE [34] architecture of Single-stage anchor-free methods as a base architecture, which both ignores the redundancy associated with 2D detection frameworks and does not require additional data. The SMOKE architecture continues the keypoint detection of centernet [38], which discards the traditional 2D detection module and retains only the 3D detection part, and improves the parameter convergence and detection accuracy by multi-step disentanglement. The 3D bounding box is obtained by predicting the object's 3D projection center on the image plane along with attribute variables. The attribute regression of the 3D bounding box is decoupled into an 8-tuple $(\alpha_x, \alpha_y, \alpha_z, \alpha_l, \alpha_w, \alpha_h, sin\beta, cos\beta)$. These parameters are optimized by a loss function, which is transformed to obtain the true parameters $(x, y, z, w, h, l, \theta)$ used to construct the 3D bounding box.

C. Coding Mechanism

In a spiking neural network it is necessary to first encode the data into the form of spikes, there are several common ways of coding:rating coding [39] encodes information based on the average rate of neuron firing, phase coding [40] is expressed by the temporal position of the spike relative to a certain reference event, temporal coding [41], [42], [43] represents information based on the time interval and order of the spikes, and frequency coding involves the frequency at which neurons emit spikes over a specific time period. In addition to this, there are various complex coding methods ([44] [45] [46] [47]), and choosing an appropriate coding method is crucial for the representation performance of the model, in order to reduce the gap between the discrete and continuous signals in the spiking neural network, we propose a novel gated coding method in conjunction with the cross-scale fusion unit.

D. Light-Weight Model

Light-weight model refers to reducing the number of parameters and computation and compressing the model training time by removing or transforming part of the model architecture and connections. There are several common methods for lightweighting models in neural networks, a) Pruning: Chen et al [48] proposed an SNN pruning regeneration mechanism based on neuron criticality to remove unnecessary connections or neurons in the network. SSNN [49] designed sparse RSNN by pruning randomly initialized model. b) Model quantization: Spiking-Diffusion [50] proposed a diffusion model based on vector quantization discretization. c) Model distillation: TSSD [51] proposed a spatio-temporal distillation method. d) Depth-wise separable convolutions: the first proposed in MobileNets [52], and later extended and applied widely in the field of ANN ([53], [54], [55], [56]). There has been a great deal of experimental or theoretical evidence for the effectiveness of depth-wise separable convolutions, and our proposed lightweight residual unit originates from this idea.

III. Methods

A. SpikeSMOKE

SpikeSMOKE architecture, as shown in Figure 2, takes the single-stage object detection network SMOKE as a infrastructure and retains its macro-architecture, because this architecture can ignore the redundancy of 2D detection framework and does not require additional data.

Backbone: Since DLA34 can take advantage of fusing feature maps of different scales through the depth convergence feature, we use the spike firing rate of the spike neurons to mimic its ReLU activation function, converting it to Spike-DLA34 as our backbone.

Neck: In order to ensure that the model is fully spike-driven, we add a spike neuron before each vanilla convolution in the DLAup structure, which can use deformable convolution to improve feature representation.

Head: This component consists two branches. One is keypoint classification used by heatmap, the other is 3D bounding box regression. This two branches can process the feature maps from the Neck network to obtain the 3D object detection results.

B. Cross-Scale Gated Coding (CSGC)

For ANN2SNN, the information loss is inevitable because the continuous signals are transformed to the discrete signals. In order to improve the feature expression capability, we propose a cross-scale fusion of attention coding unit named as CSGC. This method can utilize different convolution kernel to fuse cross-scale contextual information.

For CSGC, we design a parallel architecture including channel attention and spatial attention and a gate control unit. The input $x \in R^{B \times C \times H \times W}$ is firstly repeated at each time step to obtain information in the time dimension. Then, in the channel attention part, we use a Linear-ReLU-Linear structure to learn and update the weights in the channel dimension by the Equation (2):

$$CA(x) = Linear\left(ReLU(Linear(x))\right). \tag{2}$$

In the spatial attention part, we conduct feature extraction on the same feature map by different convolution kernel (Equation (3)). Small convolution kernels are employed to capture local subtle features for small objects, enabling the detection of small-sized objects. Conversely, larger convolution kernels are utilized to cover larger areas for detecting large objects and effectively capturing global information. We employ three different sizes of convolution kernels and assign dynamic weights to the feature maps after different convolutional processing by the learnable parameters α, β and γ, respectively. At the same time, we invoke the residual idea of [57] to connect the original feature map with the output part.

$$\begin{aligned} SA(x) =& Conv_{3\times3}\left(ReLU\left(Conv_{3\times3}(x)\right)\right)\cdot\alpha + \\ & Conv_{5\times5}\left(ReLU\left(Conv_{5\times5}(x)\right)\right)\cdot\beta + \\ & Conv_{7\times7}\left(ReLU\left(Conv_{7\times7}(x)\right)\right)\cdot\gamma + \\ & identity. \end{aligned} \tag{3}$$

