

# Spiking Denoising Diffusion Probabilistic Models

Jiahang Cao<sup>1\*</sup> Ziqing Wang<sup>2\*</sup> Hanzhong Guo<sup>3 4\*</sup> Hao Cheng<sup>1</sup> Qiang Zhang<sup>1</sup> Renjing Xu<sup>1†</sup>

From <https://arxiv.org/abs/2306.17046>

[Submitted on 29 Jun 2023]

Aug 24, 2023  
Sungmin Yoon

---

# Index

- Background
- Review of Attention mechanism
- Model Architecture
- (Experiment & Result)
- Discussion

# Introduction

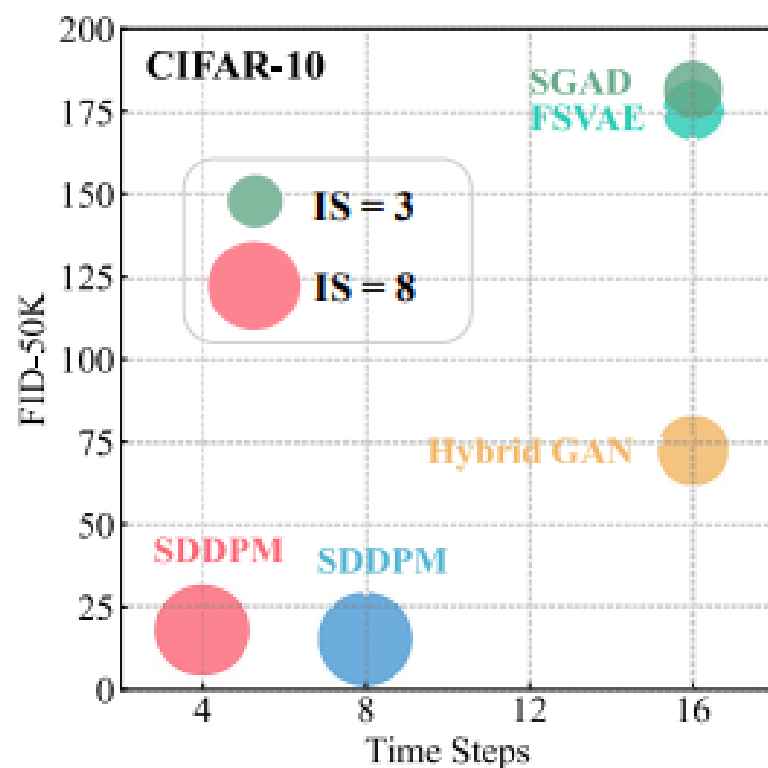


Figure 1. Comparisons of the SNN-based generative models.

## Abstract

Spiking neural networks (SNNs) have ultra-low energy consumption and high biological plausibility due to their binary and bio-driven nature compared with artificial neural networks (ANNs). While previous research has primarily focused on enhancing the performance of SNNs in classification tasks, the generative potential of SNNs remains relatively unexplored. In our paper, we put forward Spiking Denoising Diffusion Probabilistic Models (SDDPM), a new class of SNN-based generative models that achieve high sample quality. To fully exploit the energy efficiency of SNNs, we propose a purely Spiking U-Net architecture, which achieves comparable performance to its ANN counterpart using only 4 time steps, resulting in significantly reduced energy consumption. Extensive experimental results reveal that our approach achieves state-of-the-art on the generative tasks and substantially outperforms other SNN-based generative models, achieving up to  $12\times$  and  $6\times$  improvement on the CIFAR-10 and the CelebA datasets, respectively. Moreover, we propose a threshold-guided strategy that can further improve the performances by 16.7% in a training-free manner. The SDDPM symbolizes a significant advancement in the field of SNN generation, injecting new perspectives and potential avenues of exploration.

