Understanding Self-attention Mechanism via Dynamical System Perspective

Zhongzhan Huang[1*] Mingfu Liang[2*] Jinghui Qin[3] Shanshan Zhong[1] Liang Lin[1†]
[1]Sun Yat-sen University [2]Northwestern University, USA [3]Guangdong University of Technology
huangzhzh23,zhongshsh5@mail2.sysu.edu.cn, mingfuliang2020@u.northwestern.edu,
scape1989@gmail.com, linliang@ieee.org

Abstract

The self-attention mechanism (SAM) is widely used in various fields of artificial intelligence and has successfully boosted the performance of different models. However, current explanations of this mechanism are mainly based on intuitions and experiences, while there still lacks direct modeling for how the SAM helps performance. To mitigate this issue, in this paper, based on the dynamical system perspective of the residual neural network, we first show that the intrinsic stiffness phenomenon (SP) in the high-precision solution of ordinary differential equations (ODEs) also widely exists in high-performance neural networks (NN). Thus the ability of NN to measure SP at the feature level is necessary to obtain high performance and is an important factor in the difficulty of training NN. Similar to the adaptive step-size method which is effective in solving stiff ODEs, we show that the SAM is also a stiffness-aware step size adaptor that can enhance the model's representational ability to measure intrinsic SP by refining the estimation of stiffness information and generating adaptive attention values, which provides a new understanding about why and how the SAM can benefit the model performance. This novel perspective can also explain the lottery ticket hypothesis in SAM, design new quantitative metrics of representational ability, and inspire a new theoretic-inspired approach, StepNet. Extensive experiments on several popular benchmarks demonstrate that StepNet can extract fine-grained stiffness information and measure SP accurately, leading to significant improvements in various visual tasks.

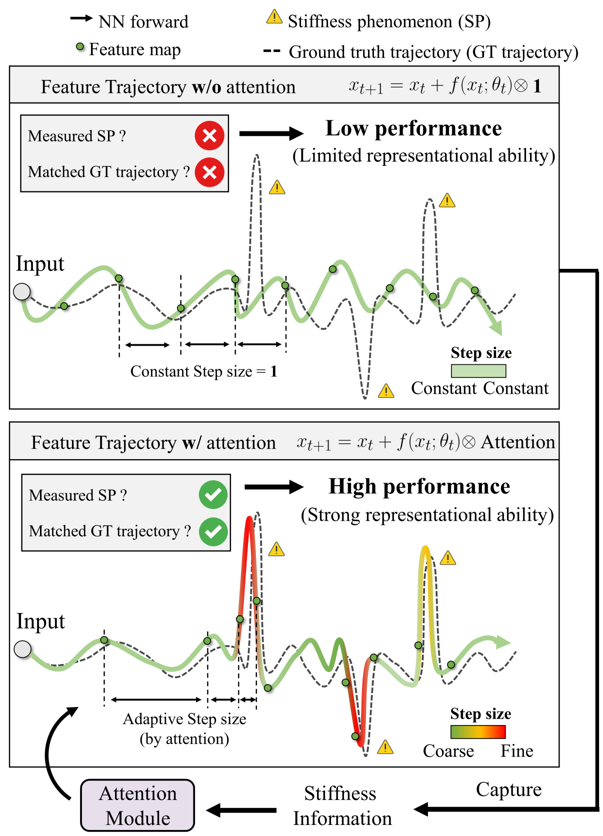

Figure 1. The feature trajectory w/ and w/o attention. The GT trajectory, generated by the best-performing model, possesses the inherent property of SP. The SAM, acting as a stiffness-aware step size adaptor, effectively extracts stiffness information and has a strong representational ability to measure the inherent SP to closely approach a GT trajectory, leading to high performance.

1. Introduction

The self-attention mechanism (SAM) [41, 15, 16, 9, 43, 4] is widely used in various artificial intelligence fields and has successfully improved the models' performance in a number of vision tasks, including image classification [22, 60, 47], object detection [37, 56, 25], instance segmentation [8, 52], image super-resolution [64, 46, 49], etc. However, most previous works lay emphasis on designing a new self-attention method, and intuitively or heuristically exploring how the self-attention mechanism helps the performance. For example, many popular channel attention methods [22, 47, 56, 35] consider the attention values as the soft weight of the channels, leading to the importance reassignment of feature maps. These soft weights can also

*co-first author † corresponding author

be seen as a gate mechanism [60, 28] to control the forward transmission of information flow, which are usually applied to neural network pruning and neural architecture search [40, 65]. Another viewpoint [38] argues that the self-attention mechanism can help to regulate the noise by enhancing instance-specific information to obtain a better regularization effect. Moreover, the receptive field [62, 63, 67] and long-range dependency [69, 17, 57] are also used to understand the role of self-attention. Although these explanations describe the behavior of self-attention mechanisms to some extent, the relationship between the SAM and model performance is still ambiguous.

To establish a more specific modeling between the SAM and model performance, in this paper, we rethink the role of the SAM with the dynamical systems perspective of neural networks (NN) with residual blocks. Specifically, we first define the stiffness phenomenon (SP) and ground truth (GT) trajectory of NNs at the feature level based on stiffness metrics and the ground truth solution of ODEs.

Next, we find that the intrinsic SP observed in the high-precision solutions of ODEs is also prevalent in high-performance NNs. This observation implies that the representational ability of NN to measure SP at the feature level is necessary to obtain high performance, as shown in Fig 1, which is hindering the learning of neural networks, and advanced training strategies are needed to achieve this requirement.

Similar to the adaptive step-size method which is effective in solving stiff ODEs, we theoretically and empirically demonstrate that the SAM is a stiffness-aware step size adaptor that can refine the estimation of stiffness information and generate the adaptive attention values to measure intrinsic SP for approaching a GT trajectory which has the upper bound performance, leading to high accuracy. This novel perspective can also explain the lottery ticket hypothesis in SAM (LTH4SA) [27], design new quantitative metrics of representational ability, and inspire a new theoretic-inspired approach, StepNet. Extensive experiments on several popular benchmarks show the effectiveness of StepNet in various vision tasks, including image classification and object detection. Our contributions are summarized as follows:

1. We propose a novel understanding of the SAM and reveal a close connection between the SAMs and the numerical solution of stiff ODEs, which is an effective explanation for understanding why and how the SAM enhances the performance of NNs.

2. Based on our novel views of SAMs, we explain the lottery ticket hypothesis in SAM, design new quantitative metrics of representational ability, and propose a powerful theoretic-inspired approach, StepNet.

2. The Stiffness and Self-attention Mechanism

In this section, we first introduce the concepts of stiffness in ODEs, SAM, and the dynamical system perspective for the NN with residual blocks. Then we further explore the SP in NNs and connect it with the SAM, which finally motivates us to propose a theoretic inspire approach.

2.1. Preliminaries and Related Works

2.1.1 The Dynamical System Perspective of NN

There are many well-known network architectures that have the residual blocks, like ResNet [18], UNet [23], Transformer [41], ResNeXt [61], etc. The residual blocks in one stage can be written as

$$x_{t+1} = x_t + f(x_t; \theta_t), \tag{1}$$

where $x_t \in \mathbb{R}^d$ is the input of NN $f(\cdot; \theta_t)$ with the learnable parameters θ_t in t^{th} block. Many recent works [3, 42, 5, 59, 51, 68, 44] have established an insightful connection between residual blocks and dynamical systems, which reveal that the residual blocks can be interpreted as one step of a forward numerical method, *i.e.*, $\mathbf{u}_{t+1} = \mathbf{u}_t + S(\mathbf{u}_t; \mathbf{f}, \Delta t) \cdot \Delta t$, for the numerical solution of an ODE as Eq.(2):

$$\mathrm{d}\mathbf{u}(t)/\mathrm{d}t = \mathbf{f}[\mathbf{u}(t)], \quad \mathbf{u}(0) = \mathbf{c}_0, \tag{2}$$

where $\mathbf{c}_0$ represents an initial condition, which corresponds to the input of the residual network. $\mathbf{u}(t) \equiv \mathbf{u}_t$ is a time-dependent d-dimensional state, which is used to describe the input feature x_t in t^{th} block. The output of neural network $f(\cdot; \theta_t)$ in t^{th} block can be regarded as an integration $S(\mathbf{u}_t; \mathbf{f}, \Delta t)$ with step size Δt using a numerical method S, e.g., the Forward Euler method [53].

