Linear Differential Vision Transformer: Learning Visual Contrasts via Pairwise Differentials

Yifan Pu[1*] **Jixuan Ying**[1*] **Qixiu Li**[1] **Tianzhu Ye**[1] **Dongchen Han**[1] **Xiaochen Wang**[2]
Ziyi Wang[1] **Xinyu Shao**[1] **Gao Huang**[1✉] **Xiu Li**[1✉]

[1] Tsinghua University [2] Peking University

Abstract

Vision Transformers (ViTs) have become a universal backbone for both image recognition and image generation. Yet their Multi–Head Self–Attention (MHSA) layer still performs a quadratic query–key interaction for *every* token pair, spending the bulk of computation on visually weak or redundant correlations. We introduce *Visual–Contrast Attention* (VCA), a drop-in replacement for MHSA that injects an explicit notion of discrimination while reducing the theoretical complexity from $\mathcal{O}(N^2C)$ to $\mathcal{O}(NnC)$ with $n \ll N$. VCA first distils each head's dense query field into a handful of spatially pooled *visual–contrast tokens*, then splits them into a learnable *positive* and *negative* stream whose differential interaction highlights what truly separates one region from another. The module adds fewer than 0.3 M parameters to a DeiT-Tiny backbone, requires no extra FLOPs, and is wholly architecture-agnostic. Empirically, VCA lifts DeiT-Tiny top-1 accuracy on ImageNet-1K from 72.2% to 75.6% (+3.4) and improves three strong hierarchical ViTs by up to 3.1%, while in class-conditional ImageNet generation it lowers FID-50K by 2.1 to 5.2 points across both diffusion (DiT) and flow (SiT) models. Extensive ablations confirm that (i) spatial pooling supplies low-variance global cues, (ii) dual positional embeddings are indispensable for contrastive reasoning, and (iii) combining the two in both stages yields the strongest synergy. VCA therefore offers a simple path towards faster and sharper Vision Transformers. The source code is available at https://github.com/LeapLabTHU/LinearDiff.

1 Introduction

Since the Vision Transformer (ViT) demonstrated that the same machinery that revolutionised natural language processing can match carefully designed CNNs on ImageNet [12], self attention has become a central ingredient of modern computer vision architectures. It now underpins recognition models (e.g., DeiT [65], Swin [43]), dense predictors, and even high-fidelity generators such as DiT [56, 61, 62, 97, 96]. Yet the way self attention is executed in vision has changed little from its language origin: for an image unfolded into N tokens every layer computes an $N \times N$ similarity matrix, leading to $\mathcal{O}(N^2C)$ multiplications and activations (C is the hidden width). With dozens of layers, quadratic self attention dominates both training and inference budgets, often forcing practitioners to shrink the patch size or the backbone depth and thus give up accuracy.

A first family of methods reduces the matrix size by limiting the receptive field: sliding windows [43], dilated blocks [30], stripes or criss–cross patterns [10] rely on the observation that many visual interactions are local. While they cut cost, they also prune long-range cues *a priori*, so they must juggle between speed and the ability to model global structures such as symmetry or repeated textures.

[*] Equal contribution. ✉ Corresponding authors.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

A second family keeps the global field of view but approximates the attention map with low-rank projections [72, 4] or fourier kernels [83]. These schemes are orthogonal to locality, yet they treat *all* correlations as equally useful. The network still has to wade through a sea of weak, often redundant similarities, which can drown signals and slow down convergence during training.

Inspired by recent progress in language modelling, *differential attention* [88] argues that the *difference* between two attention maps carries more discriminative signal than either map alone. Duplicating queries and keys and subtracting their softmaxes helps large language models focus on tokens that set one sentence apart from another, but the technique remains quadratic and ignores the particular redundancy structure of images. We start from a simple premise: it is better to compress the dense query field first and postpone any expensive comparison. Natural images exhibit spatial smoothness, which means neighbouring patches usually carry almost identical information. By leveraging this property, we can shrink the query set to just a handful of prototypes before matching. This idea materialises as Visual–Contrast Attention, a drop-in substitute for Multi-Head Self-Attention that injects an explicit notion of contrast and lowers the computational burden to $\mathcal{O}(Nnd)$ where $n \ll N$.

In the first stage, the model pools the scene for every attention head. Specifically, it average-pools the $H \times W$ query feature map to a coarse $h \times w$ grid (e.g., 8×8), flattens this grid into $n = hw$ visual-contrast tokens, and adds two distinct positional embeddings so that the tokens form a positive stream and a negative stream. Each stream attends independently to all keys and values; the two resulting outputs are subtracted and normalised, which produces a global contrast map that highlights the differences between two pooled views of identical content.

In the second stage, the module refines information at the patch level. Every one of the original N patch queries re-attends to the contrast map through a lightweight differential operation. Because the contrast map contains only n tokens, the three matrix multiplications that follow (query $\leftrightarrow$ contrast and contrast $\leftrightarrow$ value) scale with nN rather than N^2. Consequently, the module preserves the global receptive field characteristic of Vision Transformers, yet each attention weight now measures how much a patch stands out instead of merely capturing raw similarity.

This redesign yields three practical pay-offs in a single stroke. (i) Replacing quadratic MHSA with Visual Contrast Attention turns the per-layer complexity into a strictly linear form, shrinking both runtime and memory by roughly the ratio N/n, e.g., a 256^2 image patchified at 16×16 enjoys a 256× cut, without touching residual paths, layer norms, or any training hyper-parameters. (ii) The added machinery is tiny: each head only stores two n-dimensional positional embeddings, amounting to less than 0.3 M parameters on DeiT-Tiny and introducing essentially no new FLOPs because the contrast stage reuses existing key–value tensors. (iii) Because VCA dispenses with window masks, dilations, or kernel tricks, *any* ViT-style backbone that processes a 2-D patch grid can adopt it by swapping one block, leaving downstream decoders and pretrained heads entirely intact.

We validate these claims on two demanding tasks. For image classification, inserting VCA into a vanilla DeiT backbone raises ImageNet-1K top-1 accuracy from 72.2 % to 75.6 % without adding FLOPs, and integrating VCA into three hierarchical backbones (PVT [73], Swin [43], CSwin [10]) yields consistent gains of up to 3.1 percentage points. For image generation [18, 17, 54, 53, 53, 38, 45, 46], replacing the attention blocks in class-conditional generators lowers FID-50K on 256×256 ImageNet by 2.1 to 5.2 points across diffusion models (e.g., DiT [56]) and flow-based models (e.g., SiT [52]), across both Small and Base scales, and across patch sizes of 8, 4, and 2—again without extra compute.

Comprehensive ablation studies reinforce three *crucial* design choices. First, spatial pooling *reliably* supplies low-variance global cues. Second, the dual embeddings are *absolutely* essential for disentangling positive from negative evidence. Third, applying the pooled-plus-embedding recipe *symmetrically* to both streams in both stages *consistently* unlocks the full benefit of the method.

Our contributions are fourfold. First, we introduce Visual–Contrast Attention, which is the first linear-time attention module that embeds explicit contrast into Vision Transformers. Second, we provide a detailed complexity analysis verifying its linear computation complexity. Third, we demonstrate consistent accuracy and quality improvements in both image classification and image generation while keeping training budgets unchanged. Fourth, we show that VCA is architecture-agnostic, so it can serve as a drop-in upgrade for a wide range of Vision Transformer models.

In summary, Visual–Contrast Attention reconciles global reasoning with practical efficiency and offers a principled route toward faster and more descriptive Vision Transformers.

2 Related Work

Attention with Linear Complexity. A first group of studies attains linear time complexity by limiting the receptive field, such as Shifted-window attention [44] and Neighborhood Attention [31]. By re-introducing locality into the ViT architecture, these methods lower cost but partially sacrifice global context. A second research line tackles the problem directly with *linear* attention. The seminal work of [39] eliminates the Softmax and applies a feature map ϕ to Q and K, reducing complexity to $\mathcal{O}(N)$ at the cost of noticeable accuracy loss. Follow-ups proposed better approximations: Efficient Attention [64] applies Softmax separately to Q and K; SOFT [50] and Nyströmformer [83] rely on matrix decompositions; Castling-ViT [89] uses full Softmax only as an auxiliary during training; FLatten Transformer [19] introduces a focus function and depth-wise convolutions to enrich features. MLLA [21] incorporate the key design in Mamba into linear attention, while InLine [20] introduces an injective linear attention mechanism. More recently, Agent Attention [22], Anchored Stripe Attention [41], and Efficient DiT [61] insert an extra token set that mediates between queries and keys, an equivalent linearization that attains strong results for recognition and low-level vision. Our work is built on this architecture, interpreting the additional visual contrast tokens as a semantic compression.

