Beyond Generation: Unlocking Discriminative Representations from Diffusion Models

Haowen Cui[1] **Shuo Chen**[2*] **Ge Wu**[3] **Yikai Ge**[3] **Ge Gao**[2]
Xiang Li[3] **Jun Li**[1,2*] **Jian Yang**[1,2*]
[1]PCA Lab, Nanjing University of Science and Technology, China
[2]School of Intelligence Science and Technology, Nanjing University, China
[3]VCIP, Nankai University, China

Abstract

Recent advancements in generative models have increasingly leveraged self-supervised visual representations to guide the synthesis process, significantly improving generation quality and semantic consistency. However, the reverse paradigm of harnessing the generative process to enhance visual representations remains largely underexplored. In this paper, we investigate the semantic properties of class tokens synthesized by generative models equipped with representation entanglement. We observe that the generative model can generate class tokens that exhibit stronger discriminative capabilities than those extracted from the original pretrained visual models. Motivated by this observation, we propose a new Generation-to-Perception Knowledge Distillation (GPKD) framework, where generated class tokens are used as teacher signals to instill global semantics into a student model. To prevent the degradation of local details, we further incorporate a masked patch-level distillation objective. This dual distillation strategy enhances global representations while mitigating the forgetting of local details, thereby producing more robust representations. Extensive experiments demonstrate that GPKD obtains consistent improvements across various downstream tasks, achieving a gain of more than 2% in k-NN accuracy on ImageNet-1K. Code is available at https://github.com/huanmx/GPKD.

1 Introduction

The landscape of generative models has evolved rapidly, driven by the remarkable success of diffusion models in synthesizing high-fidelity images [1, 2, 3, 4, 5]. Despite this success, the internal representations of vanilla diffusion models exhibit a semantic gap compared with pretrained visual representation models [6, 7, 8]. To bridge this gap, an emerging paradigm focuses on the representation alignment of generative models with self-supervised visual models. Since modern pretrained visual representation models encode highly structured and semantically rich feature spaces, incorporating these pretrained representations into the denoising process provides strong discriminative guidance. This representation alignment not only accelerates convergence but also significantly enhances image quality by enabling the synthesis of more semantically coherent content.

Among these methods, REG [9] explicitly entangled the class token extracted from visual encoders into the denoising process. However, a fundamental question remains largely unexplored: *what are the properties of the class token generated by generative models (e.g., REG)?* To investigate this, we design a denoising process to probe the generated representations as shown in Fig. 1 (a). Specifically, we perform inference on a noise-corrupted image to guide the denoising of a purely

*Corresponding Authors

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

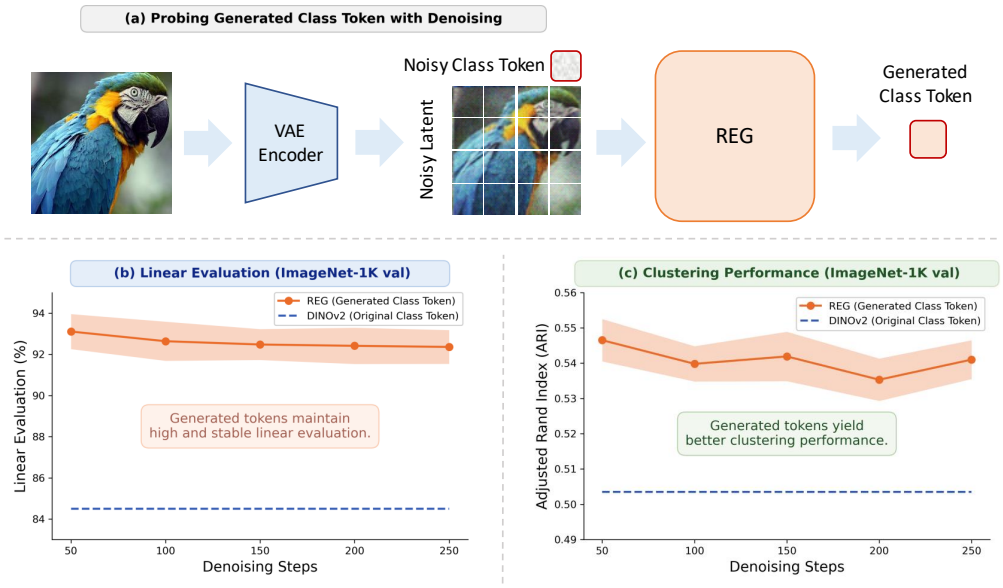

Figure 1: (a) Class token generation process. An image latent derived from a clean image is used to guide the denoising of a class token initialized purely from Gaussian noise. (b) Linear evaluation on the ImageNet val set. (c) Clustering performance on the ImageNet val set. The DINOv2 result is a fixed horizontal reference independent of the denoising step.

noise-initialized class token. We then evaluate the representation quality of the generated class tokens across different denoising steps. Surprisingly, the generative model produces a distinctly refined representation. As shown in Fig. 1 (b) and (c), the class tokens generated by REG consistently and significantly outperform the original DINOv2 [10] in both classification (linear evaluation) and clustering performance (adjusted rand index) across various denoising steps.

Motivated by this interesting phenomenon, we propose a novel Generation-to-Perception Knowledge Distillation (GPKD) framework that uses generative models to enhance pretrained visual representation models. We leverage the generative model to synthesize refined class tokens. These tokens act as a robust global teacher to supervise a perceptive student model (e.g., DINOv2). Although these generated class tokens provide superior semantic supervision, relying exclusively on them for distillation may risk driving the model towards overly invariant representations. This inevitably collapses fine-grained spatial details, resulting in forgetting of dense perception capabilities. To address this, we couple the global supervision with a complementary patch-level distillation objective, where the supervisory information comes from the original pretrained model. By explicitly constraining the student to preserve local details, this dual-distillation framework ensures the robustness of the learned representation in both global and local perspectives. Extensive experiments on various downstream benchmarks demonstrate that our method achieves state-of-the-art performance, showing strong generalizability and practical effectiveness.

The main contributions of this work are summarized as follows:

- We analyze the class tokens produced by a class-conditioned generative model, revealing that the generative process yields more perceptive class tokens compared with pretrained visual encoders.
- We propose a novel approach that transfers the global semantics from generated tokens into a student visual encoder, thereby enhancing the perceptive capability of the learned representations, complemented by a new masked patch-level objective to preserve local structural details.
- Experimental results demonstrate that our label-aware distillation method significantly enhances the representation capability of existing foundation models, e.g., our method achieves more than 2% gain in k-NN accuracy on ImageNet-1K.

2 Related Work

Generative Models Generative models aim to learn the underlying data distribution and synthesize realistic samples through a parameterized generation process. Diffusion models have recently become the dominant paradigm due to their remarkable generation quality and training stability [11, 12, 13]. Denoising diffusion probabilistic models [1] formulate generation as a gradual denoising process that transforms Gaussian noise into data samples. Subsequent works such as score-based generative models [14], latent diffusion models [15], and diffusion transformers [16] have substantially improved scalability, sample fidelity, and efficiency, making diffusion models the foundation of modern image generation. To address multi-step iterative sampling, flow matching [4] provides a simulation-free objective for training continuous normalizing flows by directly regressing the target velocity field along predefined probability paths. Recent studies [17, 18] have shown that flow matching achieves the state-of-the-art performance across multiple modalities while offering a flexible and efficient alternative to conventional diffusion training.

Self-Supervised Learning Self-Supervised Learning (SSL) aims to learn powerful and generalizable representations from unlabeled data via diverse pretext tasks. Contrastive methods such as SimCLR [19], MoCo [20], BYOL [21], and SimSiam [22] learn representations by maximizing feature similarity between different augmented views of the same sample while encouraging instance discrimination across samples. The DINO series [23, 10, 24] leverages a teacher-student framework to produce highly structured and semantically meaningful features, exhibiting remarkable capabilities in feature clustering, object discovery, and dense prediction tasks. These approaches have demonstrated strong performance in learning invariant and transferable features [25, 26, 27, 28, 29]. Another line of work focuses on reconstruction objectives. Masked autoencoder [30] and its variants such as iBOT [31] and SimMIM [32] learn representations by reconstructing masked portions of the input. These SSL methods have become the foundation of modern visual encoders, providing robust and semantically rich features that generalize well across downstream tasks.