2.1.2 The Stiffness in ODEs

In mathematics, a stiff equation is a differential equation [33], like Eq.(2), for which the numerical methods for solving that equation are numerically unstable, leading to poor prediction. For most ODEs, the stiffness is universal and intrinsic [34]. When the solution is unstable, we can use a fine step size Δt instead of a coarse step size to obtain finer differentiation, resulting in high-precision integration. Therefore, utilizing an adaptive step size based on a specific numerical method is the most straightforward way to solve stiffness ODEs, like the improvement from Forth order Runge–Kutta method [2] to Runge–Kutta–Fehlberg method [58, 14]. However, there is no universally accepted mathematical definition of stiffness [34], but the main idea is that the equation includes some terms that can lead to rapid variation in the solution.

Therefore, to quantify the stiffness to a certain extent, some simplified indexes are proposed, like the versatile stiffness index (SI) [1] ζ_{SI} and the stiffness-aware index (SAI) [39, 24] ζ_{SAI}. Specifically, for the dynamics of the state $\mathbf{u}(t)$ in Eq.(2), the SI at the state $\mathbf{u}(t) \equiv u^t$ is defined by $\zeta_{\text{SI}}(u^t) = \max(|\mathbf{Re}(\lambda_i)|)$, where λ_i is the eigenvalue of the Jacobian matrix at u^t for the right-hand side of Eq.(2) [1]. The maximum eigenvalue (real part) of the Jacobian matrix represents the speed of the change for the solution. In some data-driven settings, the analytic expression of the right-hand side of Eq.(2), *i.e.,* $\mathbf{f}$, is unknown. In this case, the SAI is considered since the SAI does not require $\mathbf{f}$ to be known, and the SAI can be viewed as the proxy of the SI [39]. For the discrete observation data $\{u^{t_i}\}_{i-1}^N$ from a given system, the SAI at u^{t_i} can be defined as

$$\zeta_{\text{SAI}}(u^{t_i}) = \frac{1}{\|u^{t_i}\|_2} \Big\| \frac{u^{t_{i+1}} - u^{t_i}}{t_{i+1} - t_i} \Big\|_2. \tag{3}$$

From Eq.(3), the norm of the finite difference $\frac{u^{t_{i+1}} - u^{t_i}}{t_{i+1} - t_i}$ can approximate the instantaneous change of the unknown function $\mathbf{f}$ at u^{t_i}. The term $1/\|u^{t_i}\|_2$ is used to eliminate the bias of the magnitude of $\{u^{t_i}\}_{i-1}^N$ in different coordinate systems. If we only want to measure the relative stiffness information, such as the rank of the stiffness, we can use a simplified SAI, *i.e.,* $\tilde{\zeta}_{\text{SAI}}(u^{t_i}) = \frac{u^{t_{i+1}} - u^{t_i}}{t_{i+1} - t_i}$, as a new kind of stiffness information measurement for analysis [39].

2.2. Measuring Stiffness Phenomenon (SP) in NN

In this section, based on the definition of stiffness in ODEs and the dynamical systems view of residual blocks presented in Section 2.1, we define the stiffness phenomenon (SP) in the feature trajectory $[x_1, x_2, ...]$ generated by Eq.(1) from two perspectives. Firstly, (1) from a local perspective, we propose the Neural Stiffness Index (NSI) using the idea of Eq.(3) and Definition 1 to measure the SP based on the change of adjacent features. NSI can be used to visualize the SP qualitatively and intuitively. However, according to the related works about stiff ODEs [24, 39], NSI needs to be greater than a certain threshold, *i.e.*, the local change needs to be large enough, to be considered that the SP happens. However, the threshold varies in different scenarios, and it is hard to pick a unified threshold [39].

Therefore, secondly, (2) from a global perspective, we propose a versatile metric called Total Neural Stiffness (TNS) using NSI to quantitatively measure the stiffness phenomenon of any neural networks on a given dataset. The TNS defined in Eq.(5) considers all threshold settings, including the relative threshold $\mu(1 + M_1)$ and the absolute threshold M_2, $\mathbf{M} \in \mathbb{R}^+ \times \mathbb{R}^+$. A larger TNS value indicates a more apparent stiffness phenomenon on the feature trajectory. The convergence of TNS is guaranteed by Theorem 1. Furthermore, from Eq.(4), the calculation of NSI requires the 2-norm of the feature. However, previous works [36, 29, 13, 30] show that the norm in the first block in each stage are usually extremely large and sensitive, which may affect the calculation of Eq.(5). Therefore, we will exclude these features while measuring TNS.

Definition 1. *(Neural Stiffness Index) For the feature trajectories $x_1, x_2, x_3..., x_L$ generated by a neural network with L residual blocks, i.e., $x_{t+1} = x_t + f(x_t; \theta_t) \cdot \Delta_t, t = 0, 1, .., L-1$. The Neural Stiffness Index (NSI) at x_t is*

$$\zeta_{NSI}(x_t) = \frac{1}{\|x_t\|_2} \Big\| \frac{x_{t+1} - x_t}{\Delta_t} \Big\|_2, \tag{4}$$

e.g., for NN in Eq.(1), $\zeta_{NSI}(x_t) = \|x_{t+1} - x_t\|_2 / \|x_t\|_2$.

Definition 2. *(Total Neural Stiffness) In Definition 1, the feature trajectory has the stiffness $\zeta_{NSI}(x_t; \mathbf{M})$ with degree $\mathbf{M} = (M_1, M_2)$ when $\exists t$ such that $\zeta_{NSI}(x_t) \geq \max(\mu(1 + M_1), M_2)$, where μ is the mean of all features' NSI from stage S of x_t. For input x_0 sampled from test distribution $P(x_0)$, the Total Neural Stiffness is $\iint_{\mathbf{M}} \delta(\mathbf{M}) d\mathbf{M}$, where*

$$\delta(\mathbf{M}) = \mathbb{E}_{x_0 \sim P(x_0)} \mathbf{I}_{\exists t, s.t. \zeta_{NSI}(x_t) \geq \max(\mu(1+M_1), M_2)}, \tag{5}$$

and $\mathbf{I}$ is the characteristic function, i.e., when $\zeta_{NSI}(x_t)$ not less than $\max(\mu(1 + M_1), M_2)$, $\mathbf{I} = 1$, otherwise, $\mathbf{I} = 0$, $\mathbf{M} \in \mathbb{R}^+ \times \mathbb{R}^+$.

Theorem 1. *For $\delta(\mathbf{M})$ defined as Eq.(5) , the TNS $\iint_{\mathbf{M}} \delta(\mathbf{M}) d\mathbf{M}$ is convergent. See proof in the appendix.*

2.3. The Ground Truth (GT) Trajectory

To explore the properties of the SP in the NN, it is desirable to have a "ground truth" such that we can analyze the relationship between the properties of the SP and the model's performance to propose specific improvements for the model. Given a dataset $\mathcal{D}(x, y)$ and a network $\mathcal{A}(\theta_0, s)$ with residual blocks, where θ_0 is initialization parameters and s denotes training setting (e.g., learning rate, weight decay, etc.). We consider some feature trajectories whose corresponding model $\mathcal{A}(\theta_0, s)$ have supremum performance as GT trajectories in Definition 3.