Vision Transformer. Since the arrival of the Vision Transformer (ViT) [11], self-attention has flourished in computer vision, yet the quadratic cost of conventional Softmax attention [66] remains a hurdle. Numerous remedies have been proposed. PVT [73] sparsifies global attention by down-sampling K and V; Swin [43] confines attention to local windows and shifts them to cover the whole image; NAT [32] mimics convolution by attending within each feature's neighborhood; DAT [80] introduces deformable, data-dependent patterns; BiFormer [101] routes queries to salient regions via a bi-level scheme; GRL [42] mixes stripe, window, and channel attentions for restoration. Nevertheless, these strategies either cap the global receptive field or are tailored to specific patterns, which limits their plug-and-play versatility.

Diffusion Transformer. State-of-the-art diffusion models [9, 1, 33, 47, 16, 86] are traditionally built on U-Net [63], yet recent works explore ViT backbones [87, 56, 2]. U-ViT [2] tokenizes time, conditions, and noisy patches, adding U-Net–style skip connections. DiT [56] shows ViT scales favorably, outperforming U-Net on ImageNet; SiT [52] extends DiT to continuous time with more general coefficients. MaskDiT [99] adopts masked training to cut cost, while MDT [14] and MDTv2 refine masked latent modeling for better FID and faster learning. HDiT [5] trains high resolutions with cost linear in pixel count. FiT [99] treats images as variable-sized token sequences, enabling flexible resolution and aspect ratios. These results verify that transformer backbones are both effective and scalable for generative diffusion, yet their internal architectural choices remain under-explored.

Dynamic Neural Network. Unlike static networks with fixed graphs and weights, dynamic neural networks [24, 75] adapt structure or parameters per input, gaining advantages in accuracy, adaptability [85, 15, 95, 98], efficiency [84, 68, 77, 91], and representation capacity [60]. They are commonly classified as sample-wise [13, 34, 76, 27, 23, 58, 69, 79], spatial-wise [70, 35, 28, 26, 25, 81, 82, 55, 78], and temporal-wise [29, 74, 51]. Following DETR's query paradigm [3], a query-based dynamic branch has also emerged [59]. Our method could be classified as a type of sample-wise dynamic network, since different sample generate different visual contrasts token in the first stage.

3 Approach

3.1 Preliminaries

Standard Attention in Vision. We first revisit the attention mechanism [67] in Vision Transformers [2] [12, 56, 52]. The Vision Transformer takes a visual token sequence $\boldsymbol{z}_{l-1} \in \mathbb{R}^{N \times C}$ from the previous layer $l-1$ as input (N is the token number and C is the hidden dimension), then projects it into the query, key, and value token sequences with three different linear projection layers, denoted as

[2]Throughout the paper we use **Vision Transformer (ViT)** to collectively denote the original ViT [12] and its diffusion variant Diffusion Transformer (DiT) [56]. A DiT block is identical to a ViT block except that each LayerNorm is augmented with timestep-conditioned scale and shift parameters (AdaLN) generated from the diffusion timestep embedding. This distinction is orthogonal to our contribution.

$\mathbf{W^q}, \mathbf{W^k}, \mathbf{W^v} \in \mathbb{R}^{C\times C}$ (we omit the bias term for simplicity):

$$\boldsymbol{q} = \boldsymbol{z}_{l-1}\mathbf{W^q},\ \ \boldsymbol{k} = \boldsymbol{z}_{l-1}\mathbf{W^k},\ \ \boldsymbol{v} = \boldsymbol{z}_{l-1}\mathbf{W^v}. \tag{1}$$

Then $\boldsymbol{q}, \boldsymbol{k}, \boldsymbol{v} \in \mathbb{R}^{N\times C}$ are divided into M heads $\boldsymbol{q}^{(m)}, \boldsymbol{k}^{(m)}, \boldsymbol{v}^{(m)} \in \mathbb{R}^{N\times d}$ in terms of channel C, with head dimension of $d{=}C/M$ (C is always divisible by M). Within each head, the similarity of each query $\boldsymbol{q}^{(m)}$ and key $\boldsymbol{k}^{(m)}$ is computed as:

$$\mathbf{A}^{(m)} = \text{Softmax}\left(\boldsymbol{q}^{(m)}\boldsymbol{k}^{(m)\top}/\sqrt{d}\right), \tag{2}$$

where the attention map $\mathbf{A}^{(m)}$ is an $N{\times}N$ matrix containing elements in the range $[0, 1]$, and the sum of each row is normalized to 1. The attention mechanism reweights the value sequence according to the attention map, $\boldsymbol{h}^{(m)} = \mathbf{A}^{(m)}\boldsymbol{v}^{(m)} \in \mathbb{R}^{N\times d}$, to dynamically adjust the outputs based on the dependency of each token in the inputs. In the end, each head of the reweighted representation is concatenated together to produce the final output of this layer l, written as:

$$\boldsymbol{h} = \text{Concat}\left(\boldsymbol{h}^{(1)}, \ldots, \boldsymbol{h}^{(M)}\right), \qquad \boldsymbol{z}_l = \boldsymbol{h}\mathbf{W^O}. \tag{3}$$

where $\boldsymbol{h}{\in}\mathbb{R}^{N\times C}$, $\mathbf{W^O}{\in}\mathbb{R}^{C\times C}$ (the bias term is also omitted for simplicity) is a linear projection layer to promote interaction between different heads in the multi-head attention layer.

Differential Attention in Language. We further revisit the recent proposed differential attention mechanism which is primarily used in language modeling [88]. Given the token squences from the previous layer $\boldsymbol{z}_{l-1}{\in}\mathbb{R}^{N\times C}$ (N tokens, each with a hidden dimension C), differential attention first produces *two* sets of queries and keys ($\{\boldsymbol{q}_1, \boldsymbol{k}_1\}$ and $\{\boldsymbol{q}_2, \boldsymbol{k}_2\}$) with two sets of linear projections ($\{\mathbf{W}_1^\mathbf{q}, \mathbf{W}_1^\mathbf{k}\}$ and $\{\mathbf{W}_2^\mathbf{q}, \mathbf{W}_2^\mathbf{k}\}$) and *one* set of values $\boldsymbol{v}$ with a linear projection $\mathbf{W^v}$:

$$\left[\boldsymbol{q}_1; \boldsymbol{q}_2\right] = \boldsymbol{z}_{l-1}\left[\mathbf{W}_1^\mathbf{q}; \mathbf{W}_2^\mathbf{q}\right], \quad \left[\boldsymbol{k}_1; \boldsymbol{k}_2\right] = \boldsymbol{z}_{l-1}\left[\mathbf{W}_1^\mathbf{k}; \mathbf{W}_2^\mathbf{k}\right], \quad \boldsymbol{v} = \boldsymbol{z}_{l-1}\mathbf{W^v}, \tag{4}$$

where $\mathbf{W}_1^\mathbf{q}, \mathbf{W}_2^\mathbf{q}, \mathbf{W}_1^\mathbf{k}, \mathbf{W}_2^\mathbf{k} \in \mathbb{R}^{(C/2)\times C}$, $\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{k}_1, \boldsymbol{k}_2 \in \mathbb{R}^{N\times(C/2)}$,$\mathbf{W^v} \in \mathbb{R}^{C\times C}$,$\boldsymbol{v} \in \mathbb{R}^{N\times C}$. Then $\boldsymbol{q}_1, \boldsymbol{q}_2, \boldsymbol{k}_1, \boldsymbol{k}_2, \boldsymbol{v}$ are divided into M heads $\boldsymbol{q}_1^{(m)}, \boldsymbol{q}_2^{(m)}, \boldsymbol{k}_1^{(m)}, \boldsymbol{k}_2^{(m)} \in \mathbb{R}^{N\times(d/2)}$, $\boldsymbol{v}^{(m)} \in \mathbb{R}^{N\times d}$, with (double) head dimension of $d{=}C/M$. Within each head, we compute two attention maps

$$\mathbf{A}_1^{(m)} = \text{Softmax}\left(\boldsymbol{q}_1^{(m)}\boldsymbol{k}_1^{(m)\top}/\sqrt{d/2}\right), \qquad \mathbf{A}_2^{(m)} = \text{Softmax}\left(\boldsymbol{q}_2^{(m)}\boldsymbol{k}_2^{(m)\top}/\sqrt{d/2}\right), \tag{5}$$

and take their *difference* as the final attention weight of each head:

$$\mathbf{A}^{(m)} = \mathbf{A}_1^{(m)} - \lambda\,\mathbf{A}_2^{(m)}, \qquad \lambda = \exp(\boldsymbol{\lambda}_{q_1}{\cdot}\boldsymbol{\lambda}_{k_1}) - \exp(\boldsymbol{\lambda}_{q_2}{\cdot}\boldsymbol{\lambda}_{k_2}) + \lambda_{\text{init}}. \tag{6}$$

