Denoising Vision Transformers

Jiawei Yang^{*,†,1,*} Katie Z Luo^{*,2} Jiefeng Li³ Congyue Deng⁴ Leonidas Guibas⁴ Dilip Krishnan⁵ Kilian Q Weinberger² Yonglong Tian⁵ Yue Wang¹

> ¹University of Southern California ²Cornell University ³Shanghai Jiaotong University ⁴Stanford University ⁵Google Research *equal technical contribution [†]project lead



Fig. 1: **Denoising Vision Transformers (DVT)** effectively suppresses noisy artifacts in the visual features of all Vision Transformers (ViTs) we have tested and improves performance on a broad spectrum of dense prediction tasks, including semantic segmentation, depth estimation, object detection, and object discovery. Our evaluation encompasses a representative set of ViTs, including DINOv2 [23], DeiT-III [32], EVA-02 [11], CLIP [24], and DINOv2-reg [6]. We visualize the features before and after DVT, colored via principal component analysis (PCA). Best viewed in color. **Right**: We report the downstream dense prediction task performances, averaged over all models.

Abstract. We study a crucial yet often overlooked issue inherent to Vision Transformers (ViTs): feature maps of these models exhibit grid-like artifacts ("Original features" in Fig. 1), which hurt the performance of ViTs in downstream dense prediction tasks such as semantic segmentation, depth prediction, and object discovery. We trace this issue down to the positional embeddings at the input stage. To mitigate this, we

^{*} Work partially completed while interning at Google Research

propose a two-stage denoising approach, termed Denoising Vision Transformers (DVT). In the first stage, we separate the clean features from those contaminated by positional artifacts by enforcing cross-view feature consistency with neural fields on a per-image basis. This per-image optimization process extracts artifact-free features from raw ViT outputs, providing clean feature estimates for offline applications. In the second stage, we train a lightweight transformer block to predict clean features from raw ViT outputs, leveraging the derived estimates of the clean features as supervision. Our method, DVT, does not require retraining the existing pre-trained ViTs, and is immediately applicable to any Vision Transformer architecture. We evaluate our method on a variety of representative ViTs (DINO, DeiT-III, EVA02, CLIP, DINOv2, DINOv2-reg) and demonstrate that DVT consistently improves existing state-of-the-art general-purpose models in semantic and geometric tasks across multiple datasets (Fig. 1, right, Tabs. 2 to 4). We hope our study will encourage a re-evaluation of ViT design, especially regarding the naive use of positional embeddings. Our code and checkpoints are publicly available in our project page.

1 Introduction

In recent years, Transformers [34] have emerged as the universal architecture for modern foundation models across many modalities, from text [1,5,25,27] to audio [18,37], and images [2,8]. Among these, Vision Transformers (ViTs) [8] trained at scale not only achieve state-of-the-art under multiple benchmarks but also exhibit intriguing behaviors and capabilities across various tasks [3, 14, 23, 24].

Despite these significant strides made by ViTs, our work reveals a crucial yet often overlooked challenge: the presence of persistent noise artifacts in ViT outputs, observable across various training algorithms [3,8,11,23,24,32] (illustrated in Fig. 1 left). These artifacts not only compromise visual clarity but also hinder feature interpretability and disrupt semantic coherence. For example, Fig. 2 demonstrates that applying clustering algorithms directly on the raw ViT output results in noisy clusters, and the patch feature similarity is less reliable. Additionally, these artifacts are frequently concealed by *seemingly* impressive performance on downstream tasks, thus evading thorough examination or detection by the research community. Addressing these artifacts can unleash the potential of pre-trained ViTs and lead to substantial performance improvements (Fig. 1 right). Therefore, our work aims to answer a crucial research question: *Is it feasible to effectively denoise these artifacts in pre-trained ViTs, ideally without model retraining*?

