SpikingIR: A Novel Converted Spiking Neural Network for Efficient Image Restoration

Yang Ouyang[1], Zihan Cheng[1], Xiaotong Luo[2], Guoqi Li[3], Yanyun Qu[1]*

[1]Key Laboratory of Multimedia Trusted Perception and Efficient Computing. Ministry of Education of China, School of Informatics, Xiamen University, Fujian, China

[2]The Hong Kong Polytechnic University, Hong Kong, China

[3]Institute of Automation, Chinese Academy of Sciences, Beijing, China

ouyangyang@stu.xmu.edu.cn, yyqu@xmu.edu.cn

Abstract

Image restoration has made great progress with the rise of deep learning, but its energy consumption limits its real-world applications. Spiking Neural Networks (SNNs) are seen as energy-efficient alternatives to Artificial Neural Networks (ANNs). Applying SNNs to image restoration (IR) remains challenging, primarily due to the limited information capacity of spike-based signals. This limitation leads to quantization errors and information loss, while IR tasks are highly sensitive to output precision and error. Thus, the restoration performance suffers significantly. To address this challenge, we propose SpikingIR, an ANN-to-SNN conversion framework for IR that reduces information loss and quantization error. SpikingIR mainly consists of two components: Convolutional Pixel Mapping (CPM) and Membrane Potential Reuse Neuron (MPRN), which are designed to alleviate quantization errors and information loss in the output and intermediate layers, respectively. Specifically, CPM maps discrete outputs into a continuous space, better aligning with pixel-level details. From the perspective of information entropy, we show that outputs of CPM contain more information than the original outputs. MPRN introduces a post-processing step with relaxed firing conditions to extract residual membrane potential, reducing information waste. Furthermore, we fine-tune the converted model to jointly optimize both accuracy and energy efficiency. Experimental results demonstrate that SpikingIR achieves performance comparable to ANN counterparts across various IR benchmarks while reducing energy consumption by up to 50%.

Introduction

Image restoration is a well-established low-level computer vision task which focuses on remove the diverse degradation of an image, e.g. super resolution, image dehazing, etc., and has made great progress with the rising of deep learning (Cai et al. 2016; Li et al. 2017; Liang et al. 2021; Qu et al. 2019; Hong and Lee 2024; Liu et al. 2024). It is observed that deeper networks usually achieve better performance. However, the large energy consumption of the deeper neural network makes it hard to implement in edge environments such as mobile phones or unmanned drones. It is increasing in demand to make the efficient energy model for image restoration.

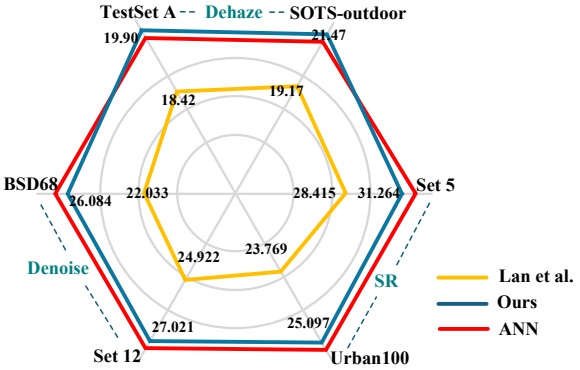

Figure 1: Performance comparison on different image restoration tasks and datasets.

Inspired by the success of Spiking Neural Networks (SNNs) in high-level computer vision such as image classification and segmentation (Shi, Hao, and Yu 2024; Yao et al. 2024b; Guo et al. 2023; Deng et al. 2022; Liu et al. 2022), we investigate spiking image restoration.

However, there are few discuss about the SNNs application to image restoration, which is an cutting-edge vision task. Different from SNNs on image classification with discrete predictions, SNNs need output continous high-precision regression value for image restoration. It causes the quantization errors and information loss to make the discrete SNNs to simulate the continous output. Image restoration is highly sensitive to errors, and even minute inaccuracies can lead to severe degradation. Applying the commonly used rate encoding to images using integrate-and-fire (IF) neurons (Liu and Wang 2004) results in severe image degradation. To address this problem, longer simulation timesteps are often used to improve accuracy, but it leads to increased latency and energy consumption, limiting the practicality of SNNs in image restoration.

Given the stringent accuracy requirements of image restoration, reducing information loss and quantization error accumulation in SNNs are two critical challenges. To address these, we propose a novel ANN-to-SNN conversion framework, **SpikingIR**, which optimizes both the output and intermediate layers of SNNs to enhance their performance

*Corresponding author.

on image restoration tasks.

SpikingIR contains two important components: **Convolutional Pixel Mapper (CPM)** and **Mambrane Potential Reuse Neuron(MPRN)**. CPM aims to mitigate information loss and quantization error in the output layer of SNNs. Draw lessons from application of SNNs to image classification, we use full-precision output feature which is recognized to significantly improve SNNs classification performance (Guo et al. 2024b). CPM employs convolutional operations to map the discrete raw spike outputs of SNNs into full-precision outputs. In addition, we introduce a Residual Membrane Potential Information Shortcut (RMPIS) to preserve the residual information from the original IF output layer. From the perspective of information entropy, CPM better preserves information.

MPRN is designed to mitigate information loss in the intermediate layers of SNNs. It is observed that a considerable amount of membrane potential remains in neurons during inference. This residual potential contains valuable information that is usually discarded, leading to information loss. To fully exploit this information, MPRN introduces an additional post-processing step after the final time step, allowing the residual membrane potential to discharge and release the contained information.