Here λ is a learnable scalar parameterized by a scalar λ_{init} and vectors $\lambda_{\boldsymbol{q}_1}, \lambda_{\boldsymbol{q}_2}, \lambda_{\boldsymbol{k}_1}, \lambda_{\boldsymbol{k}_2} \in \mathbb{R}^d$. The attention map $\mathbf{A}^{(m)}$ is an $N \times N$ matrix. The attention mechanism reweights the value sequence according to the attention map followed by a $\mathrm{RMSNorm}$ Layer, and scaled by $(1 - \lambda_{\text{init}})$ to match Transformer's gradient flow:

$$\hat{\boldsymbol{h}}^{(m)} = \mathbf{A}^{(m)}\boldsymbol{v}^{(m)}, \qquad \boldsymbol{h}^{(m)} = (1 - \lambda_{\text{init}})\ \mathrm{RMSNorm}(\hat{\boldsymbol{h}}^{(m)}), \tag{7}$$

where $\boldsymbol{h}^{(m)}, \hat{\boldsymbol{h}}^{(m)} \in \mathbb{R}^{N\times d}$. In the end, each head of the reweighted representation is concatenated together to produce the final output of this layer l, written as:

$$\boldsymbol{h} = \text{Concat}\left(\boldsymbol{h}^{(1)}, \ldots, \boldsymbol{h}^{(M)}\right), \qquad \boldsymbol{z}_l = \boldsymbol{h}\mathbf{W^O}. \tag{8}$$

where $\boldsymbol{h}{\in}\mathbb{R}^{N\times C}$, $\mathbf{W^O}{\in}\mathbb{R}^{C\times C}$ (the bias term is also omitted for simplicity) is a linear projection layer to promote interaction between different heads in the multi-head attention layer.

3.2 Our Approach

Visual Contrast Attention. To extend differential attention to vision *and* trim the quadratic complexity, besides the query $\boldsymbol{q}^{(m)}$, key $\boldsymbol{k}^{(m)}$, and value $\boldsymbol{v}^{(m)}$ tokens, we introduce a compact pair of *visual contrast tokens* for each head: a set of positive visual contrast tokens $\boldsymbol{t}_+^{(m)}$ and a set of negative visual contrast tokens $\boldsymbol{t}_-^{(m)}$. These pair of visual contrast tokens are both with a $n \times d$ shape, where

n is the visual contrast token length and $n \ll N$. Intuitively, the two sets act as the same mediator token viewed through two coloured lenses. Stage I lets the two sets skim the whole image and return a *contrast map*; Stage II lets the original patch queries exploit that map.

Stage I – global contrast. This pair of visual contrast tokens first each attends to all key tokens and all value tokens individually to get the intermediate results $\hat{\boldsymbol{v}}_{+}^{(m)}$, $\hat{\boldsymbol{v}}_{-}^{(m)}$:

$$\hat{\boldsymbol{v}}_{+}^{(m)} = \text{Softmax}\left(\boldsymbol{t}_{+}^{(m)}\boldsymbol{k}^{(m)\top}/\sqrt{d}\right)\boldsymbol{v}^{(m)}, \quad \hat{\boldsymbol{v}}_{-}^{(m)} = \text{Softmax}\left(\boldsymbol{t}_{-}^{(m)}\boldsymbol{k}^{(m)\top}/\sqrt{d}\right)\boldsymbol{v}^{(m)}, \tag{9}$$

where $\hat{\boldsymbol{v}}_{+}^{(m)}, \hat{\boldsymbol{v}}_{-}^{(m)} \in \mathbb{R}^{n\times d}$. Then the visual contrast results is obtained by performing a differential operation in the intermediate results, followed by a $\mathrm{RMSNorm}$ and a $(1-\lambda_{\text{init}}^{(1)})$ scalar factor:

$$\hat{\boldsymbol{v}}^{(m)} = (1-\lambda_{\text{init}}^{(1)})\,\mathrm{RMSNorm}\left(\hat{\boldsymbol{v}}_{+}^{(m)} - \lambda^{(1)}\hat{\boldsymbol{v}}_{-}^{(m)}\right), \lambda^{(1)} = \exp(\boldsymbol{\lambda}_{q_1}^{(1)}\cdot\boldsymbol{\lambda}_{k_1}^{(1)}) - \exp(\boldsymbol{\lambda}_{q_2}^{(1)}\cdot\boldsymbol{\lambda}_{k_2}^{(1)}) + \lambda_{\text{init}}^{(1)}. \tag{10}$$

Stage II – patch-wise differential attention. The pair of visual contrast tokens interact with the query tokens and extracts the results from the intermediate result $\hat{\boldsymbol{v}}^{(m)}$ in a differential way. To be specific, the query tokens $\boldsymbol{q}^{(m)}$ derive its attention scores with both the positive visual contrast tokens $\boldsymbol{t}_{+}^{(m)}$ and the negative ones $\boldsymbol{t}_{-}^{(m)}$ within each head:

$$\mathbf{A}_1^{(m)} = \text{Softmax}\left(\boldsymbol{q}^{(m)}\boldsymbol{t}_{+}^{(m)\top}/\sqrt{d}\right), \qquad \mathbf{A}_2^{(m)} = \text{Softmax}\left(\boldsymbol{q}^{(m)}\boldsymbol{t}_{-}^{(m)\top}/\sqrt{d}\right), \tag{11}$$

and take their *difference* as the final attention weight of each head:

$$\mathbf{A}^{(m)} = \mathbf{A}_1^{(m)} - \lambda^{(2)}\,\mathbf{A}_2^{(m)}, \qquad \lambda^{(2)} = \exp(\boldsymbol{\lambda}_{q_1}^{(2)}\cdot\boldsymbol{\lambda}_{k_1}^{(2)}) - \exp(\boldsymbol{\lambda}_{q_2}^{(2)}\cdot\boldsymbol{\lambda}_{k_2}^{(2)}) + \lambda_{\text{init}}^{(2)}, \tag{12}$$

where $\lambda^{(2)}$ follows the same parameterisation as $\lambda^{(1)}$. The attention map $\mathbf{A}^{(m)}$ is an $N\times n$ matrix. The attention mechanism reweights the value sequence according to the attention map followed by a $\mathrm{RMSNorm}$ Layer, and scaled by $(1-\lambda_{\text{init}}^{(2)})$ to match Transformer's gradient flow:

$$\hat{\boldsymbol{h}}^{(m)} = \mathbf{A}^{(m)}\hat{\boldsymbol{v}}^{(m)}, \qquad \boldsymbol{h}^{(m)} = (1-\lambda_{\text{init}}^{(2)})\ \mathrm{RMSNorm}(\hat{\boldsymbol{h}}^{(m)}), \tag{13}$$

where $\boldsymbol{h}^{(m)}, \hat{\boldsymbol{h}}^{(m)} \in \mathbb{R}^{N\times d}$. In the end, each head of the reweighted representation is concatenated together to produce the final output of this layer l, written as:

$$\boldsymbol{h} = \text{Concat}\left(\boldsymbol{h}^{(1)}, \ldots, \boldsymbol{h}^{(M)}\right), \qquad \boldsymbol{z}_l = \boldsymbol{h}\mathbf{W}^{\mathbf{O}}. \tag{14}$$

where $\boldsymbol{h} \in \mathbb{R}^{N\times C}$, $\mathbf{W}^{\mathbf{O}} \in \mathbb{R}^{C\times C}$ (the bias term is also omitted for simplicity) is a linear projection layer to promote interaction between different heads in the multi-head attention layer.