The output spike information after the above channel and spatial attention processing is controlled and adjusted by a Sigmoid gating unit, which is defined as Equation (4):

$$O(x) = Sig\left(SA(x) \odot CA(x)\right), \tag{4}$$

where $\odot$ denotes the Hadamard product. Then, it performs the Hadamard product operation with the output of the spiking neuron, imitating the filtering function of the synapses, as defined by Equation (5):

$$\widehat{O}(x) = O(x) \odot SNN(x). \tag{5}$$

C. Light-weight Residual

A novel light-weight residual block, as shown in Figure 3, is introduced to decrease model parameters and computational load, consequently accelerating its training speed, by integrating depth-wise separable convolutions and membrane shortcut method. Specifically, the depth-wise convolution is used to conduct a separate convolution operation for each input channel, which implies that redundant computations can be significantly reduced because it perform only one convolution per channel. Then, a point-wise convolution is applied to the output of depth-wise convolution. After that,

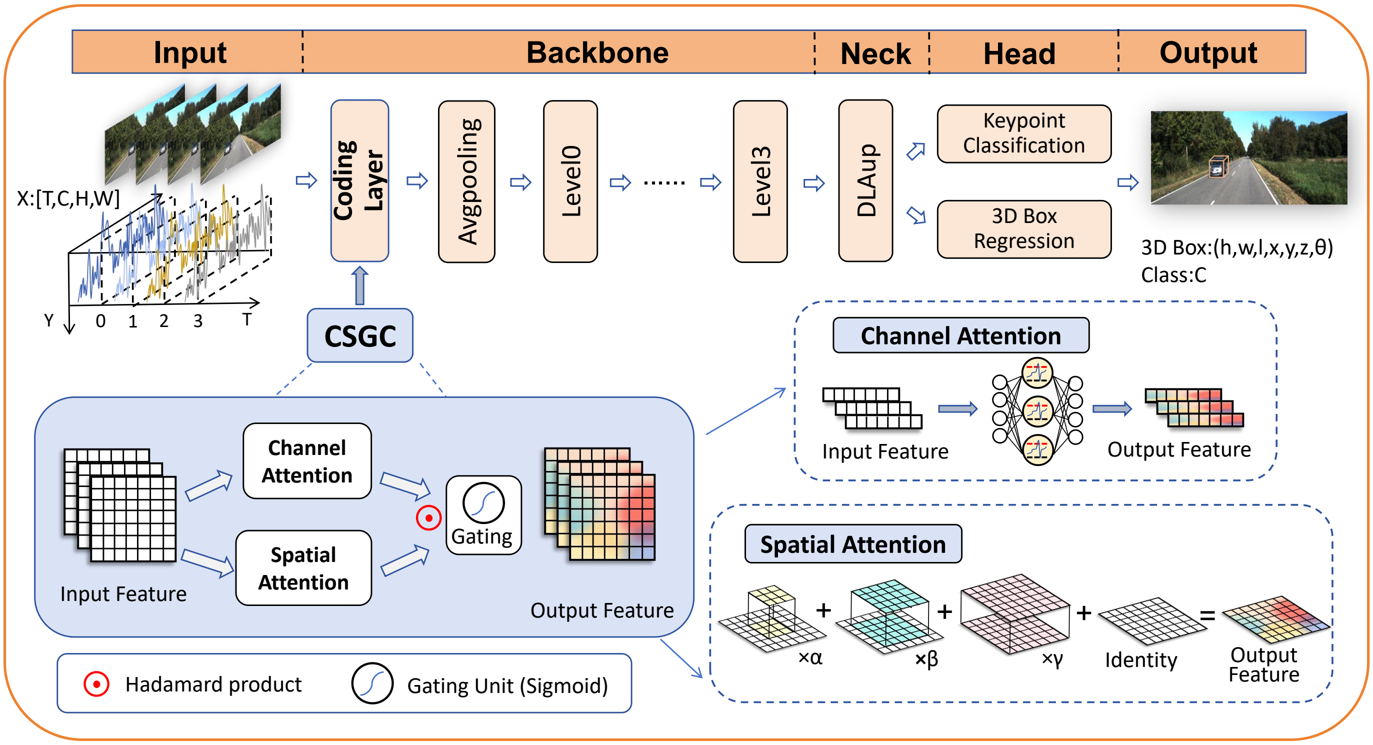

Fig. 2: SpikeSMOKE overall network architecture. The input is an image and the output is the 3D bounding box of the object detection and the object category. A CSGC coding mechanism is introduced, based on a cross-scale attention fusion, to reduce the performance loss due to data transformation and enhance the representation of complex features.

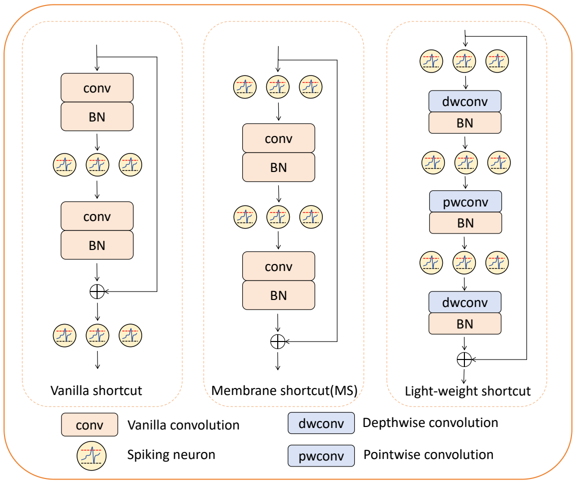

Fig. 3: Light-weight residual blocks. The rightmost figure uses depth-wise convolution and point-wise convolution to replace vanilla convolution.

another depth-wise convolution is applied. It should be noted that a membrane-based shortcut pathway is utilized to remain identity mapping.