Finally, we apply surrogate gradient-based fine-tuning to further optimize the SpikingIR model. As shown in Fig. 1, our method achieves performance close to that of ANNs on image super-resolution on Set5, Urban100, image dehazing on SOTS-outdoor, TestSetA, and image denoising on Set12, BSD68. Fig. 2 shows that a output image by the spiking neural network with IF neuron activation suffers severe degradation.

In summary, our contributions include:

- We propose **SpikingIR**, an ANN-to-SNN conversion framework tailored for image restoration. It achieves competitive performance within only 10 timesteps, significantly narrowing the performance gap between SNNs and ANNs in this domain.
- We design two simple and effective modules, **CPM** and **MPRN**, to effectively reduce information loss and quantization errors in SNNs. We also provide theoretical analysis to demonstrate their effectiveness;
- Extensive experiments on multiple image restoration tasks demonstrate that our method can achieves superior performance with significantly fewer timesteps, while reducing energy consumption by approximately 50% compared to the ANN baseline.

Related Work

SNN Training Methods. Due to the non-differentiability of spikes, we cannot simply use backpropagation to train SNNs like ANNs. The two most commonly used methods for training SNNs are direct training using surrogate gradients (DT) (Neftci, Mostafa, and Zenke 2019; Wu et al. 2018) and ANN-to-SNN conversion (A2S) (Cao, Chen, and Khosla 2015; Diehl et al. 2015; Rueckauer et al. 2017). For DT, the gradient of a surrogate function replaces that of the non-differentiable firing function during backpropagation. In this way, we can train SNNs similarly to ANNs. However, this approach still exhibits a significant gap compared to ANNs on complex tasks and large-scale datasets (Lan et al. 2023; Li and Zeng 2022). The A2S method involves transferring a pre-trained ANN model to SNN, allowing for excellent results without the need for additional training. To achieve better conversion results, the A2S method typically requires more time steps compared to DT (Luo et al. 2024; Guo et al. 2024a), meaning it needs more inference time and energy. In recent years, researchers have also proposed a hybrid training method (Rathi et al. 2019; Rathi and Roy 2021), where A2S is used as an initialization method for SNNs, followed by further optimization DT methods.

Figure 2: The image undergoes severe degradation after passing through IF neurons due to information loss and quantization error.

Image Restoration Based on SNN. SNNs have achieved great success in the field of image classification, and there has been research on their application in tasks such as object detection (Kim et al. 2020; Luo et al. 2024), recognition (Amir et al. 2017; Lan et al. 2023), and segmentation (Patel et al. 2021; Lei et al. 2025). However, there is still limited research on their use in the field of image restoration. (Song et al. 2024) introduced SNNs to the task of image deraining and achieved superior results by designing spiking residual blocks and incorporating a mixed attention mechanism. However, they introduce a large amount of floating-point computation to compensate for the information loss in SNNs, which goes against the nature of SNNs that transmit information via spikes. This leads to increased energy consumption and makes it difficult to deploy on neuromorphic hardware. (Castagnetti, Pegatoquet, and Miramond 2023) applied SNNs to the image denoising task. They employed learnable neuron models to reduce quantization errors, but did not address the issue of information loss, leading to suboptimal results.

Preliminary

Spiking Neuron Model. In SNNs, information is encoded as spike sequences via spiking neurons. The Integrate-and-Fire (IF) neuron is widely used in ANN-to-SNN conversion (Deng and Gu 2021; Bu et al. 2022; Bu, Li, and Yu 2025), and its behavior is described by the following equations:

$$v^l(t^-) = v^l(t-1) + W^l s^{l-1}(t), \tag{1}$$

$$s^l(t) = \begin{cases} 1, & \text{if } v^{l^-}(t) \geq V_{th}^l \\ 0, & \text{if } v^l(t^-) < V_{th}^l \end{cases}, \tag{2}$$