Definition 3. *(GT trajectory) For the dataset $\mathcal{D}(x, y)$, a task-oriented performance metrics κ, and a residual neural network $\mathcal{A}(\theta_0, s)$ with initialization parameters θ_0 and training setting s. After training, there are an infinite number of feature trajectories $[x_{i1}, x_{i2}, ...], i = 1, 2, ..., \infty$, from a given input x_0 to the corresponding output y_0. The GT trajectories are the feature trajectories whose performance of $\mathcal{A}(\theta_0, s)$ on $\mathcal{D}(x, y)$ can reach*

$$\sup_{\theta_0, s} \kappa(\mathcal{A}(\theta_0, s), \mathcal{D}(x, y)). \tag{6}$$

First, the GT trajectories introduced in Definition 3 exist. Let's consider the non-empty set K whose elements are the metrics κ under all initialization parameters θ_0 and training setting s. Note that the task-oriented performance metrics κ are usually bounded in various deep learning tasks, especially in supervised learning, *e.g.*, the upper

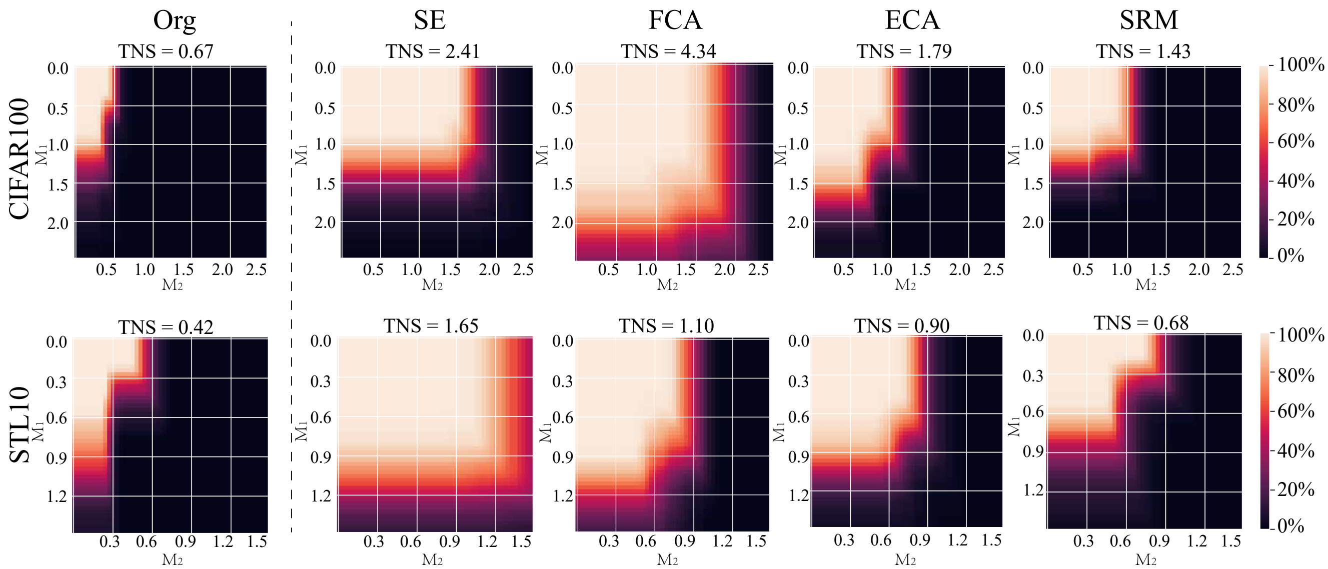

Figure 2. The quantitative visualization for SP by TNS for different models with residual blocks on CIFAR100 and STL10 datasets.

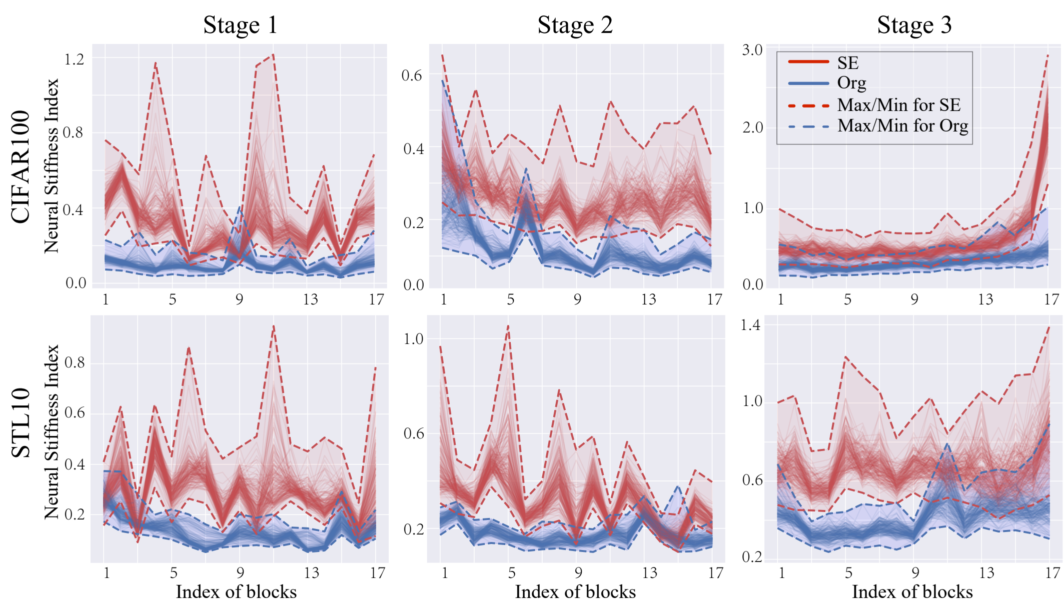

Figure 3. The intuitive visualization for SP by NSI in the SENet164 and ResNet164 on CIFAR100 and STL10 datasets.

bound for the metrics of the classification task is 100%, so $\kappa(\mathcal{A}(\theta_0, s), \mathcal{D}(x, y)) < +\infty$. According to the well-known theorem [12] that every non-empty subset $\subset \mathbb{R}$ which has an upper bound has the supremum, thus the set K has the supremum, which means the GT trajectories exist.

However, this kind of trajectory given an input x_0 may not be unique, which will be shown in Appendix. Moreover, as mentioned in Section 2.2, the feature trajectory is data-driven, and the corresponding analytic expression of the right-hand side of Eq.(2) in a NN is unknown. Thus it is infeasible to obtain the analytical form of the GT trajectories, and instead, trajectories from the model with high enough performance, like some advanced self-attention networks, can be seen as the proxies to empirically approach the properties of GT trajectories. Therefore, we take ResNet164 as the backbone and select four high-performance self-attention networks, *i.e.*, SENet [22], FCANet [50], ECANet [56], and SRMNet [35], to approximately analyze the properties of GT trajectories. In Fig.2, we show the TNS for each network with different M_1, M_2. For the CIFAR100 and STL10 datasets, from the results of all high-performance networks in Fig.2, we observe that the GT trajectories have a significantly large TNS. In other words, for most inputs, the GT trajectories have the SP. Although the TNS of the original residual neural network is relatively small, it can still measure the SP to some extent.

Moreover, as we can empirically observe that SP widely exists in these high-performance networks, we conjecture that the existence of SP is an intrinsic property of GT trajectories (More discussions are shown in Section 3.1). Thus if a NN structure can better estimate the stiffness information to measure and capture the SP, *e.g.*,, with high TNS, then such a structure can generate feature trajectories that are closer to the GT trajectories and thus achieve high performance. Otherwise, if a NN structure can not inherently measure the stiffness information, it may produce large deviations between its feature trajectory and the GT trajectories. Such a deviation will gradually accumulate with the forward process of NN, leading to poor prediction. Lastly, in Fig.3, we provide the intuitive visualization of the SP by NSI in Definition 1. Obviously, on both CIFAR100 and STL10 datasets, we can observe that the GT trajectory measured by the SENet has significant and rapid oscillations in each stage, leading to the SP. The trajectories measured by "Org" are relatively smooth but also have some oscillations that are consistent with the observation in Fig.2.