For regular membrane-based residual block, the computational volume and parametric quantities can be calculated by the Equation (6-1) and the Equation (6-2) respectively:

$$k \cdot k \cdot Cin \cdot Cout \cdot Wout \cdot Hout, \tag{6-1}$$

$$k \cdot k \cdot Cin \cdot Cout. \tag{6-2}$$

When the above light-weight residual block is introduced, the computational volume and parametric quantities can be calculated by the Equation (7-1) and the Equation (7-2) respectively:

$$k \cdot k \cdot Cin \cdot Wout \cdot Hout + Cin \cdot Cout \cdot Wout \cdot Hout, \tag{7-1}$$

$$k \cdot k \cdot Cin + Cin \cdot Cout. \tag{7-2}$$

For the SpikeSMOKE with light-weight residual block, the convolution kernel size k is 3×3, $C_{\text{in}} = 2C_{\text{out}}$, $W_{\text{out}} = \frac{1}{2}W_{\text{in}}$, $H_{\text{out}} = \frac{1}{2}H_{\text{in}}$. Then, the ratio of the computational effort can be calculated by the Equation (8):

$$\frac{Cin \cdot Wout \cdot Hout(3k^2 + 2Cin)}{k^2 \cdot CinCout \cdot Wout \cdot Hout(1 + 4Cin)} = \frac{1}{2Cin} + \frac{1}{27}. \tag{8}$$

Similarly, the ratio of their parametric quantities is the Equation (9):

$$\frac{k^2 \cdot Cin + Cin \cdot Caut + k^2Cout}{k^2Cin \cdot Cout + k^2Cout \cdot Cout} = \frac{1}{2Cin} + \frac{1}{27}. \tag{9}$$

Observably, compared to the pre-improvement phase, the proposed light-weight residual block diminishes computational and parametric quantities by approximately 27 times.

IV. Experiments

A. Datasets

KITTI.

The KITTI dataset is widely used in the field of autonomous driving and computer vision. The monocular 3D object detection utilizes single-camera data for stereo object detection across the Car, Pedestrian, and Cyclist categories, with 3,712 pictures in the training set and 3,769 pictures in the validation set, encompassing diverse scenarios such as city roads and highways. The detected objects include 3D bounding boxes, bird's eye view bounding boxes, 2D bounding boxes, and can be classified into Easy, Moderate, and Hard categories based on detection complexity. We used an evaluation metric called 11-point Interpolated Average Precision metric [58], which approximates the shape of the Precision/Recall curve, at each difficulty level. It is defined as the Equation (10):

$$AP|N = \frac{1}{|N|}\sum_{n\in N} f_{inter}(n) \tag{10}$$

Where N denotes the set of recall levels that represent the exact intervals, this experiment takes $R_{11} = \{\frac{1}{10}, \frac{2}{10}, \frac{3}{10}, \cdots, 1\}$. $f_{inter}(n)$ denotes the precision obtained after interpolation calculation, instead of averaging the precision values, the maximum precision greater than or equal to the current recall is taken in calculating the precision. It is defined as the Equation (11):

$$f_{inter}(n) = \max_{n':n'\geq n} f(n') \tag{11}$$

CIFAR-10/100.

The CIFAR-10 and CIFAR-100 datasets are commonly used for image classification. The CIFAR-10 dataset contains 10 categories of color images, each containing 6000 images of size 32x32 pixels, for a total of 60, 000 images. The CIFAR-100 dataset extends CIFAR-10 with 100 categories, each containing 600 images of size 32x32 pixels. contains 600 images, for a total of 60, 000 images with image size of 32x32 pixels.

B. Experiments Setup

We used random horizontal flip, random scale and shift as data augmentation aiming to increase the diversity of training data. In the network design, we set the number of groups for group normalization to 32 and for less than 32, we take 16. based on the analysis in literature [59], we set $\left[\bar{h},\bar{w},\bar{l},\right]^\top = [1.63, 1.53, 3.88]^\top and \left[\mu_z, \sigma_z\right]^\top = [28.01, 16.32]^\top$. The input image resolution size of the network is 1280x384, and after multiple downsampling, the size is reduced to 32 times of the original size. We trained 172 epoch using 4×4090 GPUs, with a size of 4 per batch and a learning rate set to 1.25×10^{-4}. The decay strategy for the learning rate was to reduce the learning rate to 10 times of the original at epoch 47 and 90. We used a threshold of 0.25 to filter the detection object.

C. Main Results

Object Detection Performance on KITTI.

We validate the performance on the validation and test sets of the KITTI dataset for 3D object detection and bird's-eye view under three levels of Easy, Moderate and Hard. * denotes the experimental results obtained by us after configuring the environment and training based on the SMOKE. The 3d object detection of our proposed SpikeSMOKE with CSGC (SpikeSMOKE-CSGC) can reach 28.83/11.78 (Easy), 22.75/10.69 (Moderate), 19.44/10.48 (Hard), and for the BEV detection, they can reach 36.23/15.67 (Easy), 26.88/13.68 (Moderate), 25.75/11.83 (Hard), as shown in Table I, where the metrics is evaluated by $AP|_{R_{11}}$ at 0.5/0.7 IoU threshold. From the results of these experiments, it can be found that although our detection performance has a little gap compared with the SMOKE-ANN*, energy consumption has been significantly reduced. For example, we calculate the energy consumption of the hard category for the 0.7 IoU threshold and find that it can be reduced by 72.2%, while the detection performance is reduced by only 4%. The formula for calculating energy consumption is as follows:

$$\begin{aligned} EC_{SNN} &= Synapsed^{SNN}_{activated} \times 0.9, \\ EC_{ANN} &= FLOPS^{ANN} \times 4.6, \end{aligned} \tag{12}$$

where 0.9 denotes the consumption per accumulator operation and 4.6 denotes the multiplier word operation.