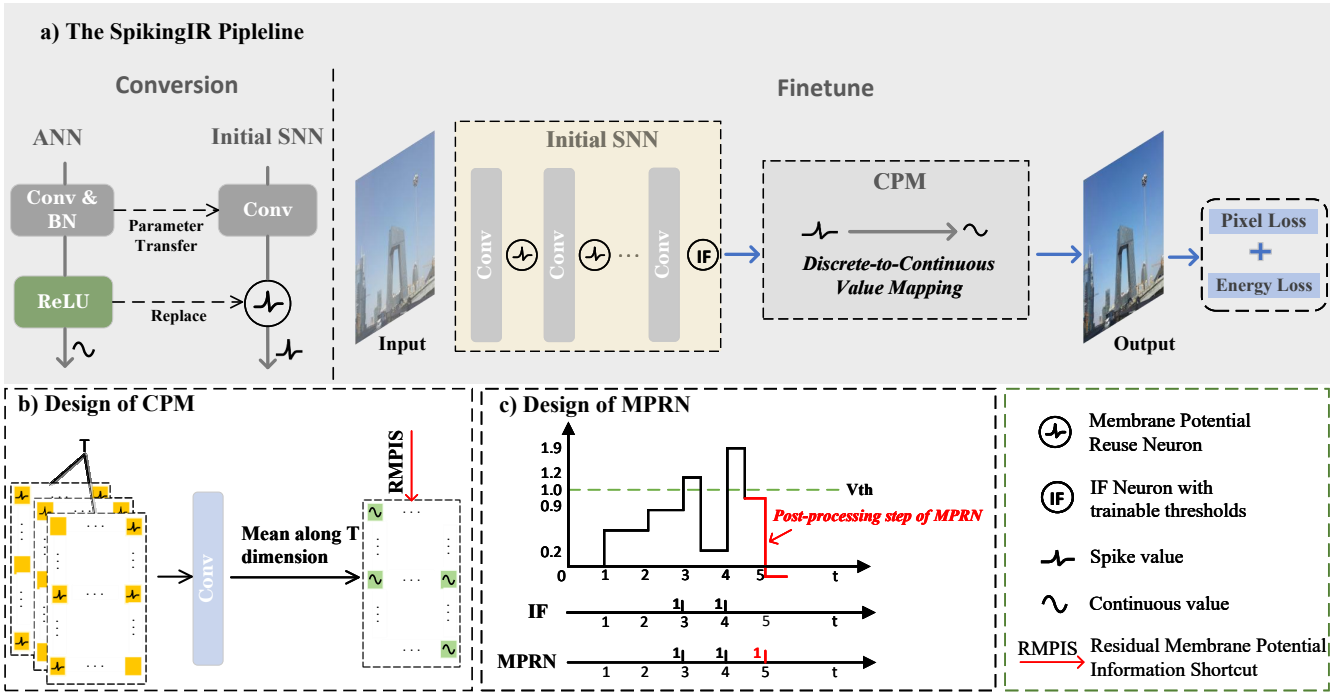

Figure 3: **a) The proposed SpikingIR.** On the left is the conversion process from ANN to SNN, where parameter transfer occurs and ReLU is replaced by spiking neurons. On the right is the fine-tuning process, where the CPM is added to the output layer after the initial SNN, followed by backpropagation training using the surrogate gradient method to optimize model performance under energy constraints. **b) CPM design.** CPM consists of a convolutional layer and a Residual Potential Membrane Information Shortcut (RMPIS). The convolution maps discrete spikes to the continuous space, while RMPIS preserves the residual information of the final IF neuron. **c) MPRN design.** MPRN adds one post-processing step after the last timestep, allowing neurons with membrane potential exceeding αV_{th} to continue firing and better utilize residual potential. Shown is a comparison between IF and MPRN under $\alpha = 0.9$, $T = 4$, $V_{th} = 1$.

$$v^l(t) = v^l(t^-) - s^l(t) \cdot V_{th}^l. \tag{3}$$

W^l denotes the weights between layer $l-1$ and layer l ($l \in \{0, 1, ..., L\}$), and $s^l(t)$ describes whether the neuron in layer l fires at time step t. The neuron fires if the membrane potential before firing $v^l(t^-)$ exceeds the threshold V_{th}^l. To reduce information loss, we adopt the "soft reset" strategy(Han, Srinivasan, and Roy 2020) described in Eq. (3) to update the membrane potential $v^l(t)$.

ANN-to-SNN conversion. The basic principle of the conversion is to use the firing rate of spiking neurons to approximate the ReLU activation values of an ANN. The firing rate of neurons in layer l is calculated as:

$$r^l = \frac{1}{T} \sum_{t=1}^{T} s^l(t) \tag{4}$$

Here T denotes the number of timesteps, as the model is required to run for T steps. The final output is denoted as r^L (Lan et al. 2023). Based on Eq. (1 - 4), we further obtain:

$$r^l = W^l r^{l-1} - \frac{v^l(T)}{T V_{th}^l} \tag{5}$$

In ANNs with ReLU activation, the relationship between activations in adjacent layers is given by:

$$a^l = \mathrm{ReLU}(W^l a^{l-1}) \tag{6}$$

The mismatch between Eq. (5) and Eq. (6) introduces conversion error, denoted as $\left|a^l - r^l\right|$. As $T \to \infty$, the conversion error approaches zero. (Rueckauer et al. 2017). This is why ANN-to-SNN method typically require a large number of timesteps.

Method

As shown in Fig. 3, SpikingIR is a two-stage conversion framework. In the conversion stage, we first adopt the method in (Rueckauer et al. 2017) to convert a pre-trained ANN into an SNN. The IF neurons in the first $L-1$ layers are then replaced with MPRN, while the neurons in the last layer remain IF, yielding the initial SNN. Subsequently, in the fine-tuning stage, we append CPM after the output layer of the initial SNN to map discrete spikes into a continuous space, and then fine-tune the entire network with backpropagation. This section introduces the CPM and MPRN modules, followed by the fine-tuning process.

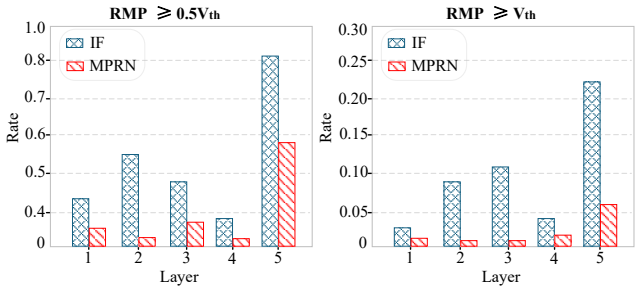

Figure 4: The distribution of Residual Membrane Potential (RMP) after inference in AOD-Net. Left: proportion of RMP $\geq 0.5V_{th}$; Right: proportion of RMP $\geq V_{th}$.

Convolutional Pixel Mapper (CPM)

Output Enhancement by Convolution. We consider the output layer of the initial SNN. The network output can be expressed as $O_{vanilla} = r^L$. Since each neuron can fire at most once at each time step, we have $s \in \{0, \ldots, T\}$, and thus the output of the vanilla SNN, $O_{vanilla}$ can take only $T + 1$ discrete values. Previous studies (Guo et al. 2024b) have demonstrated that using full-precision output features significantly improves the classification accuracy of SNNs. We argue that this observation holds for image restoration, where output precision directly impacts quality. Thus, we add a convolutional layer (denoted as Conv) after the IF output as an enhancer, mapping discrete spikes to a continuous representation:

$$F_{enh} = \frac{1}{T}\sum_{t=1}^{T} Conv(s^L(t)) \tag{7}$$

Here, F_{enh} is the result of applying convolutional enhancement to the original output r^L. The added Conv not only maps the discrete output to a continuous space to better approximate continuous pixel values, but also serves the purpose of error induction, which corrects the quantization errors accumulated in earlier layers through training. To avoid the Conv interfering with the initial SNN output, we initialize it as an identity mapping, leaving the input unchanged to enhance subsequent training speed.