# Introduction :

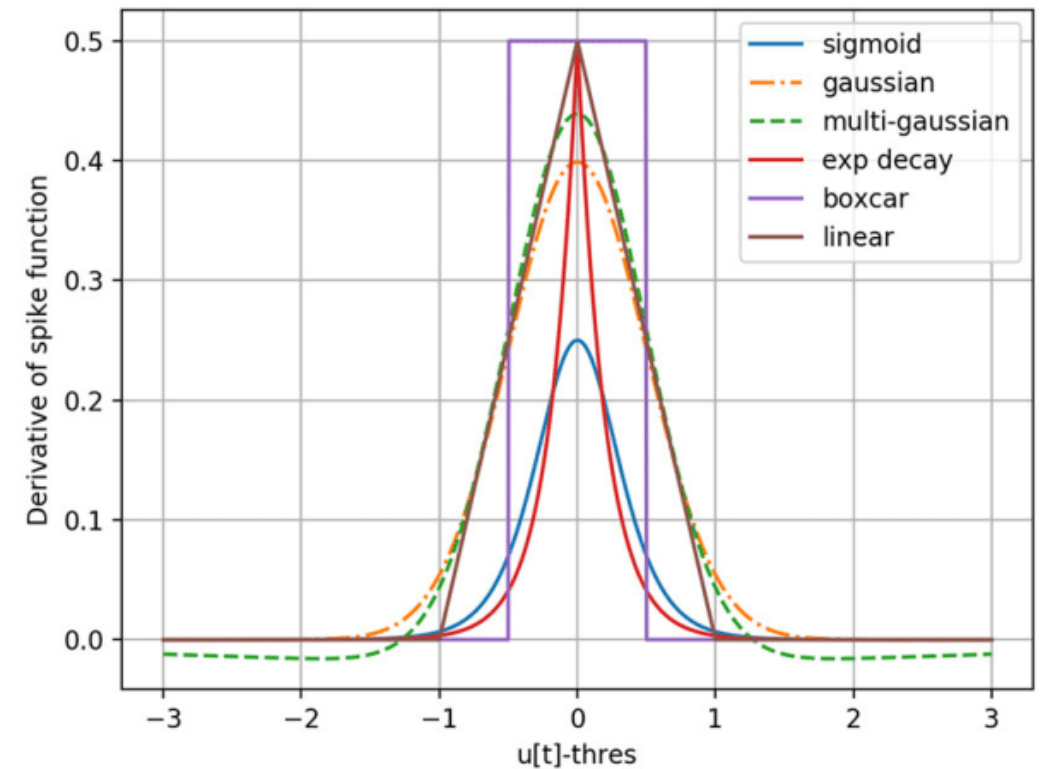
## Why they choose diffusion model?

- The generative potential of SNNs remains relatively unexplored.
- vs GAN
  - Diffusion model is more stable
  - Easier to optimize
- They said SDDPM is the first work that employs SNNs on diffusion models.

# Introduction :

## Two ways to obtain deep SNN models

- ANN-to-SNN conversion
  - Replacing the ReLU with spiking neurons
- Direct training
  - Surrogate gradients for backprop.



## Introduction :

**Ann-to-SNN usually achieve higher accuracy.  
But they choose Direct Training. Why?**

- Ann-to-SNN conversion require a longer time to train.
- Ann-to-SNN conversion could not be fully deployed on neuromorphic hardware.
  - (i.e. to reduce power consumption)

# Introduction :


## Discretize LIF

dynamic process of spike generation and can be defined as:


$$\tau \frac{dV(t)}{dt} = -(V(t) - V_{\text{reset}}) + I(t), \quad (1)$$

In practice, the dynamics need to be discretized to facilitate reasoning and training. The discretized version of LIF model can be described as:


Membrain potential  
Before reset


$$U[n] = e^{\frac{1}{\tau}} V[n-1] + I[n], \quad (2)$$

Output spike  
(formed with step func)


$$S[n] = \Theta(U[n] - \vartheta_{\text{th}}), \quad (3)$$

Membrain potential  
After triggering spike


$$V[n] = U[n](1 - S[n]) + V_{\text{reset}}S[n], \quad (4)$$

# Introduction :

## What technics they use?

- Spiking U-Net architecture
  - purely
- Pre-spike residual structure  
(first time)
- Training-free  
threshold guidance
  - (first time)