It is well known that monocular 3D object detection is commonly used in resource-constrained scenarios such as edge devices, embedded devices, etc., therefore, lightweighting is an important part of research. Based on this, we further discuss SpikeSMOKE-CSGC's lightweighting by light-weight residul block. The experiments show that the number of parameters is only 1/3 of SMOKE-ANN*, and the amount of computation is only 1/10, as shown in Table II. As a result, the SpikeSMOKE model may provide a new effective solution to reduce the energy consumption of monocular 3D object detection and improve its energy efficiency.

Comparing with the baseline SpikeSMOKE, the proposed SpikeSMOKE-CSGC shows significant improvements in both 3D object detection and BEV detection, with gains of 2.82, 3.2, and 3.17 for 3D object detection, and 3.6, 3.73, and 2.51 for BEV detection, as displayed in Table I. The results of the SpikeSMOKE-CSGC with lightweight treatment (SpikeSMOKE-LCSGC) can also demonstrate notable enhancements compared to that of the SpikeSMOKE-L, as indicated in Table II. Therefore, the improved CSGC method has a significant effect on 2D/3D object detection.

Efficiency and Generalizability Validation for CSGC.

We validate the classification results of the proposed CSGC coding strategy on MS-ResNet18 on the classification dataset CIFAR-10/100, as shown in Table IV and Table V. We note that the accuracy rate of the MS-ResNet18 with CSGC coding on the CIFAR-10 is 1.06% higher than the direct coding method. On the CIFAR-100, the proposed method MS-ResNet18-SNN with CSGC coding can achieve 79.58% by 6 time steps, which is 3.17% higher than MS-ResNet18-SNN. Consequently, our proposed CSGC coding method is effective and generalized.

Ablation Studies on the Effects of Various Attentions.

We performed ablation experiments on 2D object detection about spatial attention (SA) and channel attention (CA) of CSGC at time steps of 4, 6, and 8, respectively, as shown

TABLE I: The results of 3D object detection and detection under BEV are performed on the validation set of KITTI dataset for the car class.

Methods	Parameters (M)	Power (pJ)	3D Object Detection			Birds' Eye View		
			Easy	Moderate	Hard	Easy	Moderate	Hard
SMOKE-ANN*(0.5)	19.51	2.17E+11	29.16	25.22	21.47	32.52	29.59	25.86
SpikeSMOKE(0.5)	19.51	5.97E+10	20.87	19.72	16.89	27.64	22.47	21.65
SpikeSMOKE-CSGC(0.5)	**19.56**	**6.04E+10**	**28.83**	**22.75**	**19.44**	**36.23**	**26.88**	**25.75**
SMOKE-ANN(0.7) [34]	19.51	2.17E+11	14.76	12.85	11.5	19.99	15.61	15.28
SMOKE-ANN*(0.7)	19.51	2.17E+11	12.03	11.14	10.92	17.97	13.08	12.06
SpikeSMOKE(0.7)	19.51	5.97E+10	8.96	7.49	7.31	12.07	9.95	9.32
SpikeSMOKE-CSGC(0.7)	**19.56**	**6.04E+10**	**11.78**	**10.69**	**10.48**	**15.67**	**13.68**	**11.83**

(0.5/0.7) indicates that the metrics are evaluated by $AP|_{R_{11}}$ at the 0.5/0.7 IoU thresholds.

TABLE II: The results of 3D object detection and detection under BEV are performed on the test set of KITTI dataset for the car class.

Methods	Parameters (M)	Power (pJ)	3D Object Detection			Birds' Eye View		
			Easy	Moderate	Hard	Easy	Moderate	Hard
SMOKE-ANN*	19.51	2.17E+11	28.97	23.57	20.21	30.48	27.76	25.81
SpikeSMOKE	**19.51**	**5.97E+10**	**21.80**	**15.23**	**14.99**	**25.99**	**20.69**	**17.92**
SpikeSMOKE-L	6.32	2.24E+10	19.46	10.52	10.23	24.01	15.74	15.37
SpikeSMOKE -LCSGC	6.37	2.29E+10	20.64	15.32	12.93	25.15	18.98	18.08

The table is evaluated using the $AP|_{R_{11}}$ with a 0.5 IoU threshold.