Visual Contrast Token Generation The visual contrast tokens are distilled directly from the query tokens through spatial average pooling. Since our attention module operates on visual latent features, the query matrix $\boldsymbol{q}^{(m)} \in \mathbb{R}^{N\times d}$ can be rearranged back to its 2-D spatial layout, i.e., $\boldsymbol{q}^{(m)} \to \tilde{\boldsymbol{q}}^{(m)} \in \mathbb{R}^{H\times W\times d}$ with $H\times W = N$. We then apply average pooling along the spatial dimensions, with kernel size and stride chosen to reduce the resolution from $H\times W$ to $h\times w$:

$$\tilde{\boldsymbol{t}}^{(m)} = \text{AvgPool}\left(\tilde{\boldsymbol{q}}^{(m)}\right) \ \in \ \mathbb{R}^{h\times w\times d}. \tag{15}$$

To further disentangle helpful and distracting correlations, we split the visual contrast branch into a *positive* and a *negative* stream, inspired by the core idea of Differential Transformer. Specifically, we add two distinct learnable positional embeddings, $\boldsymbol{e}^{+}, \boldsymbol{e}^{-} \in \mathbb{R}^{h\times w\times d}$, to create two groups of visual contrast tokens:

$$\tilde{\boldsymbol{t}}_{+}^{(m)} = \tilde{\boldsymbol{t}}^{(m)} + \boldsymbol{e}^{+}, \qquad \tilde{\boldsymbol{t}}_{-}^{(m)} = \tilde{\boldsymbol{t}}^{(m)} + \boldsymbol{e}^{-}. \tag{16}$$

Finally, we flatten the positive and negative tensors over the spatial axes to obtain the visual contrast token matrices:

$$\boldsymbol{t}_{+}^{(m)}, \boldsymbol{t}_{-}^{(m)} = \text{Flatten}\left(\tilde{\boldsymbol{t}}_{+}^{(m)}\right), \text{Flatten}\left(\tilde{\boldsymbol{t}}_{-}^{(m)}\right) \in \mathbb{R}^{n \times d}, \quad n = h \cdot w \ll N. \tag{17}$$

Each visual contrast token thus represents the average feature of a non-overlapping image patch, providing a compact summary. The target spatial size (h, w)—and thus the number of visual contrast tokens n—is a tunable hyper-parameter that balances computational cost and representational fidelity.

3.3 Complexity Analysis

We retain the notations of Section 3.2: N visual tokens, n visual contrast tokens ($n \ll N$), d features per head, M heads and $C = Md$ channels in total.

Stage I – global contrast. For each head, the positive and negative contrast tokens execute the two attention steps in Equation (9):

$$\underbrace{\text{Softmax}\big(\boldsymbol{t}_{\pm}^{(m)} \boldsymbol{k}^{(m)\top} / \sqrt{d}\big)}_{\mathbb{R}^{n \times N}} \underbrace{\boldsymbol{v}^{(m)}}_{\mathbb{R}^{N \times d}} \longrightarrow \hat{\boldsymbol{v}}_{\pm}^{(m)} \in \mathbb{R}^{n \times d}. \tag{18}$$

Each of the two matrix products ($n \times d$ by $d \times N$, then $n \times N$ by $N \times d$) costs $\mathcal{O}(Nnd)$; performing them for both "$+$" and "$-$" streams therefore costs at most

$$\text{Stage I:} \qquad 4\,Nnd \;=\; \mathcal{O}(Nnd). \tag{19}$$

The subsequent subtraction, normalisation, and scalar modulation are all $\mathcal{O}(nd)$ and thus negligible in the big-$\mathcal{O}$ sense.

Stage II – patch-wise differential attention. In each differential stream, we calculate the two attention map in Equation (11):

$$\text{Softmax}\Big(\boldsymbol{q}^{(m)}\, \boldsymbol{t}_{\pm}^{(m)\top} / \sqrt{d}\Big) \quad \in \mathbb{R}^{N \times n}. \tag{20}$$

Each map requires one $N \times d$ by $d \times n$ multiply, i.e. $\mathcal{O}(Nnd)$. Both maps together give a cost of $2\,\mathcal{O}(Nnd)$. The differential combination $\mathbf{A}^{(m)} = \mathbf{A}_1^{(m)} - \lambda^{(2)} \mathbf{A}_2^{(m)}$ is only $\mathcal{O}(Nn)$.

Finally, value aggregation in Equation (13) multiplies an $N \times n$ matrix by an $n \times d$ matrix, adding another $\mathcal{O}(Nnd)$. Hence

$$\text{Stage II:} \qquad 3\,Nnd \;=\; \mathcal{O}(Nnd). \tag{21}$$

Per-head and per-layer complexity. Both stages are linear in N, n, and d. For one head

$$\mathcal{C}_{\text{head}} = \mathcal{O}(Nnd), \tag{22}$$

and for the whole layer

$$\mathcal{C}_{\text{layer}} = M\,\mathcal{C}_{\text{head}} = \mathcal{O}(NnC). \tag{23}$$

Comparison with vanilla self-attention. Standard self-attention forms an $N \times N$ attention map, incurring $\mathcal{O}(N^2 C)$ time. Replacing the quadratic query–key interaction by the two linear query–contrast *and* contrast–key interactions reduces the cost by a factor of N/n:

$$\frac{\mathcal{O}(N^2 C)}{\mathcal{O}(NnC)} = \frac{N}{n} \gg 1. \tag{24}$$

Because $n \ll N$, the proposed visual-contrast attention substantially lowers computation while preserving global context. Overheads from RMS normalisation and the learnable scalars are at most $\mathcal{O}(Nd)$ or $\mathcal{O}(nd)$ and are therefore non-dominant.

Table 1: Image classification results on ImageNet-1K

Method	#Params	FLOPs	Top-1 Acc.
DeiT-T	5.7M	1.2G	72.2
+Ours	6.0M	1.2G	**75.6**$_{(\uparrow \mathbf{3.4})}$
DeiT-S	22.1M	4.6G	79.8
+Ours	22.6M	4.6G	**80.7**$_{(\uparrow \mathbf{0.9})}$
PVT-T	13.2M	1.9G	75.1
+Ours	11.6M	2.0G	**78.2**$_{(\uparrow \mathbf{3.1})}$
PVT-S	24.5M	3.8G	79.8
+Ours	20.6M	4.1G	**82.3**$_{(\uparrow \mathbf{2.5})}$
PVT-M	35.9M	7.0G	81.2
+Ours	35.8M	7.2G	**83.2**$_{(\uparrow \mathbf{2.0})}$

Method	#Params	FLOPs	Top-1 Acc.
Swin-T	28.9M	4.5G	81.3
+Ours	28.5M	4.6G	**82.3**$_{(\uparrow \mathbf{1.0})}$
Swin-S	49.7M	8.7G	83.0
+Ours	49.6M	8.7G	**83.7**$_{(\uparrow \mathbf{0.7})}$
Swin-B	88.1M	15.4G	83.5
+Ours	87.9M	15.5G	**83.9**$_{(\uparrow \mathbf{0.4})}$
CSwin-T	20.5M	4.3G	82.7
+Ours	20.4M	4.3G	**83.3**$_{(\uparrow \mathbf{0.6})}$
CSwin-S	32.8M	6.8G	83.6
+Ours	32.7M	6.8G	**84.0**$_{(\uparrow \mathbf{0.4})}$

4 Experiments

In section, we empirically evaluate our visual contrasts attention method on both image recognition and generation tasks. We first introduce the detailed experiment setup in Section 4.1, including datasets and training configurations. Then the main results of our method with various backbone architectures on different tasks are presented in Section 4.2 and Section 4.3. Finally, the ablation study in Section 4.4 further validate the effectiveness of the proposed method.