2.4. Self-attention Mechanism (SAM) and Stiffness-aware Step Size Adaptor

In this section, we introduce the self-attention mechanism (SAM) and reveal the role it plays in improving model performance based on the dynamical perspective of the neural networks introduced in the above sections. For the analysis in the main paper, we use the channel attention neural networks [22, 56, 28] as an example, and the transformer-based self-attention models [41, 54] will be discussed in Appendix. For the channel self-attention networks, they can be written as follows by comparing with Eq.(1):

$$\hat{x}_{t+1} = x_t + \underbrace{f(x_t;\theta_t)}_{\text{Feature map}} \otimes \underbrace{\mathbf{F}(f(x_t;\theta_t);\phi_t)}_{\text{Attention value}}, \tag{7}$$

where $\otimes$ is the Hadamard product and $\mathbf{F}(\cdot;\phi_t)$ is the self-attention module with learnable parameters ϕ_t in t^{th} blocks based on different attention methods. For example, in SENet, $\mathbf{F}(\cdot;\phi_t) = \mathbf{W_{t1}}(\text{ReLU}(\mathbf{W_{t2}}(\cdot)))$, where $\mathbf{W_{t1}} \in \mathbb{R}^{d\times r}$ and $\mathbf{W_{t2}} \in \mathbb{R}^{r\times d}$ are learnable matrices, $r < d$.

Compared with the forward numerical method mentioned in Section 2.1.2, the attention value $\mathbf{F}(f(x_t;\theta_t);\phi_t)$ in Eq.(7) can be referred to as the step size in solving ODEs, and the original residual neural networks can be rewritten as

$$x_{t+1} = x_t + f(x_t;\theta_t) \otimes \Delta t, \tag{8}$$

where the step size $\Delta t = \mathbf{1}$. By comparing Eq.(7) and Eq.(8), we can readily find that the attention value generated by the SAM exactly serves as an adaptive step size, *i.e.*, $\Delta t = \mathbf{F}(f(x_t;\theta_t);\phi_t)$. Moreover, since the last layers of various self-attention modules are usually Sigmoid function or Softmax function, thus the attention values in Eq.(7) are less than 1, *i.e.*, $\Delta t = \mathbf{F}(f(x_t;\theta_t);\phi_t) < 1$. This implies that with SAM, we can provide a smaller and more flexible step size than that from the original residual neural network. Now we show that the step size generated by the SAM is also stiffness-aware. As mentioned in Section 2.1.2, we can use $(x_{t+1} - x_t)/\Delta t$ to measure the stiffness information in feature trajectory at x_t. Then for the the original residual neural networks in Eq.(1), we have

$$f(x_t;\theta_t) = \frac{x_{t+1} - x_t}{\Delta t}|_{\Delta t=1} \equiv \check{\zeta}_{\text{NSI}}(x_t), \tag{9}$$

and hence the $f(x_t;\theta_t)$ can be regarded as a kind of coarse stiffness information with step size $\Delta t = 1$, which is consistent with the discussions of Fig.2 and Fig.3 in Section 2.3. For the SAM, we have

$$\begin{aligned}\mathbf{F}(f(x_t;\theta_t);\phi_t) &= \mathbf{F}(\underbrace{\frac{1}{\Delta t}(x_{t+1} - x_t)|_{\Delta t=1}}_{\text{Coarse stiffness information}};\phi_t) \\ &= \mathbf{F}(\check{\zeta}_{\text{NSI}}(x_t);\phi_t).\end{aligned} \tag{10}$$

From Eq.(10), we can summarize how the self-attention module helps the performance of the original residual neural networks: <u>(1) Capture the stiffness information.</u> the self-attention module $\mathbf{F}(\cdot;\phi_t)$ take the accessible and coarse stiffness information $f(x_t;\theta_t)$ from Eq.(9) as input. Then as shown in Fig.2 and Fig.3, the self-attention module can refine this coarse information to obtain a finer estimation of stiffness information; <u>(2) Generate the adaptive step size</u>. Based on this finer estimation, the module $\mathbf{F}(\cdot;\phi_t)$ outputs suitable attention values $\mathbf{F}(f(x_t;\theta_t);\phi_t)$ to adaptively measure the SP in the neural network, which means the SAM can enhance the representational ability of NN. For example, if the feature trajectory at x_t needs to measure a large NSI, from Eq.(4), the attention value $\mathbf{F}(f(x_t;\theta_t);\phi_t)$ can be small to get the large $\zeta_{\text{NSI}}(x_t) = \frac{1}{\|x_t\|_2}\left\|\frac{\hat{x}_{t+1}-x_t}{\mathbf{F}(f(x_t;\theta_t);\phi_t)}\right\|_2$.

2.5. Theoretic-inspired Approach: StepNet

From Section 2.4 and Eq.(10), we know that the ability to properly estimate the stiffness information is essential for the performance of the self-attention module. Thus if we want to obtain better model performance, we can consider estimating other accessible and better stiffness information in the self-attention module. Now we introduce a better self-attention formulation to capture better stiffness information, which is motivated by the asymptotic analysis between SI and SAI as follows. In Section 2.1.2, SAI is used as a proxy for the versatile index of stiffness (SI) to measure stiffness information in ODEs for tackling the computational difficulties of SI in data-driven problems. In Theorem 2, we first show how SAI can approximate the SI.

Theorem 2. *For an ODE $d\mathbf{u}(t)/dt = \mathbf{f}[\mathbf{u}(t)]$ defined at Eq.(2), if the Jacobian matrix $\mathbf{J}_{u^t}$ at u^t is a $n\times n$ symmetric real matrix and $\{\lambda_i\}_{i=1}^n$ are its n distinct eigenvalues, and*

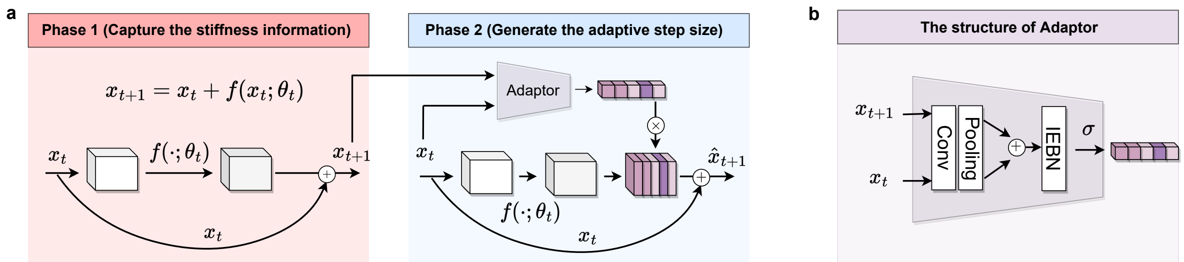

Figure 4. The architecture of StepNet. **a,** the forward process of StepNet includes two phases. **b,** the structure of Adaptor. "Conv" is group convolution, "Pooling" is global average pooling and "IEBN" [38] is the combination of batch normalization and a linear transformation "IE" from [25]. σ is Sigmoid activation function. More training details are provided in Appendix.