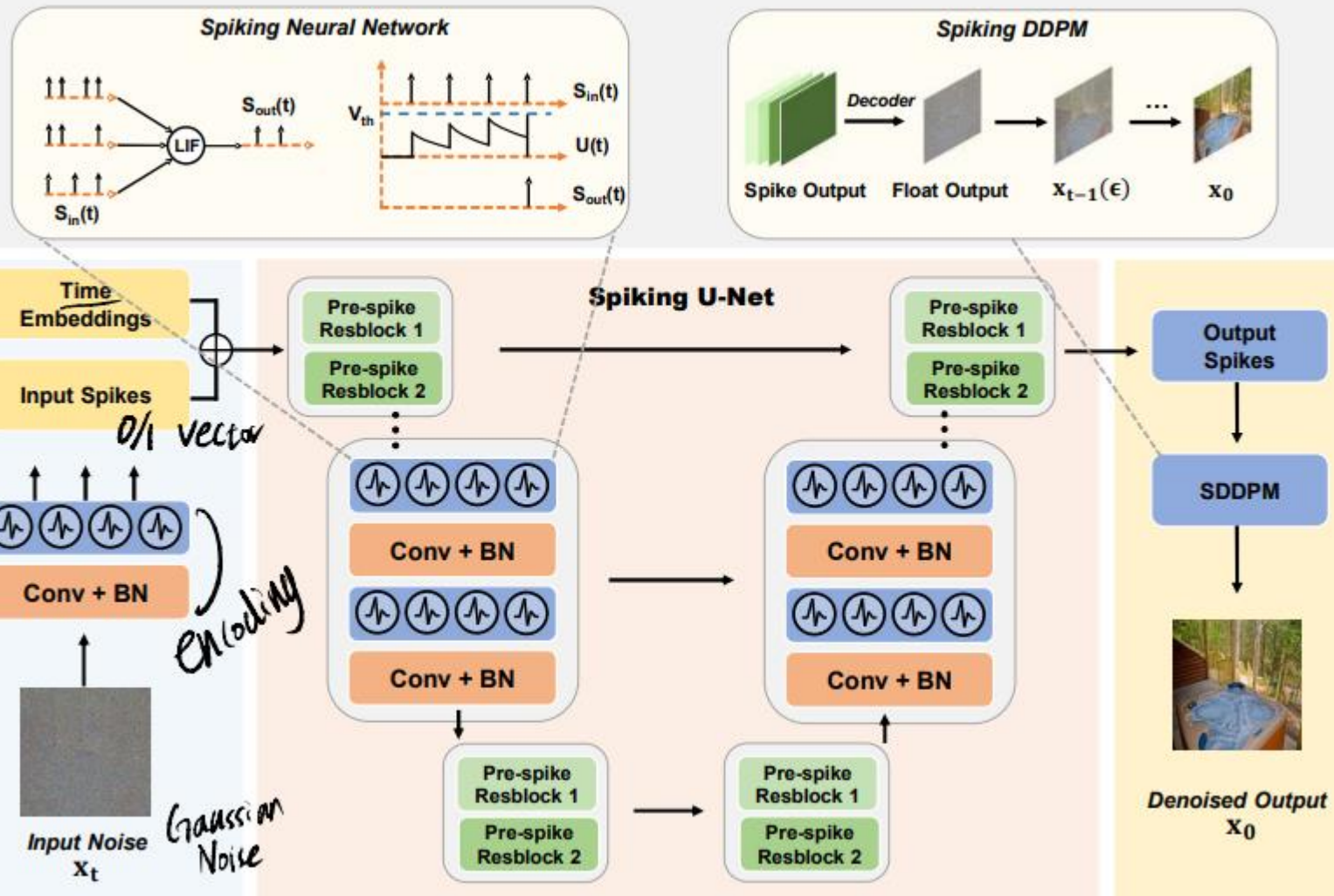
of SNNs, we propose the Spiking U-Net architecture that achieves comparable performance to its ANN counterpart while employing only 4 spiking time steps, resulting in significantly reduced energy consumption. Moreover, we employ a pre-spike structure to ensure the accurate transmission of spikes. We also propose training-free threshold guidance, which further enhances the quality of the generated images by adjusting the threshold value of the spiking neurons. Comprehensive experimental results demonstrate