4.1 Experiment settings

Datasets The ImageNet-1K [7] recognition dataset contains 1.28M training images and 50K validation images with a total of 1,000 classes. For image recognition experiments, images are trained and evaluated in 224×224 size. The top-1 accuracy on the validation set is adopted as the evaluation metric. For image generation tasks, we train and evaluate the images in 256×256 size, following the commonly used practice in class-condition generation. We use FID-50K as the evaluation metric, which measures the Fréchet distance between the Inception-V3 features of 50 000 generated images and 50 000 real validation images.

Training Configuration For image recognition experiments, we use the same training setup as the baseline models to ensure fair comparison. All models are trained from scratch using the AdamW [48] optimizer for 300 epochs. We apply cosine learning rate decay, starting with 20 epochs of linear warm-up, and set the initial learning rate to 1×10^{-3} with a weight decay of 0.05. The data augmentation and regularization methods include RandAugment [6], Mixup [93], CutMix [92], and random erasing [100]. We also follow CSwin [10] and use EMA [57] during training. For image generation tasks, we follow DiT [56] and SiT [52] to train class-conditional diffusion transformer models on the ImageNet-1K [8] dataset. All models are trained with the AdamW [40, 49] optimizer and no weight decay. For 256×256 resolution, we train from scratch with a global batch size of 256 for $400{,}000$ iterations. The learning rate is kept constant at 1×10^{-4}. We use only random horizontal flip for data augmentation during training. Additionally, we apply exponential moving average (EMA) to the model weights with a decay rate of 0.9999.

4.2 Image recognition

The image recognition experiments are conducted on ImageNet-1K [7] dataset. We conduct ImageNet classification on both plain vision transformer architectures (e.g., DeiT [65]) and hierarchical counterparts (e.g., PVT [65], Swin [43], CSwin [10]). On the *plain* Vision Transformer line, as is illustrated in Table 1, our method consistently enlarges the accuracy–efficiency Pareto front: DeiT-Tiny takes a $+3.3$ accuracy gain (from $72.2\,\%$ to $75.6\,\%$) with only $0.3\,\mathrm{M}$ additional parameters and no extra computational cost, while DeiT-Small still enjoys a +0.9 performance improvement under the same computational budget. For *hierarchical* vision transformer architectures, which include the multi-stage PVT, the shifted-window Swin, and the cross-shaped CSwin architectures, the proposed block remains universally beneficial. PVT experiences the largest margins, up to +3.1 percentage point on PVT-Tiny and +2.5/+2.0 on PVT-Small/PVT-Medium, respectively. On more

Table 2: Class-conditional image generation results on ImageNet-1K with 256×256 resolution.

Method	#Params	FLOPs	FID-50K(↓)	Method	#Params	FLOPs	FID-50K(↓)
DiT-S/8	33.0M	0.4G	151.9	SiT-S/8	33.0M	0.4G	149.5
+Ours	33.8M	0.4G	**148.3**$_{(\downarrow 3.6)}$	+Ours	33.0M	0.4G	**147.4**$_{(\downarrow 2.1)}$
DiT-S/4	32.9M	1.4G	97.9	SiT-S/4	32.9M	1.4G	84.0
+Ours	33.6M	1.5G	**92.7**$_{(\downarrow 5.2)}$	+Ours	33.6M	1.5G	**80.9**$_{(\downarrow 3.1)}$
DiT-S/2	33.0M	6.1G	67.2	SiT-S/2	33.0M	6.1G	57.3
+Ours	33.6M	6.0G	**62.3**$_{(\downarrow 4.9)}$	+Ours	33.6M	6.0G	**53.0**$_{(\downarrow 4.3)}$
DiT-B/8	130.7M	1.4G	118.4	SiT-B/8	130.7M	1.4G	106.0
+Ours	132.1M	1.5G	**114.4**$_{(\downarrow 4.0)}$	+Ours	132.1M	1.5G	**102.1**$_{(\downarrow 3.9)}$
DiT-B/4	130.4M	5.6G	68.3	SiT-B/4	130.4M	5.6G	55.9
+Ours	131.8M	5.8G	**66.0**$_{(\downarrow 2.3)}$	+Ours	131.8M	5.8G	**53.6**$_{(\downarrow 2.3)}$
DiT-B/2	130.5M	23.0G	42.9	SiT-B/2	130.5M	23.0G	35.3
+Ours	131.8M	22.9G	**38.9**$_{(\downarrow 4.0)}$	+Ours	131.8M	22.9G	**32.7**$_{(\downarrow 2.6)}$

strong baselines like Swin and CSwin, our method still receive steady gains between +0.4 and +1.0 with negligible ($< 5\%$) overhead. These results demonstrate that the proposed visual contracts attention is architecture-agnostic: it complements both global self-attention in plain ViTs and the localized or cross-shaped attention patterns employed by state-of-the-art hierarchical designs.

4.3 Image generation

We evaluate our approach on class–conditional ImageNet-1K image generation at 256×256 resolution, taking the diffusion–based DiT [56] family and the flow–based SiT [52] family as baselines. For each backbone we consider two model sizes, *Small* ($\sim$33 M parameters) and *Base* ($\sim$131 M), and three different patch size (8, 4 and 2), which together sweep a wide range of computation budgets from 0.4 G to 23.0 GFLOPs. All networks are trained under the original recipes released by the authors: DiT models follow the DDPM [33] schedule with 1 000 denoising steps, while SiT counterparts are optimized with the latent flow objective; the only change is that we replace the original attention with the proposed visual contrast attention, adding fewer than 1.3 M parameters and at most 0.1 GFLOPs. Following standard protocol we report Fréchet Inception Distance on 50 000 validation samples (FID-50K), computed with the same code base as DiT [56] paper.

As summarized in Table 2, our method consistently lowers FID across various configuration. Along the *model-size axis*, the absolute gains on Small backbones reach 3.6 to 5.2 points for DiT and 2.1 to 4.3 points for SiT, whereas Base models still benefit by 2.3 to 4.0 and 2.3 to 3.9 points respectively, indicating diminishing but non-negligible returns as capacity grows. Along the *patch-resolution axis*, the most fine-grained patches (e.g., /2) exhibit the largest relative improvement (up to 4.9 gain in FID in DiT-S/2), yet even the largest variants (e.g., /8) obtain solid reductions in FID-50K of 2.1 to 4.0. Finally, comparing the two *training paradigms* (e.g., diffusion-based, flow-based), we observe that the proposed module is agnostic to the underlying generative mechanism: it offers similar FID reductions for the DDPM pipeline of DiT and the rectified flow pipeline of SiT, thereby confirming its general applicability to both diffusion and flow-based training configurations.

4.4 Ablation Studies

Ablation on Detailed Model Architectures. We first investigates *where* the standard Multi-Head Self-Attention(MHSA) are replaced by our modified counterparts, incluing the first attention operation in Stage I (global contrast) and that in Stage II (patch-wise differential attention) in all the attention blocks. We also ablate the result by using the original differential attention [88] in both stages. We conduct the ablation studies both on image classification with DeiT-Tiny and image generation task with DiT-S/2 model. The quantitative results in Table 3 reveal three clear tendencies. First, the two components of our *Visual-Contrast Attention* contribute additively. Activating only the **Stage-I global-contrast** branch raises DeiT-Tiny accuracy from the implicit vanilla baseline to 75.4 % and reduces the DiT-S/2 FID to 64.6, while switching on only the **Stage-II patch-wise differential** branch is slightly more effective (75.5 % / 64.3). When the two branches are combined, their effects accumulate almost linearly, pushing the score to 75.6 % and 62.3 FID without introducing any extra

Table 3: Ablation on detailed model architectures across image classification and generation tasks.

Attention Type		Image Classification on DeiT-Tiny			Image Generation on DiT-S/2		
Stage I	Stage II	Params	FLOPs	Top-1 Acc.(↑)	Params	FLOPs	FID-50K(↓)
Ours	Vani.	6.0M	1.2G	75.5	33.6M	5.9G	64.3
Vani.	Ours	6.0M	1.2G	75.4	33.6M	5.9G	64.6
Diff.	Diff.	5.7M	1.2G	75.1	33.0M	5.8G	63.9
Ours	Ours	6.0M	1.2G	75.6	33.6M	6.0G	62.3

Table 4: Ablation on visual contrast token generation across image classification and generation tasks.