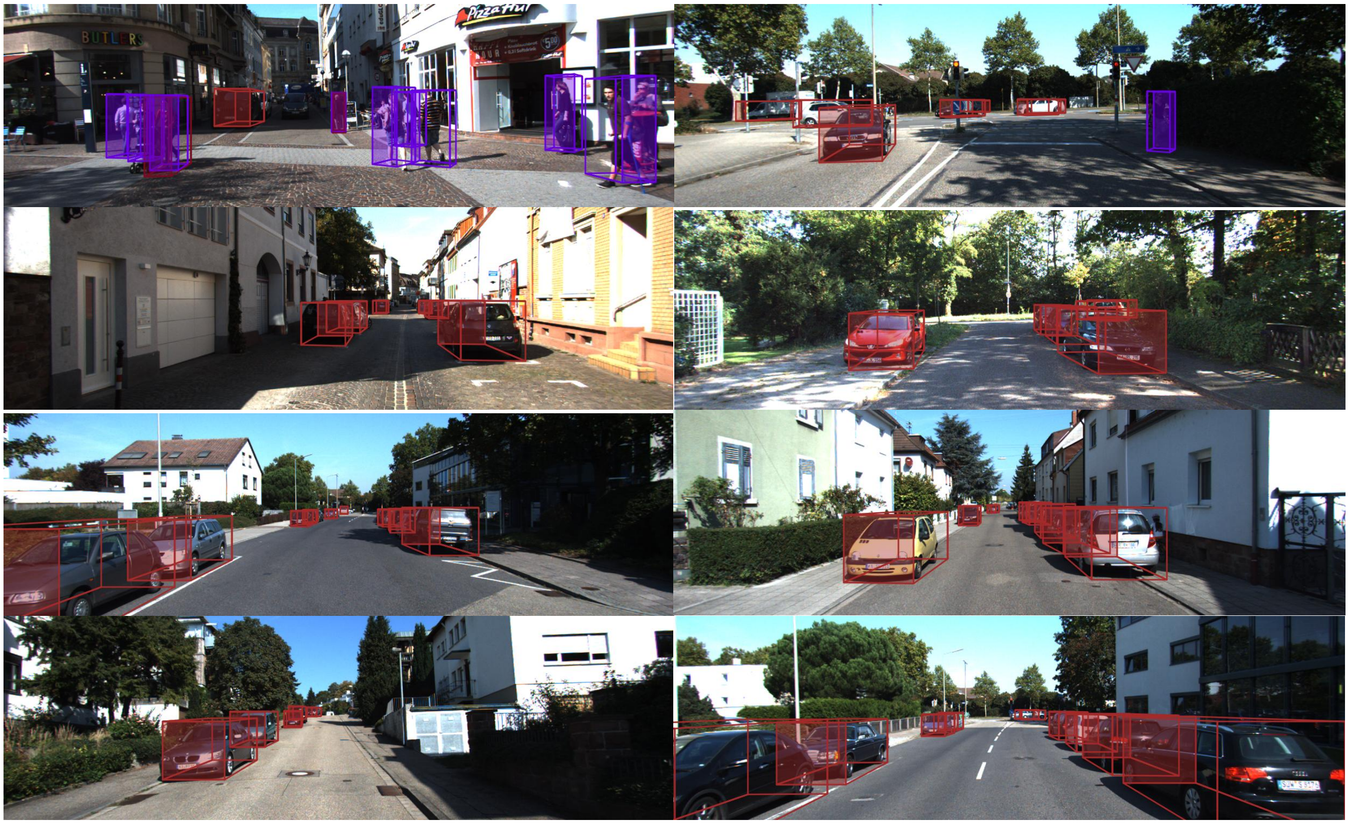

Fig. 4: The visualization results from monocular 3D object detection on the KITTI dataset present a clear and intuitive representation of the performance and accuracy of the detection algorithm in identifying and localizing 3D objects within the complex road scenarios depicted in the dataset.

TABLE III: The results of 2D object detection is performed on the validation set of KITTI dataset for the car class.

Methods	2D Object Detection		
	Easy	Moderate	Hard
SMOKE-ANN*	80.49	72.01	68.76
SpikeSMOKE	73.32	62.85	55.67
SpikeSMOKE-CSGC	**75.58**	**65.49**	**64.37**
SpikeSMOKE-L	51.59	44.01	42.71
SpikeSMOKE-LCSGC	52.33	49.04	43.04

TABLE IV: Classification results on the classification task dataset CIFAR-10.

Architecture	Coding Schemes	Time Steps	CIFAR10 Acc. (%)
ResNet-19 [44]	Phase Coding	8	91.40
VGG-16 [44]	Temporal Coding	100	92.68
ResNet-19 [44]	Rate Coding	6	93.16
MS-ResNet-18	Direct Coding	6	94.92
MS-ResNet-18	**CSGC Coding**	**6**	**95.98(+1.06)**

in Figure 5. We observe that the detection performance is better than that of the baseline when SA or CA is used, which means that each module of CSGC is effective. Obviously, when CA and SA are used together, their detection results are better. Additionally, as the time step increases, the detection performance gets better.

Ablation Studies of Individual Neuronal Thresholds.

Since the threshold of the spike neurons has a great impact on the detection performance of the model, we conduct some ablation experiments on 2D detection by different neuron thresholds in order to get the most suitable LIF neuron threshold. Based on the experiment results, we obtain that the best performance is obtained when the neuron threshold vth=0.75, which is used in this paper, as shown in Table VI.

TABLE V: Classification results on the classification task dataset CIFAR-100.

Methods	Architecture	Spike	Params (M)	Time Steps	CIFAR100 Acc. (%)
ANN [44]	MS-ResNet-18	×	12.54	N/A	80.67
MS-ResNet-SNN [44]	MS-ResNet-18	✓	12.54	6	76.41
CSGC-SNN	MS-ResNet-18	✓	12.72	6	79.58
	MS-ResNet-18	✓	12.72	4	77.97

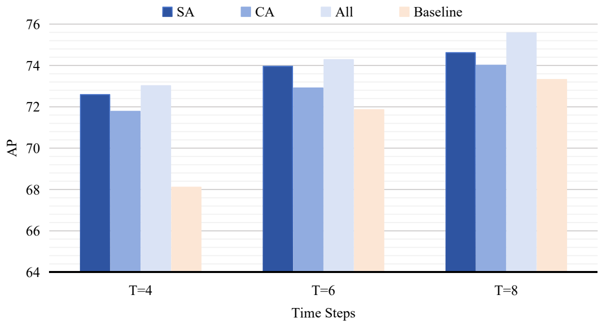

Fig. 5: Ablation experiments in CSGC with different attention modules at different time steps.