Residual Membrane Potential Information Shortcut (RMPIS). Due to the thresholding mechanism of spiking neurons, the membrane potential only releases information when it exceeds the threshold. As a result, after inference ends, a significant amount of membrane potential remains in the neurons, as shown in Fig. 4. This residual membrane potential is typically discarded directly, causing information loss. To address this, we derive the equivalent firing rate corresponding to the residual membrane potential to better understand the information it carries:

$$I = \frac{v_{res}}{TV_{th}} \tag{8}$$

Here, I denotes the equivalent firing rate, i.e., the information quantity, and v_{res} is the residual membrane potential. The detailed derivation is provided in the Appendix. To make use of the residual information from the original SNN IF output layer, we propose the RMPIS, which adds this information directly to the final output. With the CPM module applied, the final output is given by:

$$O_{CPM} = F_{enh} + I = \frac{1}{T}\sum_{t=1}^{T} Conv(s^L(t)) + \frac{v_{res}}{TV_{th}} \tag{9}$$

Analysis of CPM. We analyze the advantage of CPM from the perspective of information entropy. Given a set $\mathbf{S}$, its information capacity $\mathcal{C}(\mathbf{S})$ can be measured by its maximum entropy (Guo et al. 2025), defined as:

$$\mathcal{C}(\mathbf{S}) = max\mathcal{H}(\mathbf{S}) = max(-\sum_{s\in\mathbf{S}} p_{\mathbf{S}}(s) log p_{\mathbf{S}}(s)) \tag{10}$$

where $p_{\mathbf{S}}(s)$ is the probability of sample s appearing in $\mathbf{S}$. When all samples have equal probability, i.e., $p_{\mathbf{S}}(s) = \frac{1}{N}$, the entropy $\mathcal{H}(\mathbf{S})$ attains its maximum value log(N). Therefore, $C(S) = \log(N)$. For the original SNN output, its possible values amount to only $(T+1)^{CHW}$, thus the information capacity is:

$$\mathcal{C}(O_{vanilla}) = log((T+1)^{CHW}) = log(T+1) \cdot CHW \tag{11}$$

For the 32-bit floating-point output processed by CPM, the number of possible values is $2^{32\cdot CHW}$, thus the information capacity is:

$$\mathcal{C}(O_{CPM}) = log(2^{32\cdot CHW}) = 32 \cdot CHW \tag{12}$$

Clearly, when the number of time steps T is small, $\mathcal{C}(O_{CPM}) \gg \mathcal{C}(O_{vanilla})$, indicating that CPM conveys significantly more information than the vanilla output, thereby effectively mitigating information loss. Besides, the extra energy cost from CPM is negligible because the added convolutional layers are spike-driven, and RMPIS involves few floating-point operations. Details of the extra energy consumption can be found in the Appendix.

Membrane Potential Reuse Neuron (MPRN)

While CPM addresses information loss and quantization error in the output layer, we propose MPRN to tackle the information loss caused by residual membrane potential in the intermediate layers. The core idea of MPRN is to add a post-processing time step after the final time step, enabling neurons to fire and release the residual membrane potential. As shown in Fig. 4, most residual potentials are below V_{th}^l, so we first lower the threshold to ease firing by:

$$V_{th}^l = \alpha^l V_{th}^l, \alpha \in (0, 1) \tag{13}$$

Moreover, to avoid excessive firing and the accumulation of negative membrane potential, we add a constraint to prevent firing from small residual potentials:

$$\left|\alpha^l V_{th}^l - V_{th}^l\right| < \alpha^l V_{th}^l \tag{14}$$

This yields $\alpha \in (0.5, 1)$. Furthermore, we use the Sigmoid function to control the value of α for easier training during the fine-tuning phase:

$$\alpha^l = 0.5 \cdot Sigmoid(\beta^l) + 0.5 = 0.5 \cdot (\frac{1}{1 + e^{-\beta^l}}) + 0.5 \tag{15}$$

Here, β is a learnable parameter. Finally, as shown in Fig. 3 (C), the MPRN can be described during the post-processing time steps as:

$$s^l(T+1) = \begin{cases} 1, & \text{if } v^l(T+1) \geq \alpha^l V_{th}^l \\ 0, & otherwise \end{cases} \tag{16}$$

For the other time steps, the behavior is the same as that of the IF neuron. We further prove the following theorem:

Theorem 1. *The upper bound of the conversion error $|r^l - a^l|$ at each layer when using the MPRN model does not exceed the error upper bound when using the IF neuron.*

The detailed proof is provided in the Appendix. This indicates that MPRN can reduce the error upper bound and improve the conversion performance.

Fine-tuning

After inserting the CPM and MPRN modules, we fine-tune the CPM, MPRN, and the original parameterized modules. To prevent the model from prioritizing performance at the expense of energy consumption, we introduce an energy constraint:

$$\mathcal{L}_{energy} = \frac{1}{TLBCHW} \sum_{t,l,b,c,h,w} s^l(t) \tag{17}$$

where B denotes the batch size and C, H, W represents the feature-map dimensions. The total loss is defined as:

$$\mathcal{L}_{total} = \mathcal{L}_{pixel} + \lambda(e)\mathcal{L}_{energy} \tag{18}$$

where $\mathcal{L}_{pixel}$ denotes the task loss. To prevent the model from focusing too much on reducing energy consumption at the early stage of training at the expense of task performance, we adopt an epoch-wise increasing strategy for adjusting λ:

$$\lambda(e) = k \cdot \frac{1 - cos(\frac{\pi e}{E})}{2} \tag{19}$$

where e denotes the current epoch, E is the total number of epochs, and k is a hyperparameter that controls the maximum value of λ.