Token Type		Image Classification on DeiT-Tiny			Image Generation on DiT-S/2		
Pos. Str.	Neg. Str.	Params	FLOPs	Top-1 Acc.	Params	FLOPs	FID-50K
Emb.	Emb.	6.0M	1.2G	75.1	33.6M	6.0G	63.7
Pool	Pool+Emb.	5.9M	1.2G	75.5	33.3M	6.0G	64.1
Pool+Emb.	Pool	5.9M	1.2G	75.3	33.3M	6.0G	63.5
Pool+Emb.	Pool+Emb.	6.0M	1.2G	75.6	33.6M	6.0G	62.3

FLOPs and with only $\sim$0.3 M additional parameters. Second, we compare VCA with the language-oriented *differential attention (Diff)* [88] applied to both stages. Although Diff already improves over the single-branch variants (75.1 % / 63.9), replacing it with our vision-tailored VCA brings a further relative gain of $+0.5$ percentage points in classification accuracy and a -1.8 improvement in FID. This superiority indicates that (i) summarising the scene with a small set of learnable visual-contrast tokens in Stage I and (ii) letting Stage II queries interact with those tokens in a differential manner are both crucial for vision, and that the proposed formulation exploits their synergy more effectively than simply duplicating the original differential attention design.

Ablation on Visual–Contrast Token Generation. Table 4 investigates how the two visual–contrast streams that feed the subsequent differential operation should be formed. Each stream can be a pure learnable embedding (EMB.), a query representation obtained by spatial average pooling (POOL), or the pooled query features augmented with an independent positional embedding (POOL+EMB.) that is adopted in our final model. Using embeddings for *both* the positive and negative streams already gives a noticeable improvement over the vanilla backbone (75.1 % Top-1 Accuracy / 63.7 FID-50K), confirming that explicit differencing is helpful even with the same randomly initialised tokens for different input images. Replacing the positive stream by pooled queries while leaving the negative one unchanged (POOL / POOL+EMB.) yields a marginal additional gain (75.5 % / 64.1), whereas performing the opposite substitution (POOL+EMB. / POOL) produces a larger jump to 75.3 % and 62.3, suggesting that injecting real image statistics into the positive branch is more influential. When *both* streams adopt the full POOL+EMB. recipe, performance peaks at 75.6 % and 62.3, outperforming the embedding-only variant by +0.5 percentage points and –1.4 FID with no additional parameters and identical FLOPs. These results demonstrate that (i) spatial pooling supplies informative, low-variance global cues, (ii) separate positional embeddings remain essential for disentangling complementary correlations, and (iii) combining the two ingredients for *both* streams yields the strongest synergy across classification and generation tasks.

5 Conclusion

We have presented Visual–Contrast Attention, a plug-and-play replacement for MHSA that couples linear complexity with an explicit notion of discrimination. By first summarising the image into a handful of pooled tokens and then splitting these tokens into antagonistic positive/negative streams, VCA highlights genuinely informative relationships while discarding redundant ones. The module is parameter-light, budget-neutral in FLOPs, and universally beneficial: it boosts classification accuracy across plain and hierarchical ViTs, and further sharpens generative quality in both diffusion- and flow-based models. We hope our findings encourage the community to rethink attention not only as a similarity measure but also as a stage for explicit contrast.

Limitations

VCA reduces the quadratic burden of self-attention but is not a cure-all: (i) task-agnostic average pooling may miss edge-rich details; (ii) the added micro-attention may shrinks speed gains on small images; (iii) extensions to video [38], 3-D [36, 37], or more efficient language [90, 94, 71] tasks are still unexplored.

Acknowledgement

This work is supported in part by the National Key R&D Program of China under Grant 2024YFB4708200, the National Natural Science Foundation of China under Grants U24B20173 and 42327901, and the Scientific Research Innovation Capability Support Project for Young Faculty under Grant ZYGXQNJSKYCXNLZCXM-I20.

References

[1] Ramesh Aditya, Dhariwal Prafulla, Nichol Alex, Chu Casey, and Chen Mark. Hierarchical text-conditional image generation with clip latents. *arXiv:2204.06125*, 2022.

[2] Fan Bao, Shen Nie, Kaiwen Xue, Yue Cao, Chongxuan Li, Hang Su, and Jun Zhu. All are worth words: A vit backbone for diffusion models. In *IEEE CVPR*, 2023.

[3] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020.

[4] Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. Rethinking attention with performers. In *ICLR*, 2021.

[5] Katherine Crowson, Stefan Andreas Baumann, Alex Birch, Tanishq Mathew Abraham, Daniel Z Kaplan, and Enrico Shippole. Scalable high-resolution pixel-space image synthesis with hourglass diffusion transformers. In *ICML*, 2024.

[6] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *CVPRW*, 2020.

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[8] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE CVPR*, 2009.

[9] Prafulla Dhariwal and Alexander Nichol. Diffusion models beat gans on image synthesis. In *NeurIPS*, 2021.

[10] Xiaoyi Dong, Jianmin Bao, Dongdong Chen, Weiming Zhang, Nenghai Yu, Lu Yuan, Dong Chen, and Baining Guo. Cswin transformer: A general vision transformer backbone with cross-shaped windows. In *CVPR*, 2022.

[11] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[12] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021.

[13] Jingwen Fu, Ming Xiao, Chao Ren, and Mikael Skoglund. Computation-resource-efficient task-oriented communications. *IEEE TCOM*, 2025.

[14] Shanghua Gao, Pan Zhou, Ming-Ming Cheng, and Shuicheng Yan. Masked diffusion transformer is a strong image synthesizer. In *IEEE ICCV*, 2023.

[15] Jiayi Guo, Chaofei Wang, You Wu, Eric Zhang, Kai Wang, Xingqian Xu, Humphrey Shi, Gao Huang, and Shiji Song. Zero-shot generative model adaptation via image-specific prompt learning. In *IEEE CVPR*, 2023.

[16] Jiayi Guo, Xingqian Xu, Yifan Pu, Zanlin Ni, Chaofei Wang, Manushree Vasu, Shiji Song, Gao Huang, and Humphrey Shi. Smooth diffusion: Crafting smooth latent spaces in diffusion models. In *IEEE CVPR*, 2024.

[17] Jiayi Guo, Chuanhao Yan, Xingqian Xu, Yulin Wang, Kai Wang, Gao Huang, and Humphrey Shi. Img: Calibrating diffusion models via implicit multimodal guidance. In *ICCV*, 2025.

[18] Jiayi Guo, Junhao Zhao, Chaoqun Du, Yulin Wang, Chunjiang Ge, Zanlin Ni, Shiji Song, Humphrey Shi, and Gao Huang. Everything to the synthetic: Diffusion-driven test-time adaptation via synthetic-domain alignment. In *CVPR*, 2025.

[19] Dongchen Han, Xuran Pan, Yizeng Han, Shiji Song, and Gao Huang. FLatten transformer: Vision transformer using focused linear attention. In *IEEE ICCV*, 2023.

[20] Dongchen Han, Yifan Pu, Zhuofan Xia, Yizeng Han, Xuran Pan, Xiu Li, Jiwen Lu, Shiji Song, and Gao Huang. Bridging the divide: Reconsidering softmax and linear attention. In *NeurIPS*, 2024.

[21] Dongchen Han, Ziyi Wang, Zhuofan Xia, Yizeng Han, Yifan Pu, Chunjiang Ge, Jun Song, Shiji Song, Bo Zheng, and Gao Huang. Demystify mamba in vision: A linear attention perspective. In *NeurIPS*, 2024.

[22] Dongchen Han, Tianzhu Ye, Yizeng Han, Zhuofan Xia, Shiji Song, and Gao Huang. Agent attention: On the integration of softmax and linear attention. In *ECCV*, 2024.

[23] Yizeng Han, Dongchen Han, Zeyu Liu, Yulin Wang, Xuran Pan, Yifan Pu, Chao Deng, Junlan Feng, Shiji Song, and Gao Huang. Dynamic perceiver for efficient visual recognition. In *ICCV*, 2023.