Qualitative Results.

We provide visualization results of the monocular 3D object detection on the KITTI dataset as shown in Figure 4. Through these results, we can visually observe the model's capability to accurately identify and locate 3D objects in complex road scenarios, thereby affirming the significant potential application of this technology in areas such as autonomous driving.

V. Conclusion

With the widespread use of 3D object detection in applications such as autonomous driving, the low energy consumption problem is getting more and more attention. As widely acknowledged, low-power consumption stands out as a key feature of brain-like SNNs, offering a potential new solution for energy-efficient 3D object detection. Based on this, we construct a novel SpikeSMOKE architecture for monocular 3D object detection. Since the discrete signaling characteristics of SNNs may result in information loss and restrict their capacity for feature representation, we propose a CSGC mechanism, inspired by the synaptic filtering process in biological neurons. Furthermore, we also propose a lightweight residual block to reduce the computational effort while maintaining the impulse computation paradigm. The experimental results on the KITTI dataset, show that SpikeSMOKE can achieve higher energy efficiency compared to SMOKE, e.g., 72.2% reduction in energy consumption on Hard category, while the detection performance drops by only 4%. Moreover, the experimental results also show that the CSGC-based SpikeSMOKE can achieve significant improvement over the baseline SpikeSMOKE. SpikeSMOKE-L (lightweight) can further reduce the amount of parameters by 3 times and computation by 10 times compared to SMOKE. In the CIFAR-10/100 classification task, the CSGC encoding strategy improves the correctness by 1.06% and 3.17%, respectively, validating its generality. Overall, the SpikeSMOKE architecture and CSGC mechanism can provide an efficient and feasible solution for low-power monocular 3D object detection.

References

[1] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 25, Dec. 2012, pp. 1–9.

[2] A. Hannun, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[3] J. D. M.-W. C. Kenton and L. K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of naacL-HLT*, vol. 1, 2019, pp. 2–18.

[4] Y. Zhu, Z. Yu, W. Fang, X. Xie, T. Huang, and T. Masquelier, "Training spiking neural networks with event-driven backpropagation," in *Advances in Neural Information Processing Systems(NeurIPS)*, vol. 35, Dec. 2022, pp. 30 528–30 541.

[5] Y. Hu, H. Tang, and G. Pan, "Spiking deep residual networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 34, no. 8, pp. 5200–5205, 2021.

[6] M. Yao, G. Zhao, H. Zhang, Y. Hu, L. Deng, Y. Tian, B. Xu, and G. Li, "Attention spiking neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 8, pp. 9393–9410, 2023.

[7] Y. Wu, R. Zhao, J. Zhu, F. Chen, M. Xu, G. Li, S. Song, L. Deng, G. Wang, H. Zheng *et al.*, "Brain-inspired global-local learning incorporated with neuromorphic computing," *Nature Communications*, vol. 13, no. 1, pp. 65–79, 2022.

TABLE VI: Ablation experiments on the threshold of spike neurons.

Methods	vth=0.25	vth=0.5	vth=0.75	vth=1.0	2D Object Detection		
					Easy	Moderate	Hard
SpikeSMOKE-CSGC	✓				64.31	53.84	51.62
		✓			72.06	61.27	60.67
			✓		**75.58**	**65.49**	**64.37**
				✓	54.10	45.88	44.98

[8] W. Fang, Y. Chen, J. Ding, Z. Yu, T. Masquelier, D. Chen, L. Huang, H. Zhou, G. Li, and Y. Tian, "Spikingjelly: An open-source machine learning infrastructure platform for spike-based intelligence," *Science Advances*, vol. 9, no. 40, pp. eadi1480–1–18, 2023.

[9] H. Zheng, Z. Zheng, R. Hu, B. Xiao, Y. Wu, F. Yu, X. Liu, G. Li, and L. Deng, "Temporal dendritic heterogeneity incorporated with spiking neural networks for learning multi-timescale dynamics," *Nature Communications*, vol. 15, no. 1, pp. 277–297, 2024.

[10] C. Yu, Z. Gu, D. Li, G. Wang, A. Wang, and E. Li, "Stsc-snn: Spatio-temporal synaptic connection with temporal convolution and attention for spiking neural networks," *Frontiers in Neuroscience*, vol. 16, pp. 1 079 357–1–15, 2022.

[11] W. Fang, Z. Yu, Y. Chen, T. Huang, T. Masquelier, and Y. Tian, "Deep residual learning in spiking neural networks," in *Advances in Neural Information Processing Systems (NeurIPS)*, vol. 34, Dec. 2021, pp. 21 056–21 069.

[12] Y. Hu, L. Deng, Y. Wu, M. Yao, and G. Li, "Advancing spiking neural networks toward deep residual learning," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 2, pp. 1–15, 2024.

[13] H. Zheng, Y. Wu, L. Deng, Y. Hu, and G. Li, "Going deeper with directly-trained larger spiking neural networks," in *Proceedings of the AAAI conference on artificial intelligence(AAAI)*, Feb. 2021, pp. 11 062–11 070.