Experiments

We comprehensively evaluate our method on multiple datasets for image dehazing, denoising, and super-resolution. Considering that current SNN research and toolchains offer limited support for advanced modules such as Softmax, LayerNorm, and attention mechanisms, we adopt AOD-Net(Li et al. 2017), DnCNN(Zhang et al. 2017), and VDSR(Kim, Lee, and Lee 2016) as baseline models. These architectures are widely used, reliable in performance, and compatible with mainstream ANN-to-SNN conversion frameworks. Although not the latest designs, they provide a solid foundation for evaluating our method in the SNN domain. As shown in Fig. 1, SpikingIR achieves strong performance across all tasks.

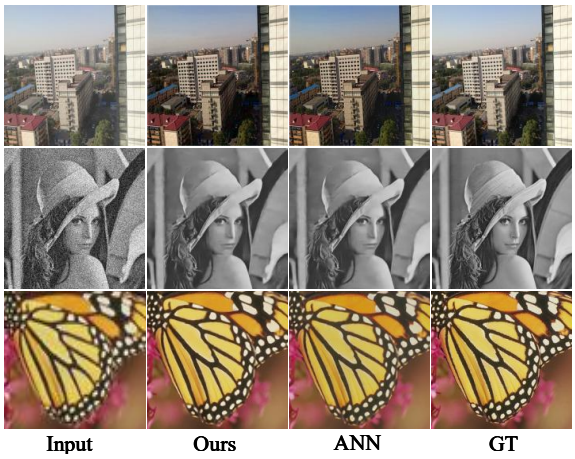

Figure 5: The visual results of the proposed method and ANNs, from top to bottom, correspond to the image dehazing, denoising, and 4× super-resolution tasks, respectively.

Experimental Setting

Datasets. For the image dehazing task, we adopt the same indoor synthetic dataset used in the AOD-Net for training, and TestSet A for testing, which contains 3,170 hazy indoor images. Additionally, we tested on 500 outdoor hazy images from the Synthetic Objective Testing Set (SOTS) dataset (Li et al. 2018). For the image denoising task, we used the same training and testing sets as DnCNN. The training set consists of 400 grayscale images, while the testing set includes Set12 and BSD68. We evaluated the model's denoising performance on additive Gaussian noise with standard deviations $\sigma \in [15, 25, 50]$, where σ denotes the noise level. For the super-resolution task, we use the same method as in VDSR to generate the training set and evaluate on Set5 and Urban100.

Implementation Setting. Similar to most direct training (DT) methods (Zhou et al. 2022; Song et al. 2024), we adopt a sigmoid-like surrogate gradient during fine-tuning. All experiments were conducted on four NVIDIA RTX A6000 GPUs using PyTorch 2.4.1. More detailed experimental settings are provided in the Appendix.

Evaluation Metrics. For all tasks, we compute PSNR and SSIM to adopt the image restoration performance of the models. In addition, we assess conversion efficiency by calculating the conversion rate, which is defined as the ratio between the SNN and ANN results (Lan et al. 2023). For energy consumption, we adopt the same methodology as in (Yao et al. 2024a), with detailed calculations provided in the Appendix.

Experiment Results

Effectiveness of SpikingIR on Image Dehazing. We compare the performance of 5 methods on the image dehazing task under varying timesteps. The compared ANN-to-SNN methods include Rueckauer et al. (Rueckauer et al. 2017), QCFS (Bu et al. 2022), Burst Spike (Li and Zeng

Method	Timestep↓	TestSet A			SOTS-outdoor			Energy (mJ)↓
		PSNR↑	SSIM↑	Rate(%)↑	PSNR↑	SSIM↑	Rate(%)↑	
ANN (AOD-Net)	–	19.53	0.8165	–	20.12	0.9000	–	2.289
Rueckauer et al.	20	18.42	0.7252	91.6	19.20	0.8066	92.5	3.572
	50	19.45	0.7964	98.6	20.10	0.8813	98.9	9.228
QCFS	5	16.69	0.5926	79.0	18.04	0.6597	81.5	–
	8	18.17	0.6503	86.3	20.26	0.7455	93.2	
	10	17.54	0.6663	85.7	19.63	0.7721	91.7	
Burst Spike	20	18.41	0.7253	91.5	19.18	0.8066	92.5	–
	50	19.45	0.7963	98.6	20.10	0.8811	98.9	
Lan et al.	20	18.42	0.7252	91.6	19.17	0.8066	92.5	–
	50	19.45	0.7964	98.6	20.10	0.8811	98.9	
Ternary Spike	15	15.02	0.7201	82.6	15.80	0.8115	84.3	0.567
SpikingIR (Ours)	5	19.63	0.7744	97.7	21.31	0.8752	**100.0**	0.931
	8	19.76	0.7811	98.5	21.39	0.8837	**100.0**	1.674
	10	**19.90**	**0.7910**	**99.4**	**21.47**	**0.8914**	**100.0**	1.863

Table 1: The baseline ANN dehazing model uses AOD-Net, with results on the TestSet A and SOTS-outdoor datasets. Rate(%) represents the average of the PSNR and SSIM conversion rate relative to the ANN.