Representation Alignment for Generation Recent studies have shown that the internal representations of vanilla diffusion models exhibit a semantic gap compared with those produced by pretrained visual encoders [6, 7, 8]. To address this issue, some methods incorporate self-supervised objectives as auxiliary signals into generative training. Disp [33] explicitly disperses internal representations in the hidden space. SRA [34] aligns earlier layer latent representations under higher noise conditions with later layer representations under lower noise conditions in a self-distillation manner. To bridge the semantic gap between generative models and visual encoders, recent works have also explored leveraging robust visual representations from pretrained foundation models to accelerate diffusion training. REPA [6] aligns intermediate diffusion features with high-quality self-supervised representations. REG [9] incorporates class tokens with rich semantic information into diffusion training, establishing context-aware semantic guidance that persists throughout the generative process.

Diffusion Models for Downstream Tasks Beyond image synthesis, diffusion models have also been widely explored for downstream visual understanding tasks. Early studies show that intermediate activations from pretrained denoising diffusion models provide effective pixel-level representations for label-efficient semantic segmentation [35]. With the emergence of large-scale text-to-image diffusion models, recent works further exploit these for dense prediction tasks. ODISE [36] combines frozen internal representations from a text-to-image diffusion model with CLIP for open-vocabulary panoptic segmentation. VPD [37] adapts pretrained text-to-image diffusion models as visual backbones for perception tasks. TADP [38] uses automatically generated captions to improve text-image alignment within diffusion backbones. DDP [39] formulates dense visual prediction as a conditional denoising process, enabling diffusion models to perform dense prediction tasks. Marigold [40] repurposes pretrained latent diffusion models for monocular depth estimation, further demonstrating the potential of generative priors for geometric perception. Unlike these works that mainly adapt pretrained diffusion models as task-specific perception modules, our work focuses on improving the perception capabilities of visual encoders by distilling knowledge from generated visual tokens.

This paper aims to further investigate the properties of generated class tokens. We explore how their unique advantages can be exploited to distill knowledge into pretrained visual representation models. By transferring these generative strengths, we aim to enhance the model's capability and downstream performance. This aligns with the emerging consensus [41], demonstrating that generative models can also serve as powerful vision learners.

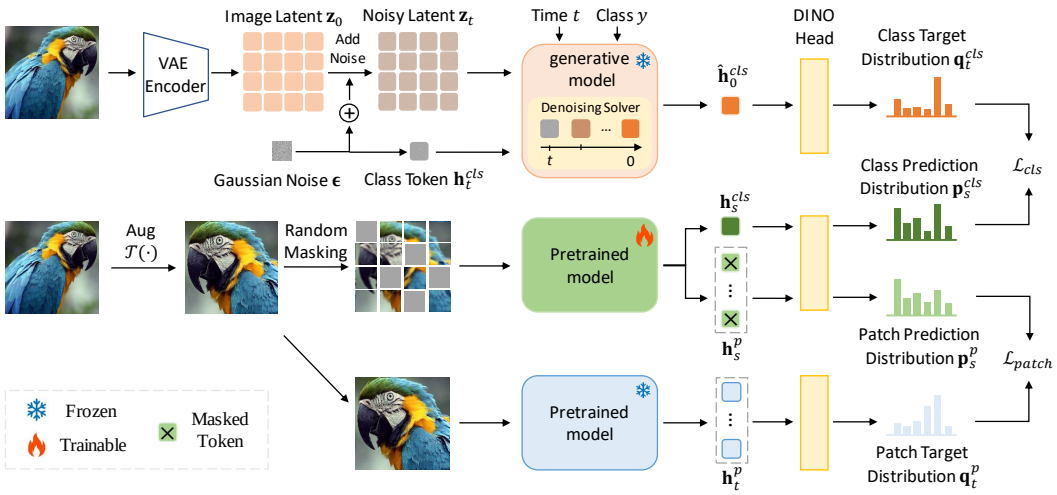

Figure 2: Overview of the proposed method. The class-conditioned generative model produces a highly discriminative class token from a noised view, serving as a teacher class token. Concurrently, the original pretrained model (e.g., DINOv2) provides teacher patch tokens from an augmented view. The student model is trained by aligning the prediction distribution of class token and masked patch token with the teacher's target distribution.

3 Method

3.1 Preliminary

Our method is built upon REG [9], which extends flow matching by incorporating high-level class tokens into the denoising process. REG is instantiated on top of Scalable Interpolant Transformer (SiT) [17], which is a class-conditioned generative model. Given an input image I, a latent representation $\mathbf{z}_0$ is first extracted by the VAE encoder. The intermediate noisy state at time t is defined as an interpolation between noise and data:

$$\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \boldsymbol{\epsilon}_z, \tag{1}$$

where $\boldsymbol{\epsilon}_z \sim \mathcal{N}(0, I)$ is the Gaussian noise. $\alpha_t, \sigma_t \in [0, 1]$ control the interpolation trajectory between data and noise. Let y denote the class label of the input image. Under this formulation, the generation process can be characterized by the probability flow Ordinary Differential Equation (ODE):

$$\dot{\mathbf{z}}_t = \mathbf{v}(\mathbf{z}_t, t \mid y), \tag{2}$$

where $\mathbf{v}(\mathbf{z}_t, t \mid y)$ denotes the velocity field, which can be formulated by the conditional expectation:

$$\mathbf{v}(\mathbf{z}_t, t \mid y) = \dot{\alpha}_t \mathbb{E}[\mathbf{z}_0 \mid \mathbf{z}_t, y] + \dot{\sigma}_t \mathbb{E}[\boldsymbol{\epsilon}_z \mid \mathbf{z}_t, y]. \tag{3}$$

Following SiT, the velocity is learned by minimizing the following loss function:

$$\mathcal{L}_v = \int \mathbb{E}\left[\|\mathbf{v}_\theta(\mathbf{z}_t, t, y) - \dot{\alpha}_t \mathbf{z}_0 - \dot{\sigma}_t \boldsymbol{\epsilon}_z\|_2^2 \right] dt. \tag{4}$$

Different from conventional diffusion models that only denoise image latents, REG introduces a class token into the generation process and conditions the diffusion transformer on an image class label y. Given the input image I, the class token $\mathbf{h}_0^{cls} = f(I)$ is obtained from a pretrained visual representation model f. To entangle semantic and generative information, REG injects synchronized noise into both the image latent and the class token:

$$\mathbf{h}_t^{cls} = \alpha_t \mathbf{h}_0^{cls} + \sigma_t \boldsymbol{\epsilon}_{cls}, \tag{5}$$

where $\boldsymbol{\epsilon}_{cls} \sim \mathcal{N}(0, I)$ denotes Gaussian noise added to the class token. During training, REG optimizes a joint denoising objective over both the image latent and the class token:

$$\mathcal{L}_v = \int \mathbb{E}\Big[\|\mathbf{v}_\theta(\mathbf{z}_t, t, y) - \dot{\alpha}_t \mathbf{z}_0 - \dot{\sigma}_t \boldsymbol{\epsilon}_z\|_2^2 + \beta \|\mathbf{v}_\theta(\mathbf{h}_t^{cls}, t, y) - \dot{\alpha}_t \mathbf{h}_0^{cls} - \dot{\sigma}_t \boldsymbol{\epsilon}_{cls}\|_2^2 \Big] dt, \tag{6}$$

where $\beta > 0$ controls the relative importance of class token denoising. This design allows the diffusion transformer to jointly process low-level visual content and high-level semantic information.

3.2 Class Token Generation

A key observation of our work is that REG can generate highly discriminative class tokens. Instead of initializing both the image latent and the class token from random noise, we consider a conditional reconstruction setting in which the image latent is derived from a clean image, while the class token is initialized from Gaussian noise.