# Introduction :

## What technics they use?

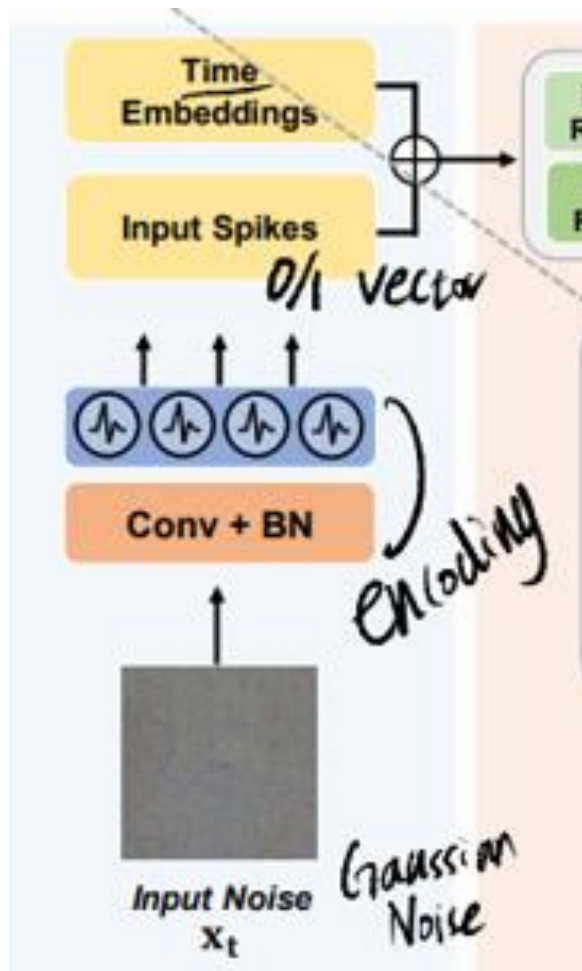
- Spiking U-Net architecture
- Pre-spike residual structure
- Training-free threshold guidance



# Introduction :

## What technics they use?

- **Spiking U-Net architecture**
- Pre-spike residual structure
- Training-free threshold guidance

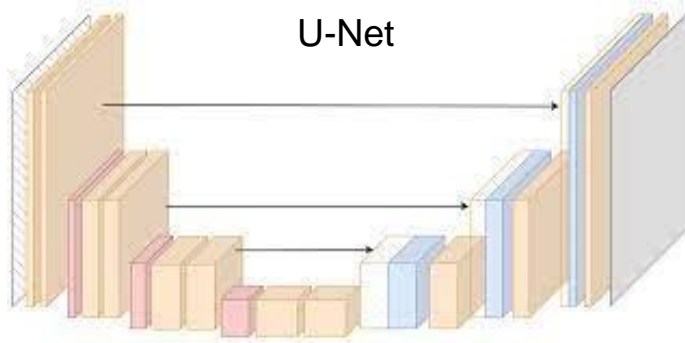


The Spiking U-Net receives an input of a 2D image batch  $I_s \in \mathbb{R}^{B \times C \times H \times W}$ , with  $B, C, H$ , and  $W$  standing for batch size, channel, height, and width, respectively. Initially, the image is replicated  $T$  times, resulting in a sequence of images  $I \in \mathbb{R}^{T \times B \times C \times H \times W}$ , a necessary operation for the SNN to incorporate temporal dimension information. However, the 2D convolution and BN cannot directly process the added  $T$  dimension. To circumvent this, we fuse the  $T$  and  $B$  dimensions, represented mathematically as  $I_{\text{fused}} \in \mathbb{R}^{TB \times C \times H \times W}$ , which allows the network to concurrently analyze spatial and temporal features.