σ	Method	Timestep↓	Set12			BSD68			Energy (mJ)↓
			PSNR↑	SSIM↑	Rate(%)↑	PSNR↑	SSIM↑	Rate(%)↑	
15	ANN (DnCNN)	–	32.859	0.9044	–	31.730	0.8930	–	384.704
	DnSNN	15	32.681	–	99.5	31.593	–	99.6	–
	SpikingIR (Ours)	10	32.631	0.8962	99.3	31.544	0.8853	99.4	190.765
		15	**32.758**	**0.9004**	**99.7**	**31.633**	**0.8891**	**99.7**	259.861
25	ANN (DnCNN)	–	30.436	0.8604	–	29.230	0.8262	–	384.704
	DnSNN	15	30.186	–	99.2	29.025	–	99.3	–
	SpikingIR (Ours)	10	30.160	0.8439	99.1	29.009	0.8108	99.2	197.196
		15	**30.285**	**0.8510**	**99.5**	**29.072**	**0.8176**	**99.5**	272.869
50	ANN (DnCNN)	–	27.178	0.7636	–	26.230	0.7097	–	384.704
	DnSNN	15	26.790	–	98.6	25.940	–	98.9	–
	SpikingIR (Ours)	10	26.861	0.7422	98.8	25.955	0.6812	99.0	212.176
		15	**27.021**	**0.7597**	**99.4**	**26.084**	**0.7011**	**99.4**	294.478

Table 2: Quantitative results on Set12 and BSD68 for image denoising under different noise levels.

2022), and Lan et al. (Lan et al. 2023), as well as the DT method Ternary Spike (Guo et al. 2024a). The results are shown in Table 1. It is observed that, SpikingIR achieves excellent performance on both the TestSetA and SOTS-outdoor datasets using only 5 timesteps, and enables near-lossless conversion within 10 timesteps. Compared to the ANN, our method reduces energy consumption by more than 50% when Timestep = 5, and by approximately 20% when Timestep = 10. In addition, the proposed CPM not only maps discrete outputs into a continuous space, but also significantly enhances the model's representation and fitting capability, allowing the converted SNN to even surpass the original ANN in terms of PSNR. Compared to other ANN-to-SNN methods, SpikingIR shows substantial advantages in performance, time steps, and energy efficiency, as other methods typically require more timesteps to mitigate quantization error and information loss—issues that SpikingIR effectively addresses through CPM and MPRN. When compared to the DT-based method, although it offers superior energy efficiency, its performance on image restoration tasks is noticeably inferior due to the inherent complexity of the task and the limitations of current surrogate gradient training techniques.

Effectiveness of SpikingIR on Image Denoising. We conducted image denoising experiments comparing our method with the state-of-the-art SNN-based model DnSNN (Castagnetti, Pegatoquet, and Miramond 2023), as shown in Table 2. Since DnSNN does not report SSIM, for comparison purposes, we calculate only the PSNR conversion rate. Our method consistently outperforms DnSNN across all noise levels, especially at high noise ($\sigma = 50$), demonstrating strong robustness. Even with only 10 time steps, our method maintains a conversion rate above 99%. Compared to the ANN, it reduces energy consumption by nearly 50%

Scale	Method	Timestep↓	Set5			Urban100			Energy (mJ)↓
			PSNR↑	SSIM↑	Rate↑	PSNR↑	SSIM↑	Rate↑	
×2	ANN (VDSR)	–	37.653	0.9647	–	30.833	0.9220	–	347.015
	Lan et al.	20	33.971	0.9375	93.7	27.100	0.8506	90.1	–
	SpikingIR (Ours)	10	37.408	**0.9633**	99.6	30.610	0.9194	99.5	59.582
		15	**37.459**	0.9626	**99.6**	**30.669**	**0.9194**	**99.6**	117.411
×4	ANN (VDSR)	–	31.465	0.8952	–	25.238	0.7708	–	341.360
	Lan et al.	20	28.722	0.8232	91.6	23.890	0.6948	92.4	–
	SpikingIR (Ours)	10	31.264	**0.8915**	99.5	25.097	0.7645	99.3	52.242
		15	**31.314**	0.8910	**99.5**	**25.134**	**0.7649**	**99.4**	108.051

Table 3: Quantitative comparison at different super-resolution scales on the Set5 and Urban100 datasets.

when Timestep = 10 and around 30% when Timestep = 15. Notably, our method does not surpass the ANN in PSNR on denoising, unlike in dehazing, due to DnCNN's deeper structure causing more conversion error accumulation; the CPM's extra convolution layers cannot fully compensate for this.

Effectiveness of SpikingIR on Image Super-Resolution. Table 3 presents the performance of 2 methods in 2× and 4× super-resolution tasks. It can be seen that our method achieves the best results at both upscaling factors. Under various conditions, our method consistently maintains a conversion rate above 99.3%. Compared with ANN, our method reduces energy consumption by approximately 80% when Timestep = 10, and achieves a reduction of around 70% when Timestep = 15. A point worth noting is that 4× super-resolution requires cropping more border pixels from the test inputs to satisfy the divisibility constraints; as a result, its energy efficiency is slightly lower than that of 2× super-resolution. Similar to the results on the denoising task, our method does not surpass ANN in terms of PSNR. This is primarily due to the accumulated error caused by the deep structure of VDSR.

Visual Comparison. Figure 5 presents visual comparison between our proposed SpikingIR and the ANN baseline. As shown, SpikingIR achieves a reconstruction quality that is nearly indistinguishable from that of ANN, demonstrating the effectiveness of the conversion process. Additional visual comparisons with more methods are provided in the appendix to further validate the superior performance of SpikingIR.

Ablation Study

We conduct ablation experiments on the TestSet dataset to investigate the contributions of key components in SpikingIR, including the convolutional layer in CPM (Conv), RMPIS, MPRN and Fine-tuning (FT). The experiments are conducted with a timestep of 10, and the results are shown in Table. 4.