[24] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Honghui Wang, and Yulin Wang. Dynamic neural networks: A survey. *IEEE TPAMI*, 2021.

[25] Yizeng Han, Gao Huang, Shiji Song, Le Yang, Yitian Zhang, and Haojun Jiang. Spatially adaptive feature refinement for efficient inference. *IEEE TIP*, 2021.

[26] Yizeng Han, Zeyu Liu, Zhihang Yuan, Yifan Pu, Chaofei Wang, Shiji Song, and Gao Huang. Latency-aware unified dynamic networks for efficient image recognition. *IEEE TPAMI*, 2024.

[27] Yizeng Han, Yifan Pu, Zihang Lai, Chaofei Wang, Shiji Song, Junfen Cao, Wenhui Huang, Chao Deng, and Gao Huang. Learning to weight samples for dynamic early-exiting networks. In *ECCV*, 2022.

[28] Yizeng Han, Zhihang Yuan, Yifan Pu, Chenhao Xue, Shiji Song, Guangyu Sun, and Gao Huang. Latency-aware spatial-wise dynamic networks. In *NeurIPS*, 2022.

[29] Christian Hansen, Casper Hansen, Stephen Alstrup, Jakob Grue Simonsen, and Christina Lioma. Neural speed reading with structural-jump-lstm. In *ICLR*, 2019.

[30] Ali Hassani and Humphrey Shi. Dilated neighborhood attention transformer. *arXiv:2209.15001*, 2022.

[31] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. In *IEEE CVPR*, 2023.

[32] Ali Hassani, Steven Walton, Jiachen Li, Shen Li, and Humphrey Shi. Neighborhood attention transformer. In *CVPR*, 2023.

[33] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020.

[34] Gao Huang, Danlu Chen, Tianhong Li, Felix Wu, Laurens Van Der Maaten, and Kilian Q Weinberger. Multi-scale dense networks for resource efficient image classification. In *ICLR*, 2018.

[35] Gao Huang, Yulin Wang, Kangchen Lv, Haojun Jiang, Wenhui Huang, Pengfei Qi, and Shiji Song. Glance and focus networks for dynamic visual recognition. *IEEE TPAMI*, 2022.

[36] Rui Huang, Songyou Peng, Ayca Takmaz, Federico Tombari, Marc Pollefeys, Shiji Song, Gao Huang, and Francis Engelmann. Segment3d: Learning fine-grained class-agnostic 3d segmentation without manual labels. In *ECCV*, 2024.

[37] Rui Huang, Henry Zheng, Yan Wang, Zhuofan Xia, Marco Pavone, and Gao Huang. Training an open-vocabulary monocular 3d detection model without 3d data. In *NeurIPS*, 2024.

[38] Bingyi Kang, Yang Yue, Rui Lu, Zhijie Lin, Yang Zhao, Kaixin Wang, Gao Huang, and Jiashi Feng. How far is video generation from world model: A physical law perspective. In *ICML*, 2025.

[39] Angelos Katharopoulos, Apoorv Vyas, Nikolaos Pappas, and François Fleuret. Transformers are rnns: Fast autoregressive transformers with linear attention. In *ICML*, 2020.

[40] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[41] Yawei Li, Yuchen Fan, Xiaoyu Xiang, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Efficient and explicit modelling of image hierarchies for image restoration. In *IEEE CVPR*, 2023.

[42] Yawei Li, Yuchen Fan, Xiaoyu Xiang, Denis Demandolx, Rakesh Ranjan, Radu Timofte, and Luc Van Gool. Efficient and explicit modelling of image hierarchies for image restoration. In *CVPR*, 2023.

[43] 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 *ICCV*, 2021.

[44] 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 *IEEE ICCV*, 2021.

[45] Zeyu Liu, Weicong Liang, Zhanhao Liang, Chong Luo, Ji Li, Gao Huang, and Yuhui Yuan. Glyph-byt5: A customized text encoder for accurate visual text rendering. In *ECCV*, 2024.

[46] Zeyu Liu, Zanlin Ni, Yeguo Hua, Xin Deng, Xiao Ma, Cheng Zhong, and Gao Huang. Coda: Repurposing continuous vaes for discrete tokenization. In *ICCV*, 2025.

[47] Zhixuan Liu, Peter Schaldenbrand, Beverley-Claire Okogwu, Wenxuan Peng, Youngsik Yun, Andrew Hundt, Jihie Kim, and Jean Oh. Scoft: Self-contrastive fine-tuning for equitable image generation. In *CVPR*, 2024.

[48] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2018.

[49] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

[50] Jiachen Lu, Jinghan Yao, Junge Zhang, Xiatian Zhu, Hang Xu, Weiguo Gao, Chunjing Xu, Tao Xiang, and Li Zhang. Soft: Softmax-free transformer with linear complexity. In *NeurIPS*, 2021.

[51] Kangchen Lv, Mingrui Yu, Yifan Pu, Xin Jiang, Gao Huang, and Xiang Li. Learning to estimate 3-d states of deformable linear objects from single-frame occluded point clouds. In *ICRA*, 2022.

[52] Nanye Ma, Mark Goldstein, Michael S Albergo, Nicholas M Boffi, Eric Vanden-Eijnden, and Saining Xie. SiT: Exploring flow and diffusion-based generative models with scalable interpolant transformers. In *ECCV*, 2024.

[53] Zanlin Ni, Yulin Wang, Renping Zhou, Jiayi Guo, Jinyi Hu, Zhiyuan Liu, Shiji Song, Yuan Yao, and Gao Huang. Revisiting non-autoregressive transformers for efficient image synthesis. In *CVPR*, 2024.

[54] Zanlin Ni, Yulin Wang, Renping Zhou, Rui Lu, Jiayi Guo, Jinyi Hu, Zhiyuan Liu, Yuan Yao, and Gao Huang. Adanat: Exploring adaptive policy for token-based image generation. In *ECCV*, 2024.

[55] Xuran Pan, Tianzhu Ye, Zhuofan Xia, Shiji Song, and Gao Huang. Slide-transformer: Hierarchical vision transformer with local self-attention. In *IEEE CVPR*, 2023.

[56] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *IEEE ICCV*, 2023.

[57] Boris T Polyak and Anatoli B Juditsky. Acceleration of stochastic approximation by averaging. *SIAM Journal on Control and Optimization*, 1992.

[58] Yifan Pu, Yizeng Han, Yulin Wang, Junlan Feng, Chao Deng, and Gao Huang. Fine-grained recognition with learnable semantic data augmentation. *IEEE TIP*, 2023.

[59] Yifan Pu, Weicong Liang, Yiduo Hao, Yuhui Yuan, Yukang Yang, Chao Zhang, Han Hu, and Gao Huang. Rank-detr for high quality object detection. In *NeurIPS*, 2024.

[60] Yifan Pu, Yiru Wang, Zhuofan Xia, Yizeng Han, Yulin Wang, Weihao Gan, Zidong Wang, Shiji Song, and Gao Huang. Adaptive rotated convolution for rotated object detection. In *IEEE ICCV*, 2023.

[61] Yifan Pu, Zhuofan Xia, Jiayi Guo, Dongchen Han, Qixiu Li, Duo Li, Yuhui Yuan, Ji Li, Yizeng Han, Shiji Song, et al. Efficient diffusion transformer with step-wise dynamic attention mediators. In *ECCV*, 2024.

[62] Yifan Pu, Yiming Zhao, Zhicong Tang, Ruihong Yin, Haoxing Ye, Yuhui Yuan, Dong Chen, Jianmin Bao, Sirui Zhang, Yanbin Wang, et al. ART: Anonymous region transformer for variable multi-layer transparent image generation. In *CVPR*, 2025.

[63] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional networks for biomedical image segmentation. In *MICCAI*, 2015.

[64] Zhuoran Shen, Mingyuan Zhang, Haiyu Zhao, Shuai Yi, and Hongsheng Li. Efficient attention: Attention with linear complexities. In *WACV*, 2021.

[65] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.

[66] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[67] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017.

[68] Chaofei Wang, Qisen Yang, Rui Huang, Shiji Song, and Gao Huang. Efficient knowledge distillation from model checkpoints. In *NeurIPS*, 2022.