To answer this, we first investigate the origins of these artifacts. We hypothesize that positional embeddings, a fundamental component of ViT architecture, play a pivotal role in the emergence of these artifacts. Our initial analysis supports this hypothesis: *First*, when a zero tensor is fed into a pre-trained DINOv2 model [23], the resulting output is predominantly characterized by similar noise patterns (Fig. 3-(a, 2)). *Second*, we observe the absence of such artifacts in the



Fig. 2: Artifacts hurt semantic coherence. For each triplet, we show a feature map, a K-Means cluster map, and a similarity map of the central patch (red dotted) with other patches in the image. Observe how artifacts negatively impact clustering accuracy and similarity correspondences, and how our denoising mitigates these issues.

outputs of a DINOv2 model trained without positional embeddings, which contrasts sharply with the standard model outputs (Fig. 3-(a, 1) v.s. (a, 3)). *Third*, take a video with continuous frames as an example (Fig. 3-(c)). Despite the significant differences in the context of various input frames, the artifacts maintain a generally consistent relative position in the images (Fig. 3-(c), middle row).

With these insights, we present a two-stage denoising approach, Denoising Vision Transformers (DVT), to suppress artifacts in pre-trained ViTs. In the first stage, we obtain clean features from contaminated ones by enforcing cross-view feature consistency and artifact consistency with neural fields on a per-image basis. This per-image denoising process extracts noise-free features from raw output, providing these denoised ViT features for offline applications. In the second stage, we train a lightweight denoiser model, consisting of a single transformer block, to predict the denoised features from the raw ViT outputs. More importantly, this denoiser can be seamlessly integrated into pre-trained ViTs without extensive *re-training*, providing denoised features for online applications and generalizing well to unseen data.

We conduct empirical evaluations to demonstrate the efficacy of DVT on six representative ViTs: DINO [3], DINOv2 [23], DINOv2 with Register (DINOv2reg) [6], DeiT-III [32], EVA-02 [11,12], and CLIP [24]. These evaluations demonstrate significant improvements in performance across various dense prediction vision tasks such as semantic segmentation, depth estimation, object detection, and object discovery. In summary, our contributions are:

- We identify and highlight the widespread occurrence of noise artifacts in ViT features, pinpointing positional embeddings as a crucial underlying factor. To the best of our knowledge, we are the first to provide such an analysis.
- We introduce a tailored noise model for ViTs, along with a neural field based denoising technique. This combination effectively isolates and removes noise artifacts from ViT features.
- We develop a flexible and efficient denoiser that integrates seamlessly with pre-trained ViTs, enabling real-time applications.
- Our approach results in substantial performance improvements across various ViTs and downstream dense prediction tasks (Fig. 1, right, Tabs. 2 to 4).



Fig. 3: Impact of positional embeddings in ViTs. (a) Comparison between DI-NOv2 ViTs [23] trained with and without positional embeddings (("ViT" v.s. "ViT""). We show feature maps from (1) a standard ViT, (2) a ViT using only positional embeddings (PE) as input, emphasizing the emergence of artifacts, and (3) a PE-free ViT^{*}, displaying a clear absence of these artifacts. In the figure, "Patch": patch embedding, "PE": position embedding. (b) Illustration of how ViT retains and propagates the positional embeddings. (c) Despite significant differences in the context of various frames, the artifacts largely maintain a consistent relative position in the images (central row). Our DVT effectively denoises these artifacts, demonstrated in the final row.

2 Related Works

General purpose features from Vision Transformers. Transformers have been used extensively across multiple domains as general-purpose feature extractors [1,5,7,26,27,33]. Vision Transformers [8] (ViTs) pre-trained via supervised learning [16,32,35] or self-supervised learning [3,14,23,42] have demonstrated strong generalizability to various downstream visual tasks, even without finetuning. However, we show that ViTs trained with diverse training objectives exhibit commonly observed noise artifacts in their output feature maps. These artifacts are often overlooked in practice because their presence cannot be simply reflected by image *classification* accuracy. Thus, our work focuses on evaluating pre-trained ViTs for *dense recognition* tasks such as segmentation, depth estimation, and object discovery. We demonstrate how these artifacts adversely affect dense recognition tasks, thereby motivating our method to mitigate them.