Model	CIFAR10			CIFAR100			STL10		
	#P(M)	top-1 acc. (%)	acc. ↑	#P(M)	top-1 acc. (%)	acc. ↑	#P(M)	top-1 acc. (%)	acc. ↑
ResNet [18]	1.70	93.35 ± 0.18	-	1.73	74.30 ± 0.30	-	1.70	82.66 ± 1.05	-
+SE [22]	1.91	94.26 ± 0.14	0.91	1.93	75.25 ± 0.31	0.95	1.91	83.96 ± 0.92	1.30
+CBAM [60]	1.92	93.91 ± 0.14	0.56	1.94	74.68 ± 0.22	0.38	1.92	84.11 ± 0.49	1.45
+ECA [56]	1.70	94.15 ± 0.23	0.80	1.73	74.41 ± 0.27	0.11	1.70	83.89 ± 1.21	1.23
+SEM [66]	1.95	94.52 ± 0.11	1.17	1.97	76.49 ± 0.26	2.19	1.95	<u>85.73 ± 0.16</u>	<u>3.07</u>
+SRM [35]	1.74	94.59 ± 0.12	1.24	1.76	75.49 ± 0.31	1.19	1.74	84.73 ± 0.12	2.07
+FCA [50]	1.91	94.66 ± 0.12	1.31	1.93	<u>76.52 ± 0.19</u>	<u>2.21</u>	1.91	85.34 ± 0.25	2.68
+IEBN [38]	1.73	<u>94.70 ± 0.13</u>	<u>1.35</u>	1.75	76.32 ± 0.14	2.02	1.73	85.13 ± 0.04	2.47
+StepNet (Ours)	1.75	**95.14 ± 0.24**	**1.79**	1.77	**77.04 ± 0.22**	**2.74**	1.75	**86.08 ± 0.11**	**3.42**

Table 1. The classification accuracy on CIFAR10, CIFAR100, and STL10. "#P(M)" means the number of parameters (million). Bold and underline indicate the best results and the second best results, respectively.

$\mathbf{Re}(\lambda_i) < 0, i = 1, 2, ..., n,$ *then*

$$\zeta_{SAI}(u^t) \approx \zeta_{SI}(u^t) \cdot \sqrt{c + Q[\zeta_{SI}(u^t)]}, \tag{11}$$

where c is a constant and $Q(\cdot)$ is a function with respect to $\zeta_{SI}(u^t)$ and when $\zeta_{SI}(u^t)$ is large enough, $Q[\zeta_{SI}(u^t)]$ converges to a 0. See proof in the appendix.

From Eq.(11) we can see that $\sqrt{c + Q[\zeta_{\text{SI}}(u^t)]}$ tends to $\sqrt{c}$ when SI is large and then $\zeta_{\text{SAI}}(u^t) \propto \zeta_{\text{SI}}(u^t)$, *i.e.*, $\zeta_{\text{SAI}}(u^t)$ and $\zeta_{\text{SI}}(u^t)$ are positively correlated, which make the Eq.(9) constructed from SAI provide relatively accurate stiffness information. However, if the SI is not large enough, the SAI and SI will be nonlinearly related, in other words, the stiffness information provided by Eq.(9) may not be precise enough and need further refinement.

Motivated by the above analysis, we propose a novel self-attention network called StepNet. As discussed in Section 2.1.2, the stiffness in the ODE may measure a changing trend like the rapid variation in solution, which implies that the stiffness information should be measured by two adjacent states. Therefore, for Eq.(10), we use a data-driven function $\tilde{\mathbf{F}}(x_{t+1}, x_t; \tilde{\phi}_t) = \tilde{\mathbf{F}}(x_t + f(x_t;\theta_t), x_t; \tilde{\phi}_t)$ to replace $\mathbf{F}(x_{t+1} - x_t; \phi_t)$, such that the adaptive step size generated in our StepNet is based on the intrinsic relation between two adjacent states x_{t+1} and x_t to better model the stiffness information. Thus, the Eq.(7) can be rewritten as

$$\begin{aligned} \hat{x}_{t+1} &= x_t + f(x_t;\theta_t) \otimes \mathbf{F}(f(x_t;\theta_t);\phi_t) \\ &\leftarrow x_t + f(x_t;\theta_t) \otimes \tilde{\mathbf{F}}(x_t + f(x_t;\theta_t), x_t; \tilde{\phi}_t). \end{aligned} \tag{12}$$

From Eq.(12), the calculation of our StepNet has two phases: (1) In Fig.4 (a), we first estimate a coarse $(t+1)^{\text{th}}$ feature map $x_{t+1} = x_t + f(x_t;\theta_t)$ by Eq.(1); (2) After that, the Adaptor $\tilde{\mathbf{F}}(\cdot,\cdot;\tilde{\phi}_t)$ take both the x_t and x_{t+1} as input to generate the adaptive attention values, which can better measure the stiffness information to generate finer step size for better capturing the SP to enhance the representational ability of the model and boost the performance. The network architecture of the Adaptor is shown in Fig.4 (b) and more training details are provided in Appendix.

3. Experiment

In this section, we use several popular vision benchmarks to verify the effectiveness of the proposed StepNet, including image classification and object detection. All experiments are verified 5 times with random seeds, and the average performances with standard deviations are reported. The experimental settings can be found in Appendix.

Image classification. We compare the proposed StepNet with several existing self-attention modules on four datasets for image classification. These four datasets are CIFAR10 [32], CIFAR100 [32], STL-10 [11] and ImageNet [55], and the results of these datasets are listed in Table 1 and 2, which show that the StepNet improves the accuracy significantly over the original networks and consistently compared with other existing self-attention modules under different datasets and backbones.

Object detection. We further conduct experiments for

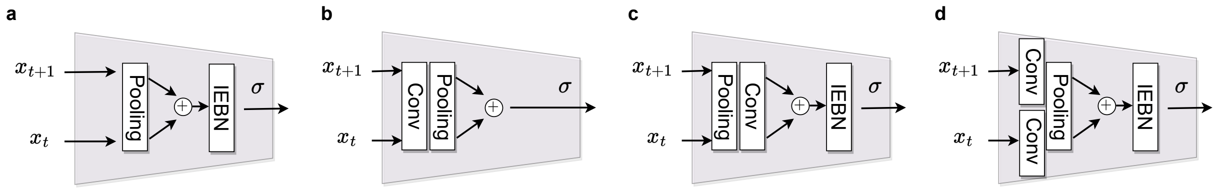

Figure 5. Different structures of Adaptor.

Model	ResNet34-acc.	ResNet50-acc.
Org	73.99± 0.04	76.02± 0.09
SE [22]	74.37± 0.10	76.62± 0.02
ECA [56]	74.56± 0.08	77.07± 0.09
CBAM [60]	74.89± 0.11	76.38± 0.10
SRM [35]	74.79± 0.04	76.49± 0.03
FCA [50]	**75.08± 0.19**	77.22± 0.15
SGE [37]	74.50± 0.03	77.13± 0.06
StepNet (Ours)	75.01± 0.09	**77.52± 0.04**

Table 2. The classification performance on ImageNet. The best and the second best results of each setting are marked in bold and *italic* fonts, respectively.

Model	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
ResNet50	36.9	57.5	39.4	22.1	39.6	46.4
+SE [22]	38.3	60.1	41.5	23.0	42.1	49.2
+ECA [56]	38.5	60.4	41.5	23.2	42.2	48.9
+SGE [37]	38.9	60.0	**42.0**	23.2	41.9	49.0
+StepNet (Ours)	**39.8**	**61.2**	41.9	**23.8**	**43.1**	**51.0**
ResNet101	38.9	60.3	41.4	22.1	43.9	50.0
+SE [22]	39.8	61.2	43.4	23.6	44.2	51.1
+ECA [56]	40.4	**62.7**	**44.5**	24.1	45.1	51.8
+SGE [37]	40.2	61.4	43.2	24.1	44.5	51.2
+StepNet (Ours)	**41.9**	62.6	44.4	**25.4**	**46.8**	**53.2**

Table 3. The object detection performance on MS COCO. The best and the second best results of each setting are marked in bold and *italic* fonts, respectively.

object detection tasks on the MS COCO dataset with Faster R-CNN, and the results are shown in Table 3, where our StepNet outperforms the baseline models by 2.9% and 3.0% in terms of AP on ResNet50 and ResNet101, respectively. Moreover, the StepNet also achieves good enough performance improvements in other object detection metrics. All the results show that our new understanding of the SAM can effectively help us propose a well design for SAM.