Effectiveness of the FT. Compared the first and second rows in Table 4, it is observed that the variant with FT gain 0.53dB and 0.0125 in PSNR and SSIM, against the baseline. It shows the effectiveness of FT.

Conv	RMPIS	MPRN	FT	PSNR↑	SSIM↑
✗	✗	✗	✗	17.73	0.6378
✗	✗	✗	✓	18.26	0.6503
✓	✗	✗	✓	19.27	0.7272
✓	✓	✗	✓	19.59	0.7734
✓	✓	✓	✓	**19.90**	**0.7910**

Table 4: Ablation Study of Different Techniques of our SpikingIR. Timestep is set to 10.

Effectiveness of the CPM. Comparing rows 2 through 4 in Table 4, it is observed that the introduction of Conv and RMPIS leads to PSNR improvements of 1.01dB and 0.32dB, and SSIM improvements of 0.0769 and 0.0462, respectively. Meanwhile, considering the current floating-point output features, we remove the IF output layer of the initial SNN and allow it to directly output floating-point features. We then compare the results with those obtained using CPM, yielding (PSNR/SSIM) of 19.23/0.7663 vs. 19.59/0.7734. This clearly demonstrates the effectiveness of the CPM module.

Effectiveness of the MPRN. As shown in Fig. 4, the proportion of neurons with residual membrane potential greater than $0.5V_{th}$ is significantly lower when using MPRN compared to using IF neurons. Besides, By comparing rows 4 and 5 in Table 4, the introduction of MPRN leads to PSNR and SSIM improvements of 0.31dB and 0.0176, respectively, indicating that MPRN effectively utilizes the residual membrane potential and reduces information loss.

Conclusion

We propose an efficient ANN-to-SNN conversion method aimed at enhancing the performance of SNNs on image restoration tasks. To mitigate quantization errors and information loss in both the output and intermediate layers, we introduce CPM and MPRN neurons, respectively. CPM effectively reduces quantization error and information loss by mapping discrete outputs to continuous values. MPRN significantly alleviates information loss by reusing the residual membrane potentials of intermediate neurons. Extensive experiments on various image restoration tasks demonstrate the practical value of our method.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China under Grant 62176224; in part by the Science and Technology on Sonar Laboratory under Grant 2024-JCJQ-LB-32/07.

References

Amir, A.; Taba, B.; Berg, D.; Melano, T.; McKinstry, J.; Di Nolfo, C.; Nayak, T.; Andreopoulos, A.; Garreau, G.; Mendoza, M.; et al. 2017. A low power, fully event-based gesture recognition system. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 7243–7252.

Bu, T.; Fang, W.; Ding, J.; DAI, P.; Yu, Z.; and Huang, T. 2022. Optimal ANN-SNN Conversion for High-accuracy and Ultra-low-latency Spiking Neural Networks. In *International Conference on Learning Representations*.

Bu, T.; Li, M.; and Yu, Z. 2025. Inference-Scale Complexity in ANN-SNN Conversion for High-Performance and Low-Power Applications. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 24387–24397.

Cai, B.; Xu, X.; Jia, K.; Qing, C.; and Tao, D. 2016. Dehazenet: An end-to-end system for single image haze removal. *IEEE transactions on image processing*, 25(11): 5187–5198.

Cao, Y.; Chen, Y.; and Khosla, D. 2015. Spiking deep convolutional neural networks for energy-efficient object recognition. *International Journal of Computer Vision*, 113: 54–66.

Castagnetti, A.; Pegatoquet, A.; and Miramond, B. 2023. SPIDEN: deep Spiking Neural Networks for efficient image denoising. *Frontiers in Neuroscience*, 17: 1224457.

Deng, S.; and Gu, S. 2021. Optimal conversion of conventional artificial neural networks to spiking neural networks. *arXiv preprint arXiv:2103.00476*.

Deng, S.; Li, Y.; Zhang, S.; and Gu, S. 2022. Temporal Efficient Training of Spiking Neural Network via Gradient Reweighting. In *International Conference on Learning Representations*.

Diehl, P. U.; Neil, D.; Binas, J.; Cook, M.; Liu, S.-C.; and Pfeiffer, M. 2015. Fast-classifying, high-accuracy spiking deep networks through weight and threshold balancing. In *2015 International joint conference on neural networks (IJCNN)*, 1–8. ieee.

Guo, Y.; Chen, Y.; Liu, X.; Peng, W.; Zhang, Y.; Huang, X.; and Ma, Z. 2024a. Ternary spike: Learning ternary spikes for spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, 12244–12252.

Guo, Y.; Liu, X.; Chen, Y.; Peng, W.; Zhang, Y.; and Ma, Z. 2025. Spiking Transformer: Introducing Accurate Addition-Only Spiking Self-Attention for Transformer. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 24398–24408.

Guo, Y.; Liu, X.; Chen, Y.; Zhang, L.; Peng, W.; Zhang, Y.; Huang, X.; and Ma, Z. 2023. Rmp-loss: Regularizing membrane potential distribution for spiking neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 17391–17401.

Guo, Y.; Peng, W.; Liu, X.; Chen, Y.; Zhang, Y.; Tong, X.; Jie, Z.; and Ma, Z. 2024b. Enof-snn: Training accurate spiking neural networks via enhancing the output feature. *Advances in Neural Information Processing Systems*, 37: 51708–51726.

Han, B.; Srinivasan, G.; and Roy, K. 2020. Rmp-snn: Residual membrane potential neuron for enabling deeper high-accuracy and low-latency spiking neural network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 13558–13567.