Given an input image I, we first obtain its latent representation $\mathbf{z}_0 = E_z(I)$ using the VAE encoder. We then perturb the latent with Gaussian noise at timestep t:

$$\mathbf{z}_t = \alpha_t \mathbf{z}_0 + \sigma_t \boldsymbol{\epsilon}_z, \qquad \boldsymbol{\epsilon}_z \sim \mathcal{N}(0, I). \tag{7}$$

In contrast, the semantic class token is initialized purely from noise:

$$\mathbf{h}_t^{cls} = \boldsymbol{\epsilon}_{cls}, \qquad \boldsymbol{\epsilon}_{cls} \sim \mathcal{N}(0, I). \tag{8}$$

The noised latent $\mathbf{z}_t$ is patchified into latent tokens $\tilde{\mathbf{z}}_t \in \mathbb{R}^{N \times D}$, where N is the number of tokens and D is the feature dimensionality. The model input is then formed as $\mathbf{x}_t = [\, \mathbf{h}_t^{cls}; \tilde{\mathbf{z}}_t \,] \in \mathbb{R}^{(N+1) \times D}$. REG then performs conditional iterative denoising and predicts the velocity associated with the class token, denoted by $\mathbf{v}_{cls}(\mathbf{x}_t, t, y) \in \mathbb{R}^{1 \times D}$.

To recover the clean representation, we sample from the probability flow ODE by integrating the predicted velocities in reverse time. Specifically, we employ a numerical ODE solver starting from timestep t down to 0. At each step with size Δt, the state of the class token is iteratively updated as:

$$\hat{\mathbf{h}}_{t-\Delta t}^{cls} = \mathbf{h}_t^{cls} - \Delta t \cdot \mathbf{v}_{cls}(\mathbf{x}_t, t, y). \tag{9}$$

Notably, the semantic structure of generated class tokens is inherited from the DINOv2 target used to train REG, while the jointly processed image latent provides instance-specific visual content. The class condition y also adds an explicit category constraint. The learned flow dynamics integrate these signals into a continuous semantic representation. Therefore, although the class token originates from pure Gaussian noise, the resulting token provides highly discriminative information than a discrete class label alone.

3.3 Class Token Distillation for Visual Representation Learning

The goal of GPKD is to obtain a stronger and reusable foundation visual encoder. The generated class token is highly discriminative and has the same feature-vector form as the DINOv2 class token, which makes it a natural teacher signal. The overview of our method is shown in Fig. 2. For an input image I, the teacher branch produces the reconstructed class token $\hat{\mathbf{h}}_0^{cls}$ through the image-conditioned generation process described in Sec. 3.2.

For the student branch, we apply a stochastic augmentation $\mathcal{T}(\cdot)$ to the same image and feed the augmented view into the student encoder f_s. The student outputs a class token and a set of patch tokens, denoted by $\mathbf{h}_s^{cls}$ and $\mathbf{H}_s^p = \{\mathbf{h}_{s,i}^p\}_{i=1}^N$, respectively. We then map both teacher and student class tokens into a shared prototype space through a DINO head $g(\cdot)$, yielding the corresponding prototype logits $\mathbf{u}_t^{cls} = g(\hat{\mathbf{h}}_0^{cls})$ and $\mathbf{u}_s^{cls} = g(\mathbf{h}_s^{cls})$. The teacher logits are scaled by the teacher temperature τ_t and converted into a target assignment distribution as $\mathbf{q}_t^{cls} = \mathrm{Sinkhorn}(\mathbf{u}_t^{cls}/\tau_t)$, while the student logits are transformed into a prediction distribution as $\mathbf{p}_s^{cls} = \mathrm{Softmax}(\mathbf{u}_s^{cls}/\tau_s)$, where τ_s is the student temperature. We align the student distribution to the teacher target with the following loss:

$$\mathcal{L}_{cls} = -\sum_{k=1}^{K} q_{t,k}^{cls} \log p_{s,k}^{cls}, \tag{10}$$

where K is the number of prototypes.

Distilling only the class token mainly transfers global semantic information, but may be insufficient for preserving strong patch-level representations. To maintain the quality of dense features, we further introduce a patch-level distillation objective. Specifically, we randomly mask a subset of patches in

the student input image. Let $M \subseteq \{1, \ldots, N\}$ denote the set of masked patch indices. The student produces the corresponding masked patch tokens $\{\mathbf{h}^p_{s,i}\}_{i \in M}$, while the teacher provides patch-level target features $\{\mathbf{h}^p_{t,i}\}_{i \in M}$ from the original DINOv2 model.

For each masked position $i \in M$, we project the teacher and student patch tokens into the same prototype space, obtaining $\mathbf{u}^p_{t,i} = g(\mathbf{h}^p_{t,i})$ and $\mathbf{u}^p_{s,i} = g(\mathbf{h}^p_{s,i})$. The teacher patch logits are scaled by τ_t and normalized into target distributions as $\mathbf{q}^p_{t,i} = \mathrm{Sinkhorn}(\mathbf{u}^p_{t,i}/\tau_t)$, while the student patch logits are converted into prediction distributions as $\mathbf{p}^p_{s,i} = \mathrm{Softmax}(\mathbf{u}^p_{s,i}/\tau_s)$. We then match the student masked-patch predictions to the corresponding teacher targets by

$$\mathcal{L}_{patch} = -\frac{1}{|M|} \sum_{i \in M} \sum_{k=1}^{K} q^p_{t,i,k} \log p^p_{s,i,k}. \tag{11}$$

The final objective combines the class-token distillation loss and the patch-token distillation loss:

$$\mathcal{L} = \lambda_{cls}\mathcal{L}_{cls} + \lambda_{patch}\mathcal{L}_{patch}, \tag{12}$$

where $\lambda_{cls}, \lambda_{patch} > 0$ denote the loss weights. In this way, the generated class token transfers high-level discriminative semantics from the generative model, while the patch-level objective preserves dense visual representations during student training. The complete training procedure of our proposed method is provided in Appendix A.

4 Experiments

In this section, we first present the implementation details of the experiments (Sec. 4.1). Then, we compare our method with existing state-of-the-art methods (Sec. 4.2, 4.3). Finally, we provide ablation studies of the proposed method (Sec. 4.4).

4.1 Implementation Details

We adopt REG as the generative teacher and follow its default setup. All experiments are conducted on ImageNet-1K [42] with images center-cropped and resized to 256×256. Each image is encoded into a latent representation of size $32 \times 32 \times 4$ using the Stable Diffusion VAE [15]. We use SiT-XL as the generative backbone with the same interpolant parameterization as REG, i.e., $\alpha_t = 1 - t$ and $\sigma_t = t$. The dimensionality of the generated class tokens is the same as that of DINOv2-B. The process of class token generation is conducted without classifier-free guidance.

We initialize the student from DINOv2 and perform distillation with the generated class tokens. We use a two-stage training recipe, where the backbone is frozen and only the DINO head is trained in the first stage, followed by joint optimization in the second stage. We train with the AdamW optimizer [43] and a total batch size of 512 for 20 epochs. The learning rate is linearly warmed up for the first 5 epochs to a base value following the linear scaling rule, i.e., $lr = 0.0002 \times \mathrm{batchsize}/1024$, and is then decayed with a cosine schedule. For the prototype assignment, the student temperature is fixed to $\tau_s = 0.1$, while the teacher temperature τ_t is linearly warmed up from 0.04 to 0.07. The number of prototypes is set to 131072. The loss weights λ_{cls} and λ_{patch} are both set to 1. For each image, we construct two global crops and eight local crops using random cropping, color jittering, horizontal flipping, and grayscale conversion. Gaussian blurring and solarization are additionally applied to the global crops. In each batch, 50% of the global views are masked, and the mask ratio is sampled uniformly from $[0.1, 0.5]$.

Since REG is a class-conditioned generative model, the ground-truth ImageNet labels are used as the class condition when constructing the generated teacher tokens. Therefore, we compare three supervised baselines in addition to the self-supervised visual encoders. First, we perform Supervised Fine-Tuning (SFT) on the pretrained DINOv2-B/14 model. Second, we fine-tune the same DINOv2 model using the training recipe proposed by DeiT III [44]. Third, we use a supervised ViT-H model with a parameter scale comparable to REG as the teacher and distill its output logits into DINOv2-B/14, which is denoted as SD. These comparisons use the same ImageNet-1K labels and provide supervision-matched references for evaluating GPKD.