# Introduction :

## What technics they use?

- **Spiking U-Net architecture**
- Pre-spike residual structure
- Training-free threshold guidance



The ANN-based U-Net utilized in DDPM [16] is characterized by a residual block (resblock) defined as:

$$O^l = \text{Conv}^l(\text{Swish}(\text{GN}^l(O^{l-1}))) + O^{l-1}, \quad (9)$$

→ **Distribution Mismatch in SNN**

From [42] Masked Spiking Transformer

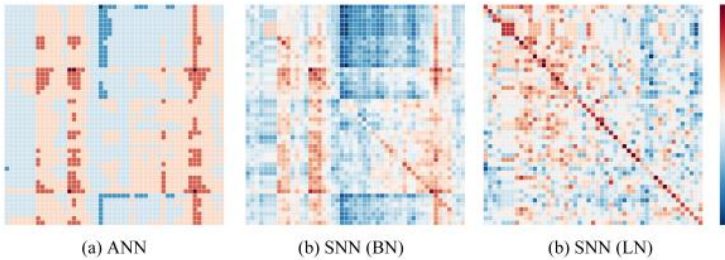


Figure 3. Illustration of distributions of (a) post-activation distribution in ANN, and (b-c) cumulative membrane potential distributions of SNN model with BN and LN, respectively. The heatmap shows a similar distribution between ANN and SNN(BN) model, but the distribution between ANN and SNN(LN) is quite different, which leads to performance degradation.

So, they use BN -> better capture spatial feature

$$O^l = \text{BN}^l(\text{Conv}^l(S^{l-1})) + S^{l-1}, \quad (10)$$

$$S^l = \text{SpikeNeuron}(O^l), \quad (11)$$

$$O^{l+1} = \text{BN}^{l+1}(\text{Conv}^{l+1}(S^l)) + S^l, \quad (12)$$

$$S^{l+1} = \text{SpikeNeuron}(O^{l+1}), \quad (13)$$

# Introduction :

## What technics they use?

- Spiking U-Net architecture
- **Pre-spike residual structure**
- Training-free threshold guidance

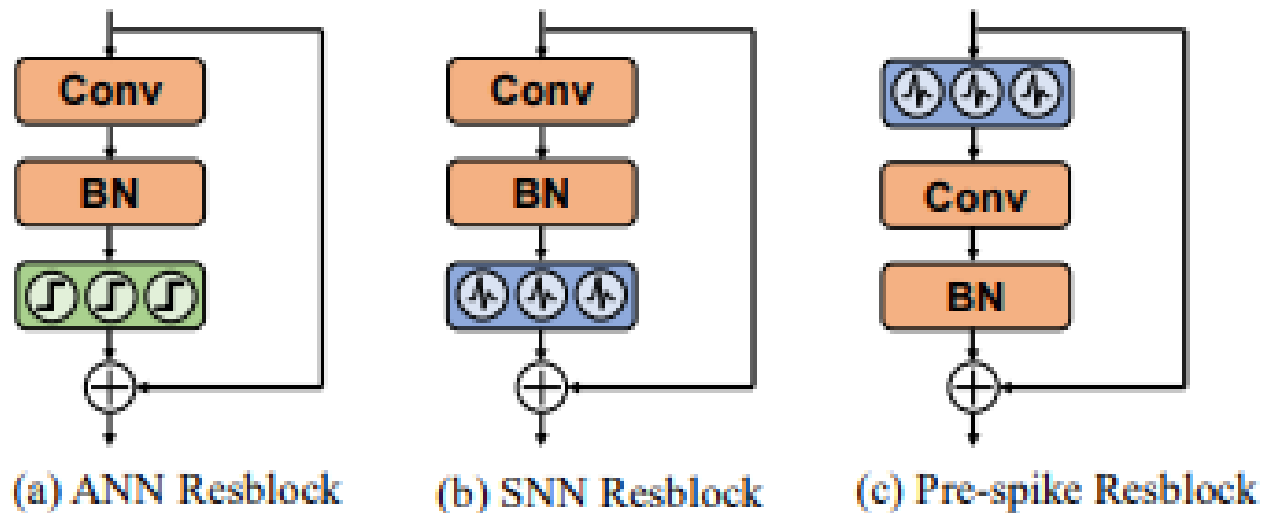


Figure 3. **Comparisons of the residual structures and Pre-spike structure.** Standard SNN resblock (b) entirely inherits from ANN structure (a). In contrast, pre-spike resblock activates first.