3.1. Ablation Study and Discussions

Now we further explore the property of the StepNet and the stiffness phenomenon in neural networks.

The structure of StepNet. In Section 2.4 we summarize that the self-attention mechanism can help the model performance in two ways, *i.e.*, the extraction of stiffness information and the generation of adaptive step sizes. In fact, these two ways correspond to (1) the input of the self-attention module and (2) the design of the module. For (1), in Table 4, we explore the performance when x_t or x_{t+1} is removed, respectively. The results show that input with x_t and x_{t+1} simultaneously is necessary to estimate fine stiffness information as discussed in Section 2.5. Moreover, $F(x_{t+1}-x_t)$ means that we only use the adaptor in StepNet as the self-attention module following the normal paradigm without considering the better estimation of stiffness information in Eq.(12). Comparing the result of $F(x_{t+1}-x_t)$ and our $F(x_{t+1}, x_t)$, we can see that a finer estimation of stiffness information like ours is necessary to achieve better performance. For (2), as shown in Fig.5, we constructed four alternative adaptor structures. The experimental results in Table 4 illustrate that all four alternatives are inferior to ours. However, the best structure to generate the adaptive step sizes is still unknown, and in the future, we can still improve the design of the adaptor, *e.g.,* to better utilize x_t and x_{t+1} through a neural network effectively.

Model	CIFAR10	CIFAR100	STL10
Org	93.35± 0.18	74.30± 0.30	82.66± 1.05
$F(x_t)$	94.59± 0.14	76.34± 0.14	84.74± 0.64
$F(x_{t+1})$	94.32± 0.23	76.53± 0.04	85.12± 0.54
$F(x_{t+1}-x_t)$	94.69± 0.33	76.42± 0.10	85.26± 0.72
$F(x_{t+1}, x_t)$ (ours)	**95.14± 0.24**	**77.04±0.22**	**86.08±0.11**
Adaptor **a**	94.39± 0.12	76.19± 0.14	84.21± 0.12
Adaptor **b**	94.05± 0.11	75.45± 0.12	84.10± 0.24
Adaptor **c**	94.19± 0.24	75.12± 0.08	84.12± 0.44
Adaptor **d**	94.92± 0.24	76.83± 0.14	85.62± 0.44

Table 4. The ablation study on the structure of StepNet.

The property of stiffness phenomenon (SP) and the LTH4SA. In fact, for any input x_0 and its corresponding output y_0, the feature trajectory built by a well-trained residual neural network has two properties: (1) For most inputs x_0, their feature trajectories have SP; (2) For each trajectory, only a few features can cause SP. Specifically, for property (1), in Section 2.3, we approximate the GT trajectories with several advanced self-attention models. From Fig.2 and Fig.3, we can empirically observe that for most of the inputs, their feature trajectories have SP. In the appendix, we provide more visualizations of feature trajectories using SENet as an example, and these visualizations also provide empirical evidence for property (1). For property (2), we define the stiffness proportion $\hat{p}=\frac{1}{L}\mathbb{E}_{x_0\sim P(x_0)}\#\{t|\zeta_{\mathrm{NSI}}(x_t)\geq\max(\mu(1+M_1),M_2)\}$ to measure the expected number that how many features from a feature trajectory have $\zeta_{\mathrm{NSI}}(x_t;\mathbf{M})$ with degree $\mathbf{M}$. If the $\hat{p}$ of a feature trajectory is close to 100%, it means that

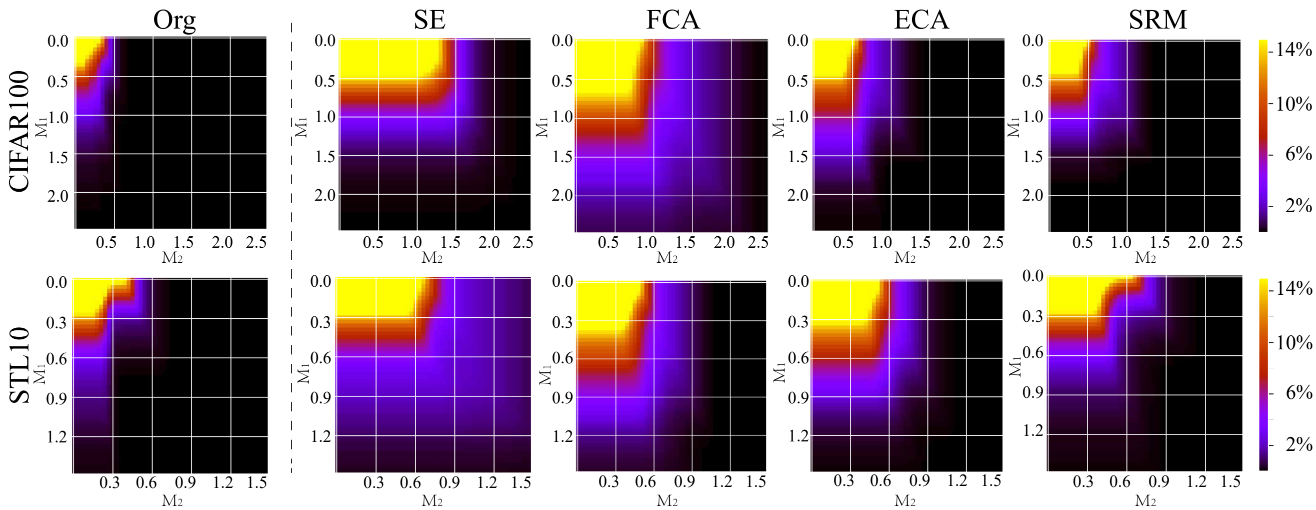

Figure 6. The stiffness proportion for different models with residual blocks on CIFAR100 and STL10.

this trajectory has many features with large enough NSI. For Fig.4, we show the corresponding stiffness proportion in Fig.6. We can observe that for various M_1 and M_2, most of the $\hat{p} \leq 10\%$, which indicates only a few features in GT trajectory can cause SP.

Moreover, these two properties are also consistent with those intrinsic properties in physics dynamical systems. For instance, the close encounter is an important factor to cause the SP in the three-body motion. [38] considers 1,000 independent simulations of three-body trajectories following [10], where 91.4% of the trajectories contain the close encounter, but on average only 4.2% of the time intervals within the trajectories contain close encounter.

In addition, we identify that these two properties are closely related to the lottery ticket hypothesis [27] in the self-attention mechanism (LTH4SA). LTH4SA reveals that we only need to insert the self-attention module on a small number of blocks to achieve remarkable improvement for a NN. According to property (1), for most inputs, their GT trajectories have SP, and as mentioned in Section 2.4, the adaptive step size generated by the SAM can improve the representational ability of NN. Thus the SAM is valid for most of the inputs. Moreover, property (2) tells us that only a small part of the features in a feature trajectory can cause SP, and thus we only need to set the module on a small number of blocks to measure the SP of the whole trajectory. So if these two properties generally hold, we argue that LTH4SA may also be an intrinsic property of the SAM.