Table 1: **Linear evaluation on ImageNet-1K of frozen pretrained features.** We report top-1 accuracy on the validation set for publicly available models trained on public or private data, and with or without text supervision (Text sup.). For reference, we also report the k-NN performance on the validation set.

Method	Arch.	Data	Text sup.	k-NN	linear		
				val	val	ReaL	V2
Weakly supervised							
CLIP [45]	ViT-L/14	WIT-400M	✓	79.8	84.3	88.1	75.3
CLIP [45]	ViT-L/14_{336}	WIT-400M	✓	80.5	85.3	88.8	75.8
SWAG [46]	ViT-H/14	IG3.6B	✓	82.6	85.7	88.7	**77.6**
OpenCLIP [47]	ViT-H/14	LAION-2B	✓	81.7	84.4	88.4	75.5
OpenCLIP [47]	ViT-G/14	LAION-2B	✓	83.2	86.2	**89.4**	77.2
EVA-CLIP [48]	ViT-g/14	custom*	✓	**83.5**	**86.4**	89.3	77.4
Self-supervised							
MAE [30]	ViT-H/14	INet-1K	×	49.4	76.6	83.3	64.8
DINO [23]	ViT-S/8	INet-1K	×	78.6	79.2	85.5	68.2
SEERv2 [49]	RG10B	IG2B	×	–	79.8	–	–
MSN [50]	ViT-L/7	INet-1K	×	79.2	80.7	86.0	69.7
EsViT [51]	Swin-B/W=14	INet-1K	×	79.4	81.3	87.0	70.4
Mugs [52]	ViT-L/16	INet-1K	×	80.2	82.1	86.9	70.8
iBOT [31]	ViT-L/16	INet-22K	×	72.9	82.3	87.5	72.4
DINOv2 [10]	ViT-S/14	LVD-142M	×	79.0	81.1	86.6	70.9
	ViT-B/14	LVD-142M	×	**82.1**	**84.5**	**88.3**	**75.1**
Supervised							
SFT	ViT-B/14	INet-1K	×	83.8	85.0	89.1	76.1
DeiT III [44]	ViT-B/14	INet-1K	×	83.7	84.9	88.8	75.8
SD	ViT-B/14	INet-1K	×	83.7	84.8	88.6	75.2
GPKD	ViT-S/14	INet-1K	×	81.4	82.1	87.9	72.5
	ViT-B/14	INet-1K	×	**84.3**	**85.4**	**89.6**	**76.8**

4.2 Classification on ImageNet-1K

k-NN and Linear Evaluation. To evaluate the quality of the distilled representations, we use a k-NN classifier and a linear classifier on the frozen features. We follow the evaluation protocols in [10]. As shown in Tab. 1, our method achieves 85.4% top-1 accuracy under linear evaluation on ImageNet-1K, improving over the DINOv2 ViT-B/14 baseline by 0.9%. A similar gain is observed in k-NN evaluation, where performance increases from 82.1% to 84.3%. To further validate the robustness and out-of-distribution generalization of the learned features, we evaluate our model on the ImageNet-ReaL and ImageNet-V2 benchmarks. Our ViT-B/14 model reaches 89.6% on ReaL and 76.8% on V2, consistently outperforming the DINOv2 baseline by large margins (+1.3% and +1.7%, respectively). Remarkably, GPKD also surpasses all supervised baselines under both evaluation protocols. These results indicate that the proposed distillation strategy substantially improves the linear separability and generalization ability of the learned frozen representations. In particular, the image-conditioned class tokens generated by REG provide a robust supervisory signal that enhances the discriminative quality of the student features even beyond traditional discrete labels.

Visualization Results To evaluate the representation quality of our method, we perform a Principal Component Analysis (PCA) visualization, following the protocol established by DINOv2. For each image, we extract the feature maps and compute the PCA. The first principal component is thresholded to separate the foreground main object from the background, while the subsequent three principal components are mapped to the RGB color space to visualize semantic part matching. The comparisons between DINOv2 and our proposed method are presented in Fig. 3. The visualization results show that our method successfully suppresses background clutter and adheres tightly to natural object boundaries. This results in dense and continuous semantic part mappings, demonstrating our model's superior capability for semantic understanding.

4.3 Downstream Tasks

Transfer Learning. Tab. 2 presents linear evaluation results on 11 fine-grained and transfer benchmarks. Under the ViT-B/14 setting, our method achieves an average accuracy of 90.7%,



Figure 3: Comparison of the PCA visualization of extracted features. For each image, we display the original image (left), the PCA visualization of DINOv2 (middle), and the PCA visualization of our proposed method (right). Our method yields highly purified feature maps that accurately delineate object boundaries.

Table 2: **Linear evaluation of frozen features on fine-grained benchmarks.** Accuracy on 11 benchmarks covering objects, scenes and textures following the evaluation protocol proposed in [19].

Method	Arch	Food	C10	C100	SUN	Cars	Aircr	DTD	Pets	Cal101	Flowers	CUB	Avg
MAE	ViT-H/14	78.4	96.1	83.9	63.9	56.1	63.4	75.4	89.4	95.9	92.3	57.2	77.5
DINO	ViT-B/8	85.1	97.2	86.9	70.3	76.6	70.6	79.6	93.2	95.4	97.6	81.7	84.9
iBOT	ViT-L/16	91.0	99.0	92.8	75.6	71.8	72.4	80.7	87.7	**97.5**	99.6	82.1	86.4
DINOv2	ViT-S/14	89.1	97.7	87.5	74.4	81.6	74.0	80.6	95.1	97.0	99.6	88.1	87.7
	ViT-B/14	92.8	98.7	91.3	77.3	88.2	79.4	83.3	**96.2**	96.1	99.6	**89.6**	90.2
SFT	ViT-B/14	89.9	97.9	87.8	72.0	86.2	75.4	80.1	95.7	96.4	98.7	88.7	88.1
DeiT III	ViT-B/14	91.0	98.4	89.8	74.7	87.8	79.0	80.9	95.5	97.0	99.3	89.8	89.4
SD	ViT-B/14	87.3	98.1	88.0	70.5	79.2	67.5	78.4	95.7	97.0	97.1	85.4	85.8
GPKD	ViT-S/14	89.6	97.8	88.2	75.0	83.2	75.3	80.3	95.3	97.1	99.6	89.1	88.2
	ViT-B/14	**93.3**	**98.9**	**91.8**	**78.7**	**88.6**	**80.8**	**84.8**	**96.2**	96.1	**99.7**	89.3	**90.7**

outperforming the DINOv2 baseline. It also ourperforms compared supervised baselines. Notably, it delivers superior performance on diverse datasets, including Food, CIFAR-10/100, SUN, Cars, Aircraft, and DTD. These datasets present a wide spectrum of visual challenges, ranging from capturing highly localized discriminative features in fine-grained categories (e.g., Cars and Aircraft) to recognizing global statistical patterns in textures and scenes (e.g., DTD and SUN). These consistent improvements across object-centric, scene-centric, and texture-centric tasks demonstrate that the benefits of our distillation strategy extend far beyond ImageNet, effectively enhancing general-purpose transferability across different levels of semantic abstraction. Crucially, the combination of global class token distillation and the patch-level objective allows the model to improve holistic semantic understanding without sacrificing the fine-grained local cues required by diverse downstream tasks.

Semantic Segmentation on ADE20k and Pascal VOC. For semantic segmentation task, we follow the evaluation protocols in [10]. As shown in Tab. 3, our method consistently improves semantic segmentation performance on both ADE20K and Pascal VOC under frozen feature evaluation. On ADE20K, our method achieves 47.9% mIoU with a linear classifier and 51.5% with multi-scale evaluation, compared with 47.3% and 51.3% for DINOv2 ViT-B/14. The performance gains are even more significant on Pascal VOC, where our method boosts the linear evaluation from 82.5% to 83.0%, and elevates the multi-scale performance from 84.9% to 85.9%. The supervised baselines perform worse than the original DINOv2 on both datasets. This result suggests that classification-oriented fine-tuning may weaken the local structures required for dense prediction. In contrast, GKPD combines generative class-token supervision with patch-token supervision, and successfully strengthens global semantics without compromising dense visual representations, preserving the spatial details and region-level consistency essential for pixel-level understanding tasks.