Hong, C.; and Lee, K. M. 2024. AdaBM: On-the-Fly Adaptive Bit Mapping for Image Super-Resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2641–2650.

Kim, J.; Lee, J. K.; and Lee, K. M. 2016. Accurate image super-resolution using very deep convolutional networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1646–1654.

Kim, S.; Park, S.; Na, B.; and Yoon, S. 2020. Spiking-yolo: spiking neural network for energy-efficient object detection. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 11270–11277.

Lan, Y.; Zhang, Y.; Ma, X.; Qu, Y.; and Fu, Y. 2023. Efficient converted spiking neural network for 3d and 2d classification. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 9211–9220.

Lei, Z.; Yao, M.; Hu, J.; Luo, X.; Lu, Y.; Xu, B.; and Li, G. 2025. Spike2former: Efficient spiking transformer for high-performance image segmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, 1364–1372.

Li, B.; Peng, X.; Wang, Z.; Xu, J.; and Feng, D. 2017. Aod-net: All-in-one dehazing network. In *Proceedings of the IEEE international conference on computer vision*, 4770–4778.

Li, B.; Ren, W.; Fu, D.; Tao, D.; Feng, D.; Zeng, W.; and Wang, Z. 2018. Benchmarking single-image dehazing and beyond. *IEEE Transactions on Image Processing*, 28(1): 492–505.

Li, Y.; and Zeng, Y. 2022. Efficient and Accurate Conversion of Spiking Neural Network with Burst Spikes. In Raedt, L. D., ed., *Proceedings of the Thirty-First International Joint Conference on Artificial Intelligence, IJCAI-22*, 2485–2491. International Joint Conferences on Artificial Intelligence Organization. Main Track.

Liang, J.; Cao, J.; Sun, G.; Zhang, K.; Van Gool, L.; and Timofte, R. 2021. Swinir: Image restoration using swin transformer. In *Proceedings of the IEEE/CVF international conference on computer vision*, 1833–1844.

Liu, F.; Zhao, W.; Chen, Y.; Wang, Z.; and Jiang, L. 2022. Spikeconverter: An efficient conversion framework zipping the gap between artificial neural networks and spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, 1692–1701.

Liu, J.; Wang, Q.; Fan, H.; Wang, Y.; Tang, Y.; and Qu, L. 2024. Residual denoising diffusion models. In *Proceedings*

of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2773–2783.

Liu, Y.-H.; and Wang, X.-J. 2004. Spike-Frequency Adaptation of a Generalized Leaky Integrate-and-Fire Model Neuron. *Journal of Computational Neuroscience*, 10: 25–45.

Luo, X.; Yao, M.; Chou, Y.; Xu, B.; and Li, G. 2024. Integer-Valued Training and Spike-Driven Inference Spiking Neural Network for High-performance and Energy-efficient Object Detection. *European Conference on Computer Vision*.

Neftci, E. O.; Mostafa, H.; and Zenke, F. 2019. Surrogate gradient learning in spiking neural networks: Bringing the power of gradient-based optimization to spiking neural networks. *IEEE Signal Processing Magazine*, 36(6): 51–63.

Patel, K.; Hunsberger, E.; Batir, S.; and Eliasmith, C. 2021. A spiking neural network for image segmentation. *arXiv preprint arXiv:2106.08921*.

Qu, Y.; Chen, Y.; Huang, J.; and Xie, Y. 2019. Enhanced pix2pix dehazing network. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 8160–8168.

Rathi, N.; and Roy, K. 2021. Diet-snn: A low-latency spiking neural network with direct input encoding and leakage and threshold optimization. *IEEE Transactions on Neural Networks and Learning Systems*, 34(6): 3174–3182.

Rathi, N.; Srinivasan, G.; Panda, P.; and Roy, K. 2019. Enabling Deep Spiking Neural Networks with Hybrid Conversion and Spike Timing Dependent Backpropagation. In *International Conference on Learning Representations*.

Rueckauer, B.; Lungu, I.-A.; Hu, Y.; Pfeiffer, M.; and Liu, S.-C. 2017. Conversion of continuous-valued deep networks to efficient event-driven networks for image classification. *Frontiers in neuroscience*, 11: 682.

Shi, X.; Hao, Z.; and Yu, Z. 2024. SpikingResformer: Bridging ResNet and Vision Transformer in Spiking Neural Networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 5610–5619.

Song, T.; Jin, G.; Li, P.; Jiang, K.; Chen, X.; and Jin, J. 2024. Learning a Spiking Neural Network for Efficient Image Deraining. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, 1254–1262. International Joint Conferences on Artificial Intelligence Organization.

Wu, Y.; Deng, L.; Li, G.; Zhu, J.; and Shi, L. 2018. Spatio-temporal backpropagation for training high-performance spiking neural networks. *Frontiers in neuroscience*, 12: 331.

Yao, M.; Hu, J.; Hu, T.; Xu, Y.; Zhou, Z.; Tian, Y.; XU, B.; and Li, G. 2024a. 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*.

Yao, M.; Hu, J.; Zhou, Z.; Yuan, L.; Tian, Y.; Xu, B.; and Li, G. 2024b. Spike-driven transformer. *Advances in neural information processing systems*, 36.

Zhang, K.; Zuo, W.; Chen, Y.; Meng, D.; and Zhang, L. 2017. Beyond a gaussian denoiser: Residual learning of deep cnn for image denoising. *IEEE transactions on image processing*, 26(7): 3142–3155.

Zhou, Z.; Zhu, Y.; He, C.; Wang, Y.; Yan, S.; Tian, Y.; and Yuan, L. 2022. Spikformer: When spiking neural network meets transformer. *arXiv preprint arXiv:2209.15425*.