[14] Z. Zhou, Y. Zhu, C. He, Y. Wang, S. YAN, Y. Tian, and L. Yuan, "Spikformer: When spiking neural network meets transformer," in *The Eleventh International Conference on Learning Representations(ICLR)*, May. 2023.

[15] M. Yao, J. Hu, Z. Zhou, L. Yuan, Y. Tian, B. Xu, and G. Li, "Spike-driven transformer," in *Advances in Neural Information Processing Systems(NeruIPS)*, vol. 36, Dec. 2023, pp. 64 043–64 058.

[16] M. Yao, J. Hu, T. Hu, Y. Xu, Z. Zhou, Y. Tian, B. XU, and G. Li, "Spike-driven transformer v2: Meta spiking neural network architecture inspiring the design of next-generation neuromorphic chips," in *The Twelfth International Conference on Learning Representations(ICLR)*, May. 2024.

[17] D. Ren, Z. Ma, Y. Chen, W. Peng, X. Liu, Y. Zhang, and Y. Guo, "Spiking pointnet: Spiking neural networks for point clouds," in *Advances in Neural Information Processing Systems(NeurIPS)*, vol. 36, Dec. 2023, pp. 41 797–41 808.

[18] X. Luo, M. Yao, Y. Chou, B. Xu, and G. Li, "Integer-valued training and spike-driven inference spiking neural network for high-performance and energy-efficient object detection," in *European Conference on Computer Vision (ECCV)*, Sep. 2024, pp. 253–272.

[19] H. Lin, Y. Zhang, S. Niu, S. Cui, and Z. Li, "Monotta: Fully test-time adaptation for monocular 3d object detection," in *European Conference on Computer Vision (ECCV)*, 2025, pp. 96–114.

[20] X. Chen, K. Kundu, Z. Zhang, H. Ma, S. Fidler, and R. Urtasun, "Monocular 3d object detection for autonomous driving," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 2147–2156.

[21] L. Yang, X. Zhang, J. Yu, J. Li, T. Zhao, L. Wang, Y. Huang, C. Zhang, H. Wang, and Y. Li, "Monogae: Roadside monocular 3d object detection with ground-aware embeddings," *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 11, pp. 17 587–17 601, 2024.

[22] H. Gao, X. Yu, Y. Xu, J. Y. Kim, and Y. Wang, "Monoli: Precise monocular 3-d object detection for next-generation consumer electronics for autonomous electric vehicles," *IEEE Transactions on Consumer Electronics*, vol. 70, no. 1, pp. 3475–3486, 2024.

[23] J. Jinrang, Z. Li, and Y. Shi, "Monouni: A unified vehicle and infrastructure-side monocular 3d object detection network with sufficient depth clues," in *Advances in Neural Information Processing Systems(NeurIPS)*, vol. 36, Dec. 2023, pp. 11 703–11 715.

[24] Y. Oh, H.-I. Kim, S. T. Kim, and J. U. Kim, "Monowad: Weather-adaptive diffusion model for robust monocular 3d object detection," in *European Conference on Computer Vision (ECCV)*, Sep. 2024, pp. 326–345.

[25] S. Kim, Y. Kim, S. Hwang, H. Jeong, and D. Kum, "Labeldistill: Label-guided cross-modal knowledge distillation for camera-based 3d object detection," in *European Conference on Computer Vision (ECCV)*, Sep. 2024, pp. 19–37.

[26] Y. Zhan, Y. Yuan, and Z. Xiong, "Mono3dvg: 3d visual grounding in monocular images," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 38, no. 7, Feb. 2024, pp. 6988–6996.

[27] L. Yan, P. Yan, S. Xiong, X. Xiang, and Y. Tan, "Monocd: Monocular 3d object detection with complementary depths," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2024, pp. 10 248–10 257.

[28] J. Lee, C. Jung, J. Kim, and H. Cha, "Panopticus: Omnidirectional 3d object detection on resource-constrained edge devices," in *Proceedings of the 30th Annual International Conference on Mobile Computing and Networking(ACM)*, 2024, p. 1207–1221.

[29] Y. Ma, H. Wang, H. Shen, S. Duan, and S. Wen, "Analog spiking u-net integrating cbam&vit for medical image segmentation," *Neural Networks*, vol. 181, p. 106765, 2025.

[30] H. Shen, Q. Zheng, H. Wang, and G. Pan, "Rethinking the membrane dynamics and optimization objectives of spiking neural networks," in *Advances in Neural Information Processing Systems(NeurIPS)*, vol. 37, Dec. 2024, pp. 92 697–92 720.

[31] T. Bu, W. Fang, J. Ding, P. DAI, Z. Yu, and T. Huang, "Optimal ANN-SNN conversion for high-accuracy and ultra-low-latency spiking neural networks," in *International Conference on Learning Representations (ICLR)*, Apr. 2022, pp. 1–19.

[32] Z. Li, J. Jia, and Y. Shi, "Monolss: Learnable sample selection for monocular 3d detection," in *2024 International Conference on 3D Vision (3DV)*, Mar. 2024, pp. 1125–1135.

[33] A. Kumar, G. Brazil, and X. Liu, "Groomed-nms: Grouped mathematically differentiable nms for monocular 3d object detection," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2021, pp. 8973–8983.