ply the U-Net into SNN, it could cause the output range of the residual block to overflow. This is due to the fact that the previous shallow network output  $S^{l-1}$  and the residual mapping representation  $S^l$  are both spike series ( $\{0, 1\}$ ), thus their summation  $O^l$  would result in a value domain of  $\{0, 1, 2\}$ , where  $\{2\}$  is a pathological case without any biological plausibility. This could lead to a potential impact on

Inspired by [31, 51], we for the first time apply pre-spike residual learning with the structure of *Activation-Conv-BatchNorm* in our Spiking U-Net, so as to overcome the



# Introduction :

## What technics they use?

- Spiking U-Net architecture
- **Pre-spike residual structure**
- Training-free threshold guidance

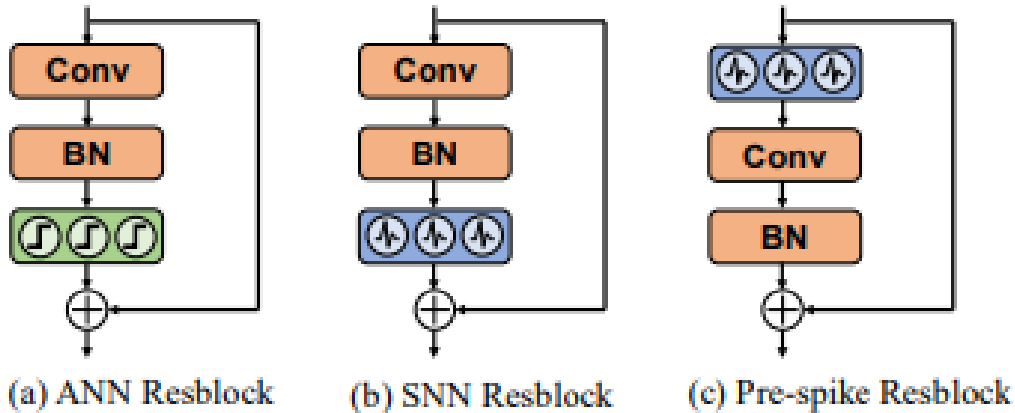


Figure 3. **Comparisons of the residual structures and Pre-spike structure.** Standard SNN resblock (b) entirely inherits from ANN structure (a). In contrast, pre-spike resblock activates first.

$$S^l = \text{SpikeNeuron}(O^{l-1}), \quad (14)$$

$$O^l = \text{BN}^l(\text{Conv}^l(S^l)) + O^{l-1}, \quad (15)$$

$$S^{l+1} = \text{SpikeNeuron}(O^l), \quad (16)$$

$$O^{l+1} = \text{BN}^{l+1}(\text{Conv}^{l+1}(S^{l+1})) + O^l. \quad (17)$$

Through the pre-spike residual mechanism, the output of the residual block can be summed by two floating points  $\text{BN}^l(\text{Conv}^l(S^l))$ ,  $O^{l-1}$  at the same scale and then enter the spiking neuron at the beginning of the next block, which guarantees that the energy consumption is still very low. We

# Introduction :

## What technics they use?

- Spiking U-Net architecture
- Pre-spike residual structure
- Training-free threshold guidance

tions (SDE):

$$dx_t = \underbrace{f(t)}_{\text{deterministic}} x_t dt + \underbrace{g(t)}_{\text{stochastic}} d\omega, \quad x_0 \sim q(x_0), \quad (5)$$

where  $\omega \in \mathbb{R}^n$  is a standard Wiener process. Let  $q(x_t)$  be the marginal distribution of the above SDE at time  $t$ . Its corresponding reversal process can be described by another SDE which recovers the data distribution from noise [42]:

$$dx = [f(t)x_t - g^2(t)\nabla_{x_t} \log q(x_t)] dt + g(t)d\bar{\omega}, \quad (6)$$

where  $\bar{\omega} \in \mathbb{R}^n$  is a reverse-time standard Wiener process and this reversal SDE starts from  $x_T \sim q(x_T)$ . In Eq. (6), the only unknown term is the score function

$$f(t) = \frac{d \log a(t)}{dt}, \quad \underbrace{g^2(t)}_{\text{variance}} = \frac{d\sigma^2(t)}{dt} - 2\sigma^2(t) \frac{d \log a(t)}{dt}. \quad (7)$$