Why do the GT trajectories have SP? Now we attempt to understand why most GT trajectories have SP, which can help us design novel methods to boost the performance of representation learning. From Eq.(4) and Eq.(9), we know that for a well-trained residual neural network, $f(x_t;\theta_t)$ provides stiffness information and $\zeta_{\mathrm{NSI}}(x_t) = \mathcal{O}(\|f(x_t;\theta_t)\|_2)$. When NSI is large, $\|f(x_t;\theta_t)\|_2$ is also large, which means that the elements (absolute values) of the output feature of the neural network $f(\cdot;\theta_t)$ at t-th block are relatively large. In some previous works [36, 21, 48], such kinds of features are considered important features and have major contributions to the model performance. In other words, a residual network can achieve high performance, *i.e.*, it can approximate the GT trajectory, probably because the network has the ability to learn such important (stiff) features by the adaptive step sizes in a few blocks. So we further calculate the rank correlation (kendall correlation [31] and spearman correlation [45]) between the TNS and the model performance. The results are presented in Fig.7, which shows that the performance of the models and their representational ability to measure the SP are positively correlated. Moreover, as the TNS can reflect the ability of the model to measure the SP, thus the TNS can also be a novel representational ability metric to evaluate the neural network in practice and has the potential to be used in network formulation, such as neural architecture search [26, 40], network pruning [36, 20, 19] or other applicaton [6, 7].

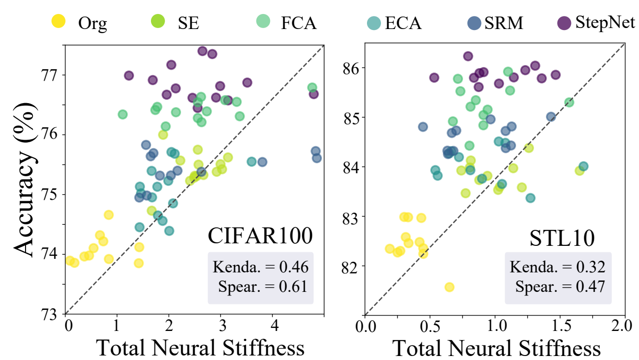

Figure 7. The correlation between the accuracy and TNS.

4. Conclusion

In this paper, we bridge the relationship between the self-attention mechanism (SAM) and the numerical solution of stiff ordinary differential equations, which reveals that the SAM is a stiffness-aware step size adaptor that can refine the estimation of stiffness information and generate suitable attention values for adaptively measuring the stiffness phenomenon in the neural network (NN) to enhance the representational ability of the NN and achieve high performance.

5. Acknowledgment

This work was supported in part by National Key R&D Program of China under Grant No. 2020AAA0109700, National Natural Science Foundation of China (NSFC) under Grant No.61836012, Guangdong Basic and Applied Basic Research Foundation No. 2023A1515011374, National Natural Science Foundation of China (NSFC) under Grant No. 62206314, GuangDong Basic and Applied Basic Research Foundation under Grant No. 2022A1515011835.

References

[1] Richard C Aiken. *Stiff computation*, volume 169. Oxford University Press Oxford, 1985.

[2] John Charles Butcher. *Numerical methods for ordinary differential equations*. John Wiley and Sons, 2016.

[3] Bo Chang, Lili Meng, Eldad Haber, Frederick Tung, and David Begert. Multi-level residual networks from dynamical systems view. *arXiv preprint arXiv:1710.10348*, 2017.

[4] Chun-Fu Richard Chen, Quanfu Fan, and Rameswar Panda. Crossvit: Cross-attention multi-scale vision transformer for image classification. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 357–366, 2021.

[5] Ricky TQ Chen, Yulia Rubanova, Jesse Bettencourt, and David K Duvenaud. Neural ordinary differential equations. *Advances in neural information processing systems*, 31, 2018.

[6] Tianshui Chen, Liang Lin, Riquan Chen, Xiaolu Hui, and Hefeng Wu. Knowledge-guided multi-label few-shot learning for general image recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(3):1371–1384, 2022.

[7] Tianshui Chen, Tao Pu, Hefeng Wu, Yuan Xie, Lingbo Liu, and Liang Lin. Cross-domain facial expression recognition: A unified evaluation benchmark and adversarial graph learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(12):9887–9903, 2022.

[8] Xier Chen, Yanchao Lian, Licheng Jiao, Haoran Wang, Yan-Jie Gao, and Shi Lingling. Supervised edge attention network for accurate image instance segmentation. In *European Conference on Computer Vision*, pages 617–631. Springer, 2020.

[9] Zhengsu Chen, Lingxi Xie, Jianwei Niu, Xuefeng Liu, Longhui Wei, and Qi Tian. Visformer: The vision-friendly transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 589–598, 2021.

[10] Zhengdao Chen, Jianyu Zhang, Martin Arjovsky, and Léon Bottou. Symplectic recurrent neural networks. *arXiv preprint arXiv:1909.13334*, 2019.

[11] Adam Coates, Andrew Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In Geoffrey Gordon, David Dunson, and Miroslav Dudík, editors, *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15 of *Proceedings of Machine Learning Research*, pages 215–223, Fort Lauderdale, FL, USA, 11–13 Apr 2011. PMLR.

[12] Richard Courant, Fritz John, Albert A Blank, and Alan Solomon. *Introduction to calculus and analysis*, volume 1. Springer, 1965.

[13] Xiaohan Ding, Xiangxin Zhou, Yuchen Guo, Jungong Han, Ji Liu, et al. Global sparse momentum sgd for pruning very deep neural networks. In *Advances in Neural Information Processing Systems*, pages 6379–6391, 2019.

[14] Erwin Fehlberg. *Low-order classical Runge-Kutta formulas with stepsize control and their application to some heat transfer problems*, volume 315. National aeronautics and space administration, 1969.

[15] Meng-Hao Guo, Tian-Xing Xu, Jiang-Jiang Liu, Zheng-Ning Liu, Peng-Tao Jiang, Tai-Jiang Mu, Song-Hai Zhang, Ralph R Martin, Ming-Ming Cheng, and Shi-Min Hu. Attention mechanisms in computer vision: A survey. *Computational Visual Media*, pages 1–38, 2022.

[16] Kai Han, Yunhe Wang, Hanting Chen, Xinghao Chen, Jianyuan Guo, Zhenhua Liu, Yehui Tang, An Xiao, Chunjing Xu, Yixing Xu, et al. A survey on vision transformer. *IEEE transactions on pattern analysis and machine intelligence*, 2022.

[17] Yifan Hao and Huiping Cao. A new attention mechanism to classify multivariate time series. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2020.

[18] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition*, 2016.

[19] Wei He, Zhongzhan Huang, Mingfu Liang, Senwei Liang, and Haizhao Yang. Blending pruning criteria for convolutional neural networks. In *International Conference on Artificial Neural Networks*, pages 3–15. Springer, 2021.

[20] Wei He, Meiqing Wu, Mingfu Liang, and Siew-Kei Lam. Cap: Context-aware pruning for semantic segmentation. In *Proceedings of the ieee/cvf winter conference on applications of computer vision*, pages 960–969, 2021.

[21] Hengyuan Hu, Rui Peng, Yu-Wing Tai, and Chi-Keung Tang. Network trimming: A data-driven neuron pruning approach towards efficient deep architectures. *arXiv preprint arXiv:1607.03250*, 2016.

[22] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.

[23] Huimin Huang, Lanfen Lin, Ruofeng Tong, Hongjie Hu, Qiaowei Zhang, Yutaro Iwamoto, Xianhua Han, Yen-Wei

Chen, and Jian Wu. Unet 3+: A full-scale connected unet for medical image segmentation. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1055–1059. IEEE, 2020.

[24] Weizhang Huang and Benedict Leimkuhler. The adaptive Verlet method. *SIAM Journal on Scientific Computing*, 18(1):239–256, 1997.

[25] Zhongzhan Huang, Senwei Liang, Mingfu Liang, Weiling He, and Liang Lin. Layer-wise shared attention network on dynamical system perspective. *arXiv preprint arXiv:2210.16101*, 2022.

[26] Zhongzhan Huang, Senwei Liang, Mingfu Liang, Wei He, and Haizhao Yang. Efficient attention network: Accelerate attention by searching where to plug. *arXiv preprint arXiv:2011.14058*, 2020.