Instance-Level Recognition To further evaluate the capability of our frozen features on instance-level recognition, we evaluate our method on the Revisited Oxford and Revisited Paris image retrieval benchmarks. As reported in Tab. 4, our method consistently surpasses the DINOv2 ViT-B/14 baseline across both datasets and evaluation protocols. On the Hard split, which requires precise instance-level matching under challenging conditions such as viewpoint changes and occlusion, our method achieves

Table 3: **Semantic segmentation on ADE20K and Pascal VOC with frozen features** and a linear classifier (lin.) and with multiscale (+ms).

Method	Arch.	ADE20k		Pascal VOC	
		lin.	+ms	lin.	+ms
OpenCLIP	ViT-G/14	39.3	46.0	71.4	79.2
MAE	ViT-H/14	33.3	30.7	67.6	63.3
DINO	ViT-B/8	31.8	35.2	66.4	75.6
iBOT	ViT-L/16	44.6	47.5	82.3	84.3
DINOv2	ViT-S/14	44.3	47.2	81.1	82.6
	ViT-B/14	47.3	51.3	82.5	84.9
SFT	ViT-B/14	36.1	44.5	73.1	81.1
DeiT III	ViT-B/14	42.8	47.6	79.0	83.7
SD	ViT-B/14	35.5	43.4	72.8	80.5
GPKD	ViT-B/14	**47.9**	**51.5**	**83.0**	**85.9**

Table 4: **Evaluation of frozen features on instance-level recognition.** We report main metrics on Oxford and Paris benchmarks.

Method	Arch.	Oxford		Paris	
		M	H	M	H
OpenCLIP	ViT-G/14	50.7	19.7	79.2	60.2
MAE	ViT-H/14	11.7	2.2	19.9	4.7
DINO	ViT-B/8	40.1	13.7	65.3	35.3
iBOT	ViT-L/16	39.0	12.7	70.7	47.0
DINOv2	ViT-S/14	68.8	43.2	84.6	68.5
	ViT-B/14	72.9	49.5	90.3	78.6
SFT	ViT-B/14	52.1	24.8	74.0	51.5
DeiT III	ViT-B/14	51.1	26.7	78.2	58.9
SD	ViT-B/14	50.5	22.7	72.4	50.3
GPKD	ViT-B/14	**73.4**	**50.6**	**90.6**	**79.3**

50.6% on Oxford and 79.3% on Paris. This corresponds to absolute gains of 1.1 and 0.7 mAP over DINOv2, respectively. The supervised baselines also exhibit a substantial performance drop from the original DINOv2. These results validate that although our generation-to-perception distillation mainly uses a class token as a global teacher signal, the overall framework combined with the masked patch-level objective successfully preserves and enhances the distinctive, instance-specific structural information necessary for fine-grained image matching. We further provide visualization results of instance-level recognition in Appendix B.4.

4.4 Ablation Studies

Different Denoising Steps. We investigate the impact of the number of denoising steps used during class token generation. As shown in Tab. 5, we observe a phenomenon where fewer denoising steps yield better representation quality, with the optimal linear and k-NN accuracy achieved at a single step. This behavior follows from the joint structure of REG rather than from denoising the class token in isolation. At every solver step, the class-token velocity depends on both the current class token and the evolving image latent, together with the fixed class condition. With one step, the update is driven mainly by the global and perceptually salient structure available in the image latent. With more steps, the image latent is repeatedly updated toward its clean state, and later updates expose the class token to increasingly fine-grained and instance-specific information. These details are useful for image reconstruction, but they can shift a single global token away from the semantic manifold that is most useful for recognition. Nevertheless, the generated representations consistently outperform the DINOv2 ViT-B/14 baseline across all steps, validating the effectiveness of the approach.

Table 5: Ablation study on different denoising steps. All experiments are conducted on ImageNet-1K, using the ViT-B/14 backbone. We report linear and k-NN accuracy.

Step	1	50	100	150	200	250
linear	**85.4**	85.3	85.3	85.2	85.1	85.0
k-NN	**84.3**	84.2	84.1	84.0	84.0	83.8

Table 6: Ablation study on different loss terms of our method using linear evaluation across various benchmarks.

$\mathcal{L}_{cls}$	$\mathcal{L}_{patch}$	INet-1K	INet-Real	INet-V2	Food	C10	C100	SUN	Cars	Aircr	DTD	Pets	Cal101	Flowers	CUB
		84.5	88.3	75.1	92.8	98.7	91.3	77.3	88.2	79.4	83.3	96.2	**96.1**	99.6	89.6
✓		85.2	89.4	76.5	93.0	98.8	91.5	78.0	88.4	**80.8**	84.3	**96.3**	**96.1**	99.7	**89.9**
✓	✓	**85.4**	**89.6**	**76.8**	**93.3**	**98.9**	**91.8**	**78.7**	**88.6**	**80.8**	**84.8**	96.2	**96.1**	**99.7**	89.3

Effect of Training Loss To systematically evaluate the performance of each loss term in our framework, we conduct ablation studies and present the results of linear evaluation across multiple benchmarks in Tab. 6. Introducing only the global class token distillation loss $\mathcal{L}_{cls}$ leads to consistent improvements across most datasets. This demonstrates that the image-conditioned class tokens provide a strong global supervisory signal, improving the overall discriminative power of the learned representations. When $\mathcal{L}_{cls}$ and $\mathcal{L}_{patch}$ are jointly applied, the performance is further improved on the majority of benchmarks. The additional gains suggest that the patch token distillation complements the global supervision by enforcing finer-grained spatial consistency, which is particularly beneficial for structured and fine-grained recognition tasks such as SUN and DTD. Their combination leads to consistently stronger generalization and robustness across both in-distribution and out-of-distribution benchmarks, confirming the effectiveness of the proposed dual distillation strategy.

5 Conclusion

In this work, we investigated the semantic properties of class tokens generated by diffusion models and explored their potential for improving representation learning. We showed that the image-conditioned generated class token exhibits strong discriminative ability. Based on this observation, we proposed a generation-to-perception distillation framework that transfers the semantic knowledge of generated class tokens into pretrained visual representation models. Experimental results demonstrated that our method enables the model to learn robust representations and achieves state-of-the-art results. More importantly, this highlights a promising new direction where generative models are repurposed from data synthesizers into foundational vision learners for perception tasks. However, our current method does not explicitly explore the generation of patch tokens and their properties. We speculate that generated dense representations could provide stronger spatial guidance. In the future, we plan to extend this framework to all visual tokens for unified distillation.

Acknowledgments and Disclosure of Funding

This work was supported by the National Natural Science Fund of China (Nos. 62506155, U24A20330 and 62361166670), Provincial Natural Science Fund of Jiangsu (Nos. BK20251985), Suzhou Municipal Leading Talents Fund (Nos. ZXL2025320), ZTE Industry-University Project (Nos. IA20260910004), Alipay Industry-University Project (Nos. 2026101543758), and Key Project of the State Key Laboratory for Novel Software Technology (Nos. ZZKT2026A14).

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[2] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. In *International Conference on Learning Representations (ICLR)*, 2021.

[3] Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning (ICML)*, pages 8162–8171, 2021.

[4] Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *International Conference on Learning Representations (ICLR)*, 2023.

[5] George Stoica, Vivek Ramanujan, Xiang Fan, Ali Farhadi, Ranjay Krishna, and Judy Hoffman. Contrastive flow matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 1185–1194, 2025.

[6] Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. *International Conference on Learning Representations (ICLR)*, 2025.

[7] Xingjian Leng, Jaskirat Singh, Yunzhong Hou, Zhenchang Xing, Saining Xie, and Liang Zheng. Repa-e: Unlocking vae for end-to-end tuning of latent diffusion transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 18262–18272, 2025.