Hence, sampling can be achieved by discretizing the reverse SDE in Eq. (6) by replacing the  $\nabla_{x_t} \log q(x_t)$  with noise network  $-\frac{\epsilon_\theta(x_t, t)}{\sigma(t)}$ . Furthermore, to enable conditional sampling, such as sampling cat images, we can refine the reverse stochastic differential equation (SDE) presented in Eq. (6) as follows [10]:

$$\epsilon_\theta(x_t, c) = \epsilon_\theta(x_t) - s\sigma(t)\nabla_{x_t} \log p_\phi(c|x_t, t), \quad (8)$$

Here,  $p_\phi(c|x_t, t)$  represents the classifier,  $s$  denotes the temperature controlling the intensity of guidance, and Eq. (8)

# Introduction :

## What technics they use?

- Spiking U-Net architecture
- Pre-spike residual structure
- Training-free threshold guidance

Therefore, in order to sample better results, we can discretize the following rectified reverse SDE [21]:

$$dx = [f(t)x_t - g^2(t)[s_\theta + c_\theta](x_t, t)] dt + g(t)d\bar{\omega}, \quad (18)$$

where  $s_\theta(x_t, t)$  represents the score network or scaled noise network, while  $c_\theta(x_t, t) = \nabla_{x_t} \log \frac{q(x_t)}{p_\theta(x_t, t)}$  denotes the rectified term for the original reverse stochastic differential equation (SDE) with the estimation errors of neural net-

$V_{th}$ , which influences the SNN's output. We put forward a threshold guidance (TG) by adjusting the threshold by:

$$\begin{aligned} & s_\theta(x_t, t, V'_{th}) \quad \text{Inference} \\ & \approx s_\theta(x_t, t, V_{th}^0) + \frac{ds_\theta(x_t, t, V_{th})}{dV_{th}} dV_{th} + O(dV_{th}) \\ & \approx s_\theta|_{V_{th}^0} + s'_\theta|_{V_{th}^0} dV_{th} + O(dV_{th}) \\ & \approx s_\theta(x_t, t) + c_\theta(x_t, t), \end{aligned} \quad (19)$$

Training time

## Experiment & Result



Figure 4. Unconditional image generation results on MNIST, Fashion-MNIST, CIFAR-10 and CelebA by using SDDPM.



# Experiment & Result

Dataset	Model	Method	#Param (M)	Time Steps	IS $\uparrow$	FID $\downarrow$
MNIST*	VAE $^{\nabla}$ [22]	ANN	1.13	/	5.947	112.50
	Hybrid GAN $^{\ddagger}$ [37]	SNN&ANN	-	16	-	123.93
	FSVAE [19]	SNN	3.87	16	6.209	97.06
	SGAD [14]	SNN	-	16	-	69.64
	<b>SDDPM</b>	SNN	63.61	4	-	<b>29.48</b>
Fashion MNIST*	VAE [22]	ANN	1.13	/	4.252	123.70
	Hybrid GAN [37]	SNN&ANN	-	16	-	198.94
	FSVAE [19]	SNN	3.87	16	4.551	90.12
	SGAD [14]	SNN	-	16	-	165.42
	<b>SDDPM</b>	SNN	63.61	4	-	<b>21.38</b>
CelebA*	VAE [22]	ANN	3.76	/	3.231	92.53
	Hybrid GAN [37]	SNN&ANN	-	16	-	63.18
	DDPM [16]	ANN	64.47	/	-	20.34
	FSVAE [19]	SNN	6.37	16	3.697	101.60
	SGAD [14]	SNN	-	16	-	151.36
CIFAR-10	<b>SDDPM</b>	SNN	63.61	4	-	<b>25.09</b>
	VAE [22]	ANN	1.13	/	2.591	229.60
	Hybrid GAN [37]	SNN&ANN	-	16	-	72.64
	DDPM [16]	ANN	64.47	/	8.380	19.04
	DDPM $_{ema}$ [16]	ANN	64.47	/	8.846	13.38
	FSVAE [19]	SNN	3.87	16	2.945	175.50
	SGAD [14]	SNN	-	16	-	181.50
	<b>SDDPM</b>	SNN	63.61	4	7.440	19.73
	<b>SDDPM</b>	SNN	63.61	8	7.584	17.27
	<b>SDDPM (TG)</b>	SNN	63.61	4	7.482	19.20
	<b>SDDPM (TG)</b>	SNN	63.61	8	<b>7.655</b>	<b>16.89</b>