[69] Jiangshan Wang, Yifan Pu, Yizeng Han, Jiayi Guo, Yiru Wang, Xiu Li, and Gao Huang. Gra: Detecting oriented objects through group-wise rotating and attention. In *ECCV*, 2024.

[70] Shenzhi Wang, Liwei Wu, Lei Cui, and Yujun Shen. Glancing at the patch: Anomaly localization with global and local feature comparison. In *IEEE CVPR*, 2021.

[71] Shenzhi Wang, Le Yu, Chang Gao, Chujie Zheng, Shixuan Liu, Rui Lu, Kai Dang, Xionghui Chen, Jianxin Yang, Zhenru Zhang, Yuqiong Liu, An Yang, Andrew Zhao, Yang Yue, Shiji Song, Bowen Yu, Gao Huang, and Junyang Lin. Beyond the 80/20 rule: High-entropy minority tokens drive effective reinforcement learning for llm reasoning. In *NeurIPS*, 2025.

[72] Sinong Wang, Belinda Z Li, Madian Khabsa, Han Fang, and Hao Ma. Linformer: Self-attention with linear complexity. *arXiv:2006.04768*, 2020.

[73] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. In *ICCV*, 2021.

[74] Yulin Wang, Zhaoxi Chen, Haojun Jiang, Shiji Song, Yizeng Han, and Gao Huang. Adaptive focus for efficient video recognition. In *IEEE ICCV*, 2021.

[75] Yulin Wang, Yizeng Han, Chaofei Wang, Shiji Song, Qi Tian, and Gao Huang. Computation-efficient deep learning for computer vision: A survey. *Cybernetics and Intelligence*, 2023.

[76] Yulin Wang, Rui Huang, Shiji Song, Zeyi Huang, and Gao Huang. Not all images are worth 16x16 words: Dynamic transformers for efficient image recognition. In *NeurIPS*, 2021.

[77] Yulin Wang, Zanlin Ni, Yifan Pu, Cai Zhou, Jixuan Ying, Shiji Song, and Gao Huang. Infopro: Locally supervised deep learning by maximizing information propagation. *IJCV*, 2025.

[78] Yulin Wang, Yang Yue, Yang Yue, Huanqian Wang, Haojun Jiang, Yizeng Han, Zanlin Ni, Yifan Pu, Minglei Shi, Rui Lu, Qisen Yang, et al. Emulating human-like adaptive vision for efficient and flexible machine visual perception. *Nature Machine Intelligence*, 2025.

[79] Zhuofan Xia, Dongchen Han, Yizeng Han, Xuran Pan, Shiji Song, and Gao Huang. GSVA: Generalized segmentation via multimodal large language models. In *IEEE CVPR*, 2024.

[80] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable attention. In *CVPR*, 2022.

[81] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Vision transformer with deformable attention. In *IEEE CVPR*, 2022.

[82] Zhuofan Xia, Xuran Pan, Shiji Song, Li Erran Li, and Gao Huang. Dat++: Spatially dynamic vision transformer with deformable attention. *arXiv:2309.01430*, 2023.

[83] Yunyang Xiong, Zhanpeng Zeng, Rudrasis Chakraborty, Mingxing Tan, Glenn Fung, Yin Li, and Vikas Singh. Nyströmformer: A nyström-based algorithm for approximating self-attention. In *AAAI*, 2021.

[84] Qisen Yang, Shenzhi Wang, Matthieu Gaetan Lin, Shiji Song, and Gao Huang. Boosting offline reinforcement learning with action preference query. In *ICML*, 2023.

[85] Qisen Yang, Shenzhi Wang, Qihang Zhang, Gao Huang, and Shiji Song. Hundreds guide millions: Adaptive offline reinforcement learning with expert guidance. *IEEE TNNLS*, 2023.

[86] Qisen Yang, Zekun Wang, Honghui Chen, Shenzhi Wang, Yifan Pu, Xin Gao, Wenhao Huang, Shiji Song, and Gao Huang. Psychogat: A novel psychological measurement paradigm through interactive fiction games with llm agents. In *ACL*, 2024.

[87] Xiulong Yang, Sheng-Min Shih, Yinlin Fu, Xiaoting Zhao, and Shihao Ji. Your ViT is secretly a hybrid discriminative-generative diffusion model. *arXiv:2208.07791*, 2022.

[88] Tianzhu Ye, Li Dong, Yuqing Xia, Yutao Sun, Yi Zhu, Gao Huang, and Furu Wei. Differential transformer. In *ICLR*, 2025.

[89] Haoran You, Yunyang Xiong, Xiaoliang Dai, Bichen Wu, Peizhao Zhang, Haoqi Fan, Peter Vajda, and Yingyan Lin. Castling-vit: Compressing self-attention via switching towards linear-angular attention during vision transformer inference. In *IEEE CVPR*, 2023.

[90] Yang Yue, Zhiqi Chen, Rui Lu, Andrew Zhao, Zhaokai Wang, Shiji Song, and Gao Huang. Does reinforcement learning really incentivize reasoning capacity in llms beyond the base model? In *NeurIPS*, 2025.

[91] Yang Yue, Yulin Wang, Bingyi Kang, Yizeng Han, Shenzhi Wang, Shiji Song, Jiashi Feng, and Gao Huang. Deer-vla: Dynamic inference of multimodal large language models for efficient robot execution. In *NeurIPS*, 2024.

[92] Sangdoo Yun, Dongyoon Han, Seong Joon Oh, Sanghyuk Chun, Junsuk Choe, and Youngjoon Yoo. Cutmix: Regularization strategy to train strong classifiers with localizable features. In *ICCV*, 2019.

[93] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. In *ICLR*, 2018.

[94] Andrew Zhao, Yiran Wu, Yang Yue, Tong Wu, Quentin Xu, Matthieu Lin, Shenzhi Wang, Qingyun Wu, Zilong Zheng, and Gao Huang. Absolute zero: Reinforced self-play reasoning with zero data. In *NeurIPS*, 2025.

[95] Wangbo Zhao, Yizeng Han, Jiasheng Tang, Zhikai Li, Yibing Song, Kai Wang, Zhangyang Wang, and Yang You. A stitch in time saves nine: Small vlm is a precise guidance for accelerating large vlms. In *CVPR*, 2025.

[96] Wangbo Zhao, Yizeng Han, Jiasheng Tang, Kai Wang, Hao Luo, Yibing Song, Gao Huang, Fan Wang, and Yang You. Dydit++: Dynamic diffusion transformers for efficient visual generation. *arXiv:2504.06803*, 2025.

[97] Wangbo Zhao, Yizeng Han, Jiasheng Tang, Kai Wang, Yibing Song, Gao Huang, Fan Wang, and Yang You. Dynamic diffusion transformer. In *ICLR*, 2025.

[98] Wangbo Zhao, Jiasheng Tang, Yizeng Han, Yibing Song, Kai Wang, Gao Huang, Fan Wang, and Yang You. Dynamic tuning towards parameter and inference efficiency for vit adaptation. In *NeurIPS*, 2024.

[99] Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers. *TMLR*, 2024.

[100] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *AAAI*, 2020.

[101] Lei Zhu, Xinjiang Wang, Zhanghan Ke, Wayne Zhang, and Rynson WH Lau. Biformer: Vision transformer with bi-level routing attention. In *CVPR*, 2023.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The main claims made in the abstract and introduction accurately reflect the paper's contributions and scope.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitations of the work in the paper.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: This paper fully discloses all the information needed to reproduce the main experimental results of the paper.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We use public data to conduct experiments. However, in the submission and the reviewing period, we do not provide open source data to prevent disclosure of author information.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide all the training and testing details in the submission.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Error bars are not reported because it would be too computationally expensive.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [No]

   Justification: We provide information on the computer resources in the first part of the experiment section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conducted in the paper conform with the NeurIPS Code of Ethics, in every respect.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: This work focuses on attention mechanism and has no direct societal impact.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: The paper poses no such risks as it is a foundational research focusing on attention paradigm.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: We use all assets properly according to their licenses, and give credits to the creators in Section 4.1.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.

- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [NA]

Justification: The LLM is only used during polishing the writing.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.