[8] Ziqiao Wang, Wangbo Zhao, Yuhao Zhou, Zekai Li, Zhiyuan Liang, Mingjia Shi, Xuanlei Zhao, Pengfei Zhou, Kaipeng Zhang, Zhangyang Wang, Kai Wang, and Yang You. REPA works until it doesn't: Early-stopped, holistic alignment supercharges diffusion training. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[9] Ge Wu, Shen Zhang, Ruijing Shi, Shanghua Gao, Zhenyuan Chen, Lei Wang, Zhaowei Chen, Hongcheng Gao, Yao Tang, Jian Yang, et al. Representation entanglement for generation: Training diffusion transformers is much easier than you think. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[10] Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy V. Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel HAZIZA, Francisco Massa, Alaaeldin El-Nouby, Mido Assran, Nicolas Ballas, Wojciech Galuba, Russell Howes, Po-Yao Huang, Shang-Wen Li, Ishan Misra, Michael Rabbat, Vasu Sharma, Gabriel Synnaeve, Hu Xu, Herve Jegou, Julien Mairal, Patrick Labatut, Armand Joulin, and Piotr Bojanowski. DINOv2: Learning robust visual features without supervision. *Transactions on Machine Learning Research*, 2024.

[11] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Bjorn Ommer. High-Resolution Image Synthesis with Latent Diffusion Models . In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10674–10685, 2022.

[12] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Raphael Gontijo-Lopes, Burcu Karagol Ayan, Tim Salimans, Jonathan Ho, David J. Fleet, and Mohammad Norouzi. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems (NeurIPS)*, 2022.

[13] Dustin Podell, Zion English, Kyle Lacey, Andreas Blattmann, Tim Dockhorn, Jonas Müller, Joe Penna, and Robin Rombach. SDXL: Improving latent diffusion models for high-resolution image synthesis. In *International Conference on Learning Representations (ICLR)*, 2024.

[14] Yang Song, Jascha Sohl-Dickstein, Diederik P Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations. In *International Conference on Learning Representations (ICLR)*, 2021.

[15] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10684–10695, 2022.

[16] William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 4195–4205, 2023.

[17] 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 *European Conference on Computer Vision (ECCV)*, pages 23–40, 2024.

[18] Zhengyang Geng, Mingyang Deng, Xingjian Bai, J Zico Kolter, and Kaiming He. Mean flows for one-step generative modeling. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[19] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International Conference on Machine Learning (ICML)*, pages 1597–1607, 2020.

[20] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9729–9738, 2020.

[21] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 33:21271–21284, 2020.

[22] Xinlei Chen and Kaiming He. Exploring simple siamese representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 15750–15758, 2021.

[23] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 9650–9660, 2021.

[24] Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

[25] Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. In *International Conference on Learning Representations (ICLR)*, 2021.

[26] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning (ICML)*, pages 12310–12320, 2021.

[27] Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[28] Haowen Cui, Shuo Chen, Jun Li, and Jian Yang. Enhancing contrastive learning with variable similarity. *Advances in Neural Information Processing Systems (NeurIPS)*, 2025.

[29] Haowen Cui, Shuo Chen, Jun Li, and Jian Yang. One-shot exemplars for class grounding in self-supervised learning. In *International Conference on Learning Representations (ICLR)*, 2026.

[30] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 16000–16009, 2022.

[31] Jinghao Zhou, Chen Wei, Huiyu Wang, Wei Shen, Cihang Xie, Alan Yuille, and Tao Kong. Image BERT pre-training with online tokenizer. In *International Conference on Learning Representations (ICLR)*, 2022.

[32] Zhenda Xie, Zheng Zhang, Yue Cao, Yutong Lin, Jianmin Bao, Zhuliang Yao, Qi Dai, and Han Hu. Simmim: A simple framework for masked image modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9653–9663, 2022.

[33] Runqian Wang and Kaiming He. Diffuse and disperse: Image generation with representation regularization. *arXiv preprint arXiv:2506.09027*, 2025.

[34] Dengyang Jiang, Mengmeng Wang, Liuzhuozheng Li, Lei Zhang, Haoyu Wang, Wei Wei, Guang Dai, Yanning Zhang, and Jingdong Wang. Representation alignment for diffusion transformers without external components. In *International Conference on Learning Representations (ICLR)*, 2026.

[35] Dmitry Baranchuk, Andrey Voynov, Ivan Rubachev, Valentin Khrulkov, and Artem Babenko. Label-efficient semantic segmentation with diffusion models. In *International Conference on Learning Representations (ICLR)*, 2022.

[36] Jiarui Xu, Sifei Liu, Arash Vahdat, Wonmin Byeon, Xiaolong Wang, and Shalini De Mello. Open-vocabulary panoptic segmentation with text-to-image diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023.

[37] Wenliang Zhao, Yongming Rao, Zuyan Liu, Benlin Liu, Jie Zhou, and Jiwen Lu. Unleashing text-to-image diffusion models for visual perception. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[38] Neehar Kondapaneni, Markus Marks, Manuel Knott, Rogério Guimaraes, and Pietro Perona. Text-image alignment for diffusion-based perception. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[39] Yuanfeng Ji, Zhe Chen, Enze Xie, Lanqing Hong, Xihui Liu, Zhaoqiang Liu, Tong Lu, Zhenguo Li, and Ping Luo. Ddp: Diffusion model for dense visual prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.

[40] Bingxin Ke, Anton Obukhov, Shengyu Huang, Nando Metzger, Rodrigo Caye Daudt, and Konrad Schindler. Repurposing diffusion-based image generators for monocular depth estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[41] Valentin Gabeur, Shangbang Long, Songyou Peng, Paul Voigtlaender, Shuyang Sun, Yanan Bao, Karen Truong, Zhicheng Wang, Wenlei Zhou, Jonathan T. Barron, Kyle Genova, Nithish Kannen, Sherry Ben, Yandong Li, Mandy Guo, Suhas Yogin, Yiming Gu, Huizhong Chen, Oliver Wang, Saining Xie, Howard Zhou, Kaiming He, Thomas Funkhouser, Jean-Baptiste Alayrac, and Radu Soricut. Image generators are generalist vision learners. *arXiv preprint arXiv:2604.20329*, 2026.

[42] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 248–255, 2009.

[43] Ilya Loshchilov, Frank Hutter, et al. Fixing weight decay regularization in adam. *arXiv preprint arXiv:1711.05101*, 2017.

[44] Hugo Touvron, Matthieu Cord, and Hervé Jégou. Deit iii: Revenge of the vit. In *European conference on computer vision (ECCV)*, 2022.

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*, pages 8748–8763, 2021.

[46] Mannat Singh, Laura Gustafson, Aaron Adcock, Vinicius de Freitas Reis, Bugra Gedik, Raj Prateek Kosaraju, Dhruv Mahajan, Ross Girshick, Piotr Dollár, and Laurens Van Der Maaten. Revisiting weakly supervised pre-training of visual perception models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 804–814, 2022.

[47] Mehdi Cherti, Romain Beaumont, Ross Wightman, Mitchell Wortsman, Gabriel Ilharco, Cade Gordon, Christoph Schuhmann, Ludwig Schmidt, and Jenia Jitsev. Reproducible scaling laws for contrastive language-image learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2818–2829, 2023.

[48] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[49] Priya Goyal, Mathilde Caron, Benjamin Lefaudeux, Min Xu, Pengchao Wang, Vivek Pai, Mannat Singh, Vitaliy Liptchinsky, Ishan Misra, Armand Joulin, et al. Self-supervised pretraining of visual features in the wild. *arXiv preprint arXiv:2103.01988*, 2021.

[50] Mahmoud Assran, Mathilde Caron, Ishan Misra, Piotr Bojanowski, Florian Bordes, Pascal Vincent, Armand Joulin, Mike Rabbat, and Nicolas Ballas. Masked siamese networks for label-efficient learning. In *European Conference on Computer Vision (ECCV)*, pages 456–473, 2022.

[51] Chunyuan Li, Jianwei Yang, Pengchuan Zhang, Mei Gao, Bin Xiao, Xiyang Dai, Lu Yuan, and Jianfeng Gao. Efficient self-supervised vision transformers for representation learning. In *International Conference on Learning Representations (ICLR)*, 2022.

[52] Pan Zhou, Yichen Zhou, Chenyang Si, Weihao Yu, Teck Khim Ng, and Shuicheng Yan. Mugs: A multi-granular self-supervised learning framework. *arXiv preprint arXiv:2203.14415*, 2022.

[53] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2961–2969, 2017.