Table 1. **Results for different dataset.** In all datasets, SDDPM (Ours) outperforms all SNN-based baselines and even some ANN models in terms of sample quality, which is mainly measured by FID $\downarrow$  and IS $\uparrow$ . Results of  $^{\nabla}$  and  $^{\ddagger}$  are taken from [19] and [14], respectively.  $_{ema}$  indicates the utilization of EMA training method [44]. For fair comparisons, we re-evaluate the results of DDPM [16] using the same U-Net architecture as SDDPM. We employ the symbol ‘/’ to represent ‘None’ since ANN does not have the concept of time step. \* denotes that only FID is used for measurement since these data distributions are far from ImageNet, making Inception Score less meaningful.

# Experiment & Result

Method	Threshold	FID↓	IS↑
Baseline	1.000	19.73	7.44
Inhibitory Guidance	0.999	<b>19.25</b>	<b>7.48</b>
	0.998	19.38	<b>7.55</b>
	0.997	<b>19.20</b>	7.47
Excitatory Guidance	1.001	20.00	7.47
	1.002	19.98	<b>7.48</b>
	1.003	20.04	7.46

Table 2. **Results on CIFAR-10 by different threshold guidances.** The top-1 and top-2 results are colored in red and blue, respectively. The findings indicate that TG can further enhance the FID score by adjusting the spike threshold.

Models	DDPM-ANN	SDDPM-4T	SDDPM-8T
<b>FID↓</b>	<b>19.04</b>	<b>18.57</b>	<b>15.45</b>
<b>Energy (mJ)↓</b>	<b>29.23</b>	<b>10.97</b>	22.96

Table 3. **Comparisons of energy and FID of SNN and ANN**

Method	IS↑	FID↓
SNN Resblock	6.25	48.69
<b>Pre-Spike Resblock</b>	<b>7.44</b>	<b>19.73</b>

Table 4. **Ablation study on spiking resblock structures.** We evaluate the performances of two SNN residual methods on the CIFAR-10 dataset. The results demonstrate the superiority of the pre-spike residual method.

Method	Time Steps	TG	FID↓	$\Delta$ (%)
SDDPM	4		19.73	+0.00
	4	✓	<b>19.20 (-0.53)</b>	+2.69
	8		17.27	+0.00
	8	✓	<b>16.89 (-0.38)</b>	+2.20

Table 5. **Ablation study on proposed TG and time step.** The experiments are conducted on SDDPM with 1k denoising steps.  $\Delta$  represents the improvement of FID. The performance of SDDPM is enhanced by both TG and the increasing time steps.

# Discussion

- Good point
  - Novel and good attempts
    - First time to use diffusion model in SNN, TG, pre-spike residual
  - Good performance with few steps
- Weak point
  - Only low resolution-datasets
  - What about DDIM, Analytic-DPM
    - -> They plan to...

[31] Zechun Liu, Baoyuan Wu, Wenhan Luo, Xin Yang, Wei Liu, and Kwang-Ting Cheng. Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In *ECCV*, pages 722–737, 2018. 5