[34] Z. Liu, Z. Wu, and R. Toth, "Smoke: Single-stage monocular 3d object detection via keypoint estimation," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2020, pp. 996–997.

[35] F. Pu, Y. Wang, J. Deng, and W. Yang, "Monodgp: Monocular 3d object detection with decoupled-query and geometry-error priors," *arXiv preprint arXiv:2410.19590*, 2024.

[36] Z. Chong, X. Ma, H. Zhang, Y. Yue, H. Li, Z. Wang, and W. Ouyang, "Monodistill: Learning spatial features for monocular 3d object detection," *arXiv preprint arXiv:2201.10830*, 2022.

[37] Z. Qin, J. Wang, and Y. Lu, "Monogrnet: A general framework for monocular 3d object detection," *IEEE transactions on pattern analysis and machine intelligence*, vol. 44, no. 9, pp. 5170–5184, 2021.

[38] K. Duan, S. Bai, L. Xie, H. Qi, Q. Huang, and Q. Tian, "Centernet: Keypoint triplets for object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 6569–6578.

[39] R. Van Rullen and S. J. Thorpe, "Rate coding versus temporal order coding: what the retinal ganglion cells tell the visual cortex," *Neural computation*, vol. 13, no. 6, pp. 1255–1283, 2001.

[40] J. Kim, H. Kim, S. Huh, J. Lee, and K. Choi, "Deep neural networks with weighted spikes," *Neurocomputing*, vol. 311, pp. 373–386, 2018.

[41] S. Zhou, X. Li, Y. Chen, S. T. Chandrasekaran, and A. Sanyal, "Temporal-coded deep spiking neural network with easy training and robust performance," in *Proceedings of the AAAI conference on artificial intelligence(AAAI)*, vol. 35, no. 12, May. 2021, pp. 11 143–11 151.

[42] I. M. Comsa, K. Potempa, L. Versari, T. Fischbacher, A. Gesmundo, and J. Alakuijala, "Temporal coding in spiking neural networks with alpha synaptic function," in *ICASSP 2020 - 2020 IEEE International*

Conference on Acoustics, Speech and Signal Processing (ICASSP), May. 2020, pp. 8529–8533.

[43] S. Park, S. Kim, B. Na, and S. Yoon, "T2fsnn: deep spiking neural networks with time-to-first-spike coding," in *2020 57th ACM/IEEE Design Automation Conference (DAC)*, Jul. 2020, pp. 1–6.

[44] X. Qiu, R.-J. Zhu, Y. Chou, Z. Wang, L.-j. Deng, and G. Li, "Gated attention coding for training high-performance and efficient spiking neural networks," in *Proceedings of the AAAI Conference on Artificial Intelligence(AAAI)*, vol. 38, no. 1, Feb. 2024, pp. 601–610.

[45] D. Windhager, B. A. Moser, and M. Lunglmayr, "Snn architecture for differential time encoding using decoupled processing time," in *2024 IEEE 6th International Conference on AI Circuits and Systems (AICAS)*, Apr. 2024, pp. 26–30.

[46] Q. Yang, M. Zhang, J. Wu, K. C. Tan, and H. Li, "Lc-ttfs: Toward lossless network conversion for spiking neural networks with ttfs coding," *IEEE Transactions on Cognitive and Developmental Systems*, vol. 16, no. 5, pp. 1626–1639, 2024.

[47] X. Chen, Q. Yang, J. Wu, H. Li, and K. C. Tan, "A hybrid neural coding approach for pattern recognition with spiking neural networks," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 46, no. 5, pp. 3064–3078, 2024.

[48] S. Chen, B. Liu, and H. You, "Brain-inspired efficient pruning: Exploiting criticality in spiking neural networks," *arXiv preprint arXiv:2311.16141*, 2023.

[49] B. Chakraborty, B. Kang, H. Kumar, and S. Mukhopadhyay, "Sparse spiking neural network: Exploiting heterogeneity in timescales for pruning recurrent SNN," in *International Conference on Learning Representations (ICLR)*, May. 2024, pp. 1–32.

[50] M. Liu, J. Gan, R. Wen, T. Li, Y. Chen, and H. Chen, "Spiking-diffusion: Vector quantized discrete diffusion model with spiking neural networks," *arXiv preprint arXiv:2308.10187*, 2023.

[51] L. Zuo, Y. Ding, M. Jing, K. Yang, and Y. Yu, "Self-distillation learning based on temporal-spatial consistency for spiking neural networks," *arXiv preprint arXiv:2406.07862*, 2024.

[52] A. G. Howard, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv preprint arXiv:1704.04861*, 2017.

[53] F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jul. 2017, pp. 1251–1258.

[54] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, and H. Adam, "Encoder-decoder with atrous separable convolution for semantic image segmentation," in *Proceedings of the European conference on computer vision (ECCV)*, Sep. 2018, pp. 801–818.

[55] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2015, pp. 815–823.

[56] M. Tan and Q. Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," in *International conference on machine learning (ICML)*, Jun. 2019, pp. 6105–6114.

[57] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Jun. 2016, pp. 770–778.

[58] S. Robertson, "On smoothing average precision," in *European Conference on Information Retrieval(ECIR)*, Dec. 2012, pp. 158–169.

[59] A. Simonelli, S. R. Bulo, L. Porzi, M. López-Antequera, and P. Kontschieder, "Disentangling monocular 3d object detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, Oct. 2019, pp. 1991–1999.