[54] Zhaowei Cai and Nuno Vasconcelos. Cascade r-cnn: High quality object detection and instance segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 43(5):1483–1498, 2019.

[55] Lucas Beyer, Olivier J Hénaff, Alexander Kolesnikov, Xiaohua Zhai, and Aäron van den Oord. Are we done with imagenet? *arXiv preprint arXiv:2006.07159*, 2020.

[56] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *International Conference on Machine Learning (ICML)*, 2019.

[57] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision (ECCV)*, 2014.

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

[59] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2010.

[60] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE International Conference on Computer Vision Workshops (ICCVW)*, 2013.

[61] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

[62] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[63] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012.

[64] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2004.

[65] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. 2008.

[66] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. 2010.

[67] Bolei Zhou, Hang Zhao, Xavier Puig, Sanja Fidler, Adela Barriuso, and Antonio Torralba. Scene parsing through ade20k dataset. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

[68] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 2010.

[69] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European Conference on Computer Vision (ECCV)*, pages 740–755, 2014.

[70] Filip Radenović, Ahmet Iscen, Giorgos Tolias, Yannis Avrithis, and Ondřej Chum. Revisiting oxford and paris: Large-scale image retrieval benchmarking. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018.

A Algorithm of GPKD

Here, we provide the complete training procedure of GPKD in Algorithm 1.

Algorithm 1 GPKD Training Procedure

Input: Dataset $\mathcal{D}$ with ImageNet labels, VAE Encoder E_z, Class-conditioned Generative Model v_θ, Pretrained Teacher f_t, Student Encoder f_s, DINO Head g, Number of Prototypes K, Timestep t, Denoising Step Size Δt.
Hyperparams: Learning rate η, Temperatures τ_s, τ_t, Loss weights $\lambda_{cls}, \lambda_{patch}$.
Initialize: Student parameters from pretrained DINOv2.
Output: Optimized Student Encoder f_s.
for training step $= 1, 2, \ldots$ **do**

1. Image-Conditioned Class Token Generation:
Sample batch image-label pairs $(I, y) \sim \mathcal{D}$.
$z_0 \leftarrow E_z(I)$
Sample noise $\epsilon_z \sim \mathcal{N}(0, \mathbf{I})$, $\epsilon_{cls} \sim \mathcal{N}(0, \mathbf{I})$.
$z_t \leftarrow \alpha_t z_0 + \sigma_t \epsilon_z$ {*Eq. 7*}
$h_t^{cls} \leftarrow \epsilon_{cls}$ {*Eq. 8*}
while $t > 0$ **do**
$v_z(t), v_{cls}(t) \leftarrow v_\theta([h_t^{cls}; z_t], t, y)$
$z_{t-\Delta t} \leftarrow z_t - \Delta t \cdot v_z(t)$
$h_{t-\Delta t}^{cls} \leftarrow h_t^{cls} - \Delta t \cdot v_{cls}(t)$ {*Eq. 9*}
$t \leftarrow t - \Delta t$
end while
$\hat{h}_0^{cls} \leftarrow h_0^{cls}$

2. View Construction & Feature Extraction:
$I_{aug} \leftarrow \mathcal{T}(I)$
$\{h_{t,i}^p\}_{i=1}^N \leftarrow f_t(I_{aug})$
$h_s^{cls}, \{h_{s,i}^p\}_{i=1}^N \leftarrow f_s(I_{aug})$

3. Global Class Token Distillation:
$u_t^{cls} \leftarrow g(\hat{h}_0^{cls}), \quad u_s^{cls} \leftarrow g(h_s^{cls})$
$q_t^{cls} \leftarrow \text{Sinkhorn}(u_t^{cls}/\tau_t), \quad p_s^{cls} \leftarrow \text{Softmax}(u_s^{cls}/\tau_s)$
$\mathcal{L}_{cls} \leftarrow -\sum_{k=1}^K q_{t,k}^{cls} \log p_{s,k}^{cls}$ {*Eq. 10*}

4. Masked Patch-Level Distillation:
Sample maksed subset $M \subseteq \{1, \ldots, N\}$
$\{u_{t,i}^p, u_{s,i}^p\}_{i \in M} \leftarrow \{g(h_{t,i}^p), g(h_{s,i}^p)\}_{i \in M}$
$\{q_{t,i}^p, p_{s,i}^p\}_{i \in M} \leftarrow \{\text{Sinkhorn}(u_{t,i}^p/\tau_t), \text{Softmax}(u_{s,i}^p/\tau_s)\}_{i \in M}$
$\mathcal{L}_{patch} \leftarrow -\frac{1}{|M|} \sum_{i \in M} \sum_{k=1}^K q_{t,i,k}^p \log p_{s,i,k}^p$ {*Eq. 11*}

5. Parameter Update:
$\mathcal{L} \leftarrow \lambda_{cls}\mathcal{L}_{cls} + \lambda_{patch}\mathcal{L}_{patch}$ {*Eq. 12*}
Update f_s via gradient descent to minimize $\mathcal{L}$
end for

B Additional Experiments

B.1 Object Detection and Instance Segmentation on COCO.

To evaluate the transferability of our learned representations, we conduct experiments on object detection and instance segmentation using the COCO benchmark. Based on [31], we use a Cascade Mask R-CNN detector [53, 54] to produce bounding boxes and instance masks. As shown in Tab. 7, our proposed distillation method consistently outperforms both supervised baselines and state-of-the-art self-supervised methods across different model sizes. Specifically, under the ViT-S/14

architecture, our method achieves 50.8 AP^b and 43.6 AP^m. This improvement scales effectively to the larger ViT-B/14 backbone, where our model obtains 54.2 AP^b and 46.6 AP^m. These results clearly demonstrate that coupling the global generative teacher signal with patch-level distillation, the student model inherits refined category-level semantics while successfully preserving and even enhancing the fine-grained localization capabilities required for complex region-level tasks.

Table 7: **Object detection and instance segmentation on COCO** using a Cascade Mask R-CNN detector. Performance is reported as bounding box mAP (AP^b) for detection and instance mask mAP (AP^m) for segmentation.

Method	Arch.	Object Detection (AP^b)	Instance Segmentation (AP^m)
Supervised	Swin-T	48.1	41.7
MoBY	Swin-T	48.1	41.5
Supervised	ViT-S/16	46.2	40.2
iBOT	ViT-S/16	49.4	42.6
Mugs	ViT-S/16	49.8	43.0
DINOv2	ViT-S/14	<u>50.5</u>	<u>43.3</u>
GPKD	ViT-S/14	**50.8**	**43.6**
Supervised	ViT-B/16	49.8	43.2
BeiT	ViT-B/16	50.1	43.5
DINO	ViT-B/16	50.1	43.4
iBOT	ViT-B/16	51.2	44.2
DINOv2	ViT-B/14	<u>53.8</u>	<u>46.2</u>
GPKD	ViT-B/14	**54.2**	**46.6**

B.2 Sampling Methods.

We also compare the performance of using an ODE sampler versus an SDE sampler during the denoising process of class token generation. As reported in Tab. 8, both sampling methods perform comparably, with the ODE sampler holding a marginal edge. This suggests that the quality of the generated class token is relatively robust to the stochasticity introduced by SDE, and a deterministic ODE trajectory is highly sufficient for extracting high-quality discriminative signals. We hypothesize that injecting randomness into a highly condensed class token may introduce slight semantic perturbations that marginally dilute its discriminative power.

Table 8: Ablation study on different sampling methods during class token generation. We report performance using either an SDE sampler or ODE sampler.

Sampler	ODE	SDE
linear	85.4	85.3
k-NN	84.3	84.3

B.3 Different Noise Scales.

Tab. 9 ablates the effect of the initial noise scale (σ_t) applied to the image latent when guiding the class token generation. We observe a clear positive correlation between the noise intensity and downstream performance: larger noise scales yield stronger, more discriminative features. A higher noise scale forces the generative model to rely more heavily on robust, global semantic structural priors to reconstruct the class token, rather than trivially extracting low-level cues from a relatively clean image latent. Therefore, a high noise scale effectively acts as an information bottleneck, purifying the distilled semantics.