[27] Zhongzhan Huang, Senwei Liang, Mingfu Liang, Wei He, Haizhao Yang, and Liang Lin. The lottery ticket hypothesis for self-attention in convolutional neural network. *arXiv preprint arXiv:2207.07858*, 2022.

[28] Zhongzhan Huang, Senwei Liang, Mingfu Liang, and Haizhao Yang. Dianet: Dense-and-implicit attention network. In *AAAI*, pages 4206–4214, 2020.

[29] Zhongzhan Huang, Wenqi Shao, Xinjiang Wang, Liang Lin, and Ping Luo. Convolution-weight-distribution assumption: Rethinking the criteria of channel pruning. *arXiv preprint arXiv:2004.11627*, 2020.

[30] Zhongzhan Huang, Wenqi Shao, Xinjiang Wang, Liang Lin, and Ping Luo. Rethinking the pruning criteria for convolutional neural network. *Advances in Neural Information Processing Systems*, 34:16305–16318, 2021.

[31] Maurice G Kendall. A new measure of rank correlation. *Biometrika*, 30(1/2):81–93, 1938.

[32] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009.

[33] John Denholm Lambert. *The initial value problem for ordinary differential equations: A survey*. University of Dundee. Department of Mathematics and Computer Science, 1976.

[34] John Denholm Lambert et al. *Numerical methods for ordinary differential systems*, volume 146. Wiley New York, 1991.

[35] HyunJae Lee, Hyo-Eun Kim, and Hyeonseob Nam. Srm: A style-based recalibration module for convolutional neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 1854–1862, 2019.

[36] Hao Li, Asim Kadav, Igor Durdanovic, Hanan Samet, and Hans Peter Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[37] Xiang Li, Xiaolin Hu, and Jian Yang. Spatial group-wise enhance: Improving semantic feature learning in convolutional networks. *arXiv preprint arXiv:1905.09646*, 2019.

[38] Senwei Liang, Zhongzhan Huang, Mingfu Liang, and Haizhao Yang. Instance enhancement batch normalization: An adaptive regulator of batch noise. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 4819–4827, 2020.

[39] Senwei Liang, Zhongzhan Huang, and Hong Zhang. Stiffness-aware neural network for learning hamiltonian systems. In *International Conference on Learning Representations*, 2021.

[40] Hanxiao Liu, Karen Simonyan, and Yiming Yang. Darts: Differentiable architecture search. *arXiv preprint arXiv:1806.09055*, 2018.

[41] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[42] Yiping Lu, Aoxiao Zhong, Quanzheng Li, and Bin Dong. Beyond finite layer neural networks: Bridging deep architectures and numerical differential equations. In Jennifer G. Dy and Andreas Krause, editors, *Proceedings of the 35th International Conference on Machine Learning, ICML 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3282–3291. PMLR, 2018.

[43] Xiaofeng Mao, Gege Qi, Yuefeng Chen, Xiaodan Li, Ranjie Duan, Shaokai Ye, Yuan He, and Hui Xue. Towards robust vision transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12042–12051, 2022.

[44] Laurent Meunier, Blaise J Delattre, Alexandre Araujo, and Alexandre Allauzen. A dynamical system perspective for lipschitz neural networks. In *International Conference on Machine Learning*, pages 15484–15500. PMLR, 2022.

[45] Jerome L Myers, Arnold D Well, and Robert F Lorch. *Research design and statistical analysis*. Routledge, 2013.

[46] Ben Niu, Weilei Wen, Wenqi Ren, Xiangde Zhang, Lianping Yang, Shuzhen Wang, Kaihao Zhang, Xiaochun Cao, and Haifeng Shen. Single image super-resolution via a holistic attention network. In *European conference on computer vision*, pages 191–207. Springer, 2020.

[47] Jongchan Park, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Bam: Bottleneck attention module. *arXiv preprint arXiv:1807.06514*, 2018.

[48] Adam Polyak and Lior Wolf. Channel-level acceleration of deep face representations. *IEEE Access*, 3:2163–2175, 2015.

[49] Jinghui Qin, Yongjie Huang, and Wushao Wen. Multiscale feature fusion residual network for single image super-resolution. *Neurocomputing*, 379:334–342, 2020.

[50] Zequn Qin, Pengyi Zhang, Fei Wu, and Xi Li. Fcanet: Frequency channel attention networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 783–792, 2021.

[51] Alejandro F Queiruga, N Benjamin Erichson, Dane Taylor, and Michael W Mahoney. Continuous-in-depth neural networks. *arXiv preprint arXiv:2008.02389*, 2020.

[52] Mengye Ren and Richard S Zemel. End-to-end instance segmentation with recurrent attention. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6656–6664, 2017.

[53] Lawrence F Shampine. *Numerical solution of ordinary differential equations*. Routledge, 2018.

[54] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017.

[55] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. Matching networks for one shot learning. In D. Lee, M. Sugiyama, U. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc., 2016.

[56] Qilong Wang, Banggu Wu, Pengfei Zhu, Peihua Li, Wangmeng Zuo, and Qinghua Hu. Eca-net: Efficient channel attention for deep convolutional neural networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

[57] Shuohang Wang, Luowei Zhou, Zhe Gan, Yen-Chun Chen, Yuwei Fang, Siqi Sun, Yu Cheng, and Jingjing Liu. Clusterformer: Clustering-based sparse transformer for long-range dependency encoding. *arXiv preprint arXiv:2009.06097*, 2020.

[58] Herman A Watts. Runge-kutta-fehlberg methods: scaled stability regions. Technical report, Sandia Labs., Albuquerque, N. Mex.(USA), 1975.

[59] E Weinan. A proposal on machine learning via dynamical systems. *Communications in Mathematics and Statistics*, 1(5):1–11, 2017.

[60] Sanghyun Woo, Jongchan Park, Joon-Young Lee, and In So Kweon. Cbam: Convolutional block attention module. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018.

[61] Saining Xie, Ross Girshick, Piotr Dollár, Zhuowen Tu, and Kaiming He. Aggregated residual transformations for deep neural networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1492–1500, 2017.

[62] Lu Yang, Qing Song, Yingqi Wu, and Mengjie Hu. Attention inspiring receptive-fields network for learning invariant representations. *IEEE transactions on neural networks and learning systems*, 30(6):1744–1755, 2018.

[63] Jiacheng Zhang, Zhicheng Zhao, and Fei Su. Efficient-receptive field block with group spatial attention mechanism for object detection. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 3248–3255. IEEE, 2021.

[64] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *Proceedings of the European conference on computer vision (ECCV)*, pages 286–301, 2018.

[65] Zhen Zhang, Shanghao Liu, Yang Zhang, and Wenbo Chen. Rs-darts: A convolutional neural architecture search for remote sensing image scene classification. *Remote Sensing*, 14(1):141, 2021.

[66] Shanshan Zhong, Wushao Wen, and Jinghui Qin. Switchable self-attention module. *arXiv preprint arXiv:2209.05680*, 2022.

[67] Lin Zhou, Haoming Cai, Jinjin Gu, Zheyuan Li, Yingqi Liu, Xiangyu Chen, Yu Qiao, and Chao Dong. Efficient image super-resolution using vast-receptive-field attention. *arXiv preprint arXiv:2210.05960*, 2022.

[68] Mai Zhu, Bo Chang, and Chong Fu. Convolutional neural networks combined with runge–kutta methods. *Neural Computing and Applications*, pages 1–15, 2022.

[69] Xiangyuan Zhu, Kehua Guo, Sheng Ren, Bin Hu, Min Hu, and Hui Fang. Lightweight image super-resolution with expectation-maximization attention mechanism. *IEEE Transactions on Circuits and Systems for Video Technology*, 32(3):1273–1284, 2021.