Table 9: Ablation study on different noise scales during class token generation. All experiments are conducted on ImageNet-1K, using the ViT-B/14 backbone. We report linear and k-NN accuracy.

Scale	0.99	0.95	0.9	0.85	0.8
linear	**85.4**	85.2	85.0	84.7	84.4
k-NN	**84.3**	84.2	84.0	83.6	83.0

B.4 Visualization of Instance-Level Recognition

To provide a more intuitive understanding of our method's superiority, we visualize the top-5 retrieval results for several challenging queries in Fig. 4. As depicted, our method consistently retrieves more

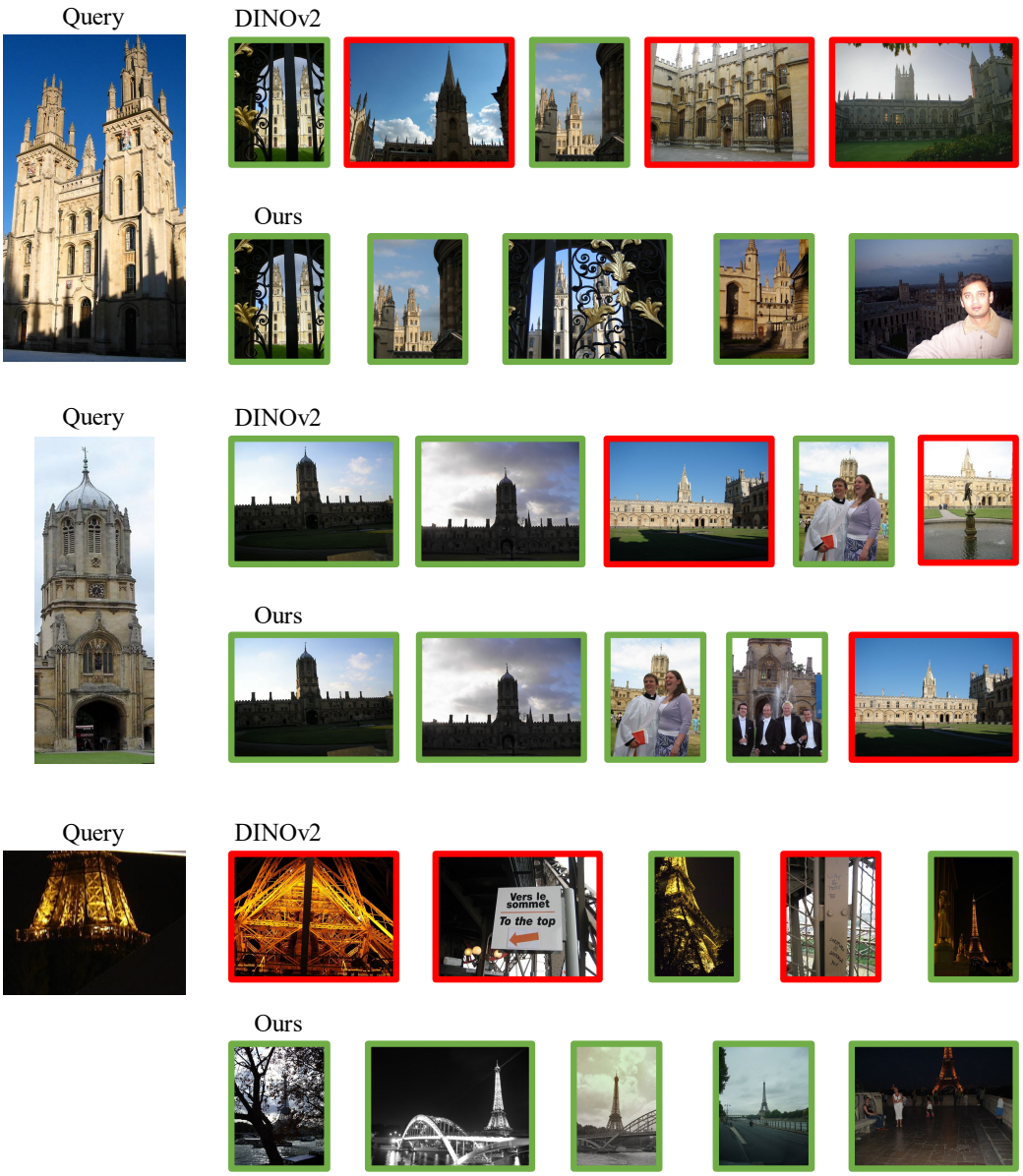

Figure 4: Instance-level recognition results. Correct and incorrect results are marked with green and red borders, respectively.

accurate matches. These qualitative results align with our quantitative findings, confirming that our method captures highly discriminative instance-level semantic representations.

C Broader Impacts

This work presents a novel perspective that generative models can serve as powerful foundation learners for visual perception, demonstrating how transferring generative representations can systematically enhance pretrained visual models.

D Computational Cost and Training Efficiency

Our proposed method introduces no structural modifications to the student network architecture. Consequently, the model parameter count and inference GPU memory remain exactly identical to the standard DINOv2 baseline. Furthermore, because our approach leverages generation-to-perception distillation rather than self-supervised pretraining from scratch, the training process is highly efficient. In practice, the generated class tokens are computed offline, which takes approximately 0.5 hours. Distilling the student model takes approximately 7.5 hours using a total batch size of 512 on a single node equipped with 8 NVIDIA RTX 4090 GPUs. This demonstrates that our framework can significantly enhance representation quality and downstream performance with a remarkably modest computational budget.

E List of Datasets Used

In Tab. 10, we present the list of datasets used for distillation and evaluation.

Table 10: List of datasets used for distillation and evaluation.

Dataset	Distillation	Evaluation	Task	Citation
ImageNet-1K	✓	✓	Classification	[42]
ImageNet-Real	✗	✓	Classification	[55]
ImageNet-V2	✗	✓	Classification	[56]
Food-101	✗	✓	Classification	[57]
CIFAR-10	✗	✓	Classification	[58]
CIFAR-100	✗	✓	Classification	[58]
SUN397	✗	✓	Classification	[59]
StanfordCars	✗	✓	Classification	[60]
FGVC-Aircraft	✗	✓	Classification	[61]
DTD	✗	✓	Classification	[62]
Oxford Pets	✗	✓	Classification	[63]
Caltech101	✗	✓	Classification	[64]
Flowers	✗	✓	Classification	[65]
CUB200	✗	✓	Classification	[66]
ADE20k	✗	✓	Segmentation	[67]
VOC 2012	✗	✓	Segmentation	[68]
COCO	✗	✓	Detection & Segmentation	[69]
Paris	✗	✓	Retrieval	[70]
Oxford	✗	✓	Retrieval	[70]

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 contributions and scope of this paper are accurately reflected in the abstract and introduction.

   Guidelines:

   - The answer [N/A] 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 [N/A] 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: In section 5, we discuss the limitations of our work and identify them as valuable future work.

   Guidelines:

   - The answer [N/A] 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: [N/A]

Justification: This paper does not include theoretical results.

Guidelines:

- The answer [N/A] 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: We claim the details of methods and the experiments settings in our paper.

   Guidelines:

   - The answer [N/A] 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: Code [Yes] Data [No]

Justification: We will soon open-source the code, and the datasets used are publicly available for download by anyone.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.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://neurips.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) necessary to understand the results?

   Answer: [Yes]

   Justification: All detailed specifications are presented in the experiment section.

   Guidelines:

   - The answer [N/A] 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: We lack sufficient computational resources to calculate error bars for every experiment.

   Guidelines:

   - The answer [N/A] 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: [Yes]

   Justification: See the Appendix D.

   Guidelines:

   - The answer [N/A] 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: We confirm that our research conducted in the paper conforms with the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] 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: [Yes]

    Justification: See the Appendix C.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] 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., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: The paper poses no such risks.

    Guidelines:

    - The answer [N/A] 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 cite and comply with the licenses of the public datasets we use.

    Guidelines:

    - The answer [N/A] 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: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

    Guidelines:

    - The answer [N/A] 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: [N/A]

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

    Guidelines:

    - The answer [N/A] 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 rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The paper does not involve the usage of LLMs.

Guidelines:

- The answer [N/A] 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 in the NeurIPS handbook for what should or should not be described.