ViT artifacts. Our work studies the noise artifacts in ViTs, an issue that has been previously observed but often remains unexplored. These artifacts manifest as noisy attention maps in supervised ViTs (*i.e.*, ViTs do not attend to objects of interest well) [3, 4]. Concurrently with our study, two recent studies similarly have also identified artifacts in self-supervised ViTs [6, 40]. Specifically, [6] describe these as "high-norm" patches in low-informative background regions, hypothesizing their occurrence is limited to large (*e.g.* ViT-*large* or greater) and sufficiently trained ViTs. However, our analysis indicates that this may not be the *full picture*, as we observe similar artifacts in small or base ViTs that cannot be easily identified by extremely high feature norm values. Instead, we find a strong correlation between the presence of artifacts and the use of positional embeddings in ViTs. This finding suggests that artifacts are not strictly confined to certain model sizes or training scales but are more fundamentally linked to the inherent design of ViTs. Moreover, unlike the method proposed by [6] that retrains ViTs with register tokens [13,39] from scratch, our approach directly denoises pre-trained models without retraining. Users can *dynamically* enable or disable the plugged-in denoiser as needed. Lastly, we note that some *weak* artifacts still exist in DINOv2 models trained with registers [6] (see Fig. 1 DINOv2-reg and appendix), and our DVT can effectively denoise them, improving the performance of DINOv2-reg.

3 Preliminaries

Forward process in ViTs. Despite varying training approaches, the ViT architecture has mostly remained consistent with its original design as presented in [8] and [35]. The forward process of a ViT, depicted in Fig. 3-(b), starts by converting images into 2D patches and then embedding them, followed by a forward process of Transformer blocks. Specifically, an image $\mathbf{x} \in \mathbb{R}^{H \times W \times C}$ is first divided into patches $\mathbf{x}_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$, where (H, W) denotes the image resolution, P is the patch resolution, C represents the number of pixel channels and N is the number of total patches. These patches are then mapped to Ddimensions using a trainable linear projection $\mathbf{E} \in \mathbb{R}^{(P^2 \cdot C) \times D}$ to generate patch embeddings. To inject spatial information, positional embeddings, which encode patch coordinates and are denoted \mathbf{E}_{pos}^i , are added to the patch embeddings. Formally, the forward process of a ViT is as follows:

$$\mathbf{z}_{0} = [\mathbf{x}_{cls} + \mathbf{E}_{pos}^{cls}; \mathbf{x}_{p}^{0}\mathbf{E} + \mathbf{E}_{pos}^{0}; \cdots; \mathbf{x}_{p}^{N-1}\mathbf{E} + \mathbf{E}_{pos}^{N-1}]$$
(1)

$$\mathbf{z}'_{l} = \mathrm{MSA}\left(\mathrm{LN}(\mathbf{z}_{l-1})\right) + \mathbf{z}_{l-1}, \quad l = 1 \cdots L$$
(2)

$$\mathbf{z}_{l} = \mathrm{MLP}\left(\mathrm{LN}(\mathbf{z}'_{l})\right) + \mathbf{z}'_{l}, \qquad l = 1 \cdots L$$
(3)

$$\mathbf{y} = \mathrm{LN}(\mathbf{z}_L) \tag{4}$$

Here, \mathbf{x}_{cls} and \mathbf{E}_{pos}^{cls} represent the class token and its positional embedding, respectively, L denotes the number of layers, and LN stands for layer normalization. Multi-head self-attention layers and multi-layer perceptron layers are termed MSA and MLP, respectively. Note how the *input-independent* positional embeddings function as a spatial inductive basis, intermixing with inputs and propagating throughout ViT.

4 Denoising Vision Transformers

In this section, we start by analyzing ViT outputs to motivate our approach (§4.1). Then, we introduce our per-image denoising method, which removes artifacts and produces noise-free features (§4.2). Lastly, we explain how the noise-free features are utilized as pseudo-labels to train a generalizable denoiser (§4.3). Our method pipeline is depicted in Fig. 4.



Fig. 4: **Method Overview.** DVT consists of a two-stage denoising pipeline. (a) In the first stage, our method decomposes the raw feature of an image crop into a noise-free semantics term \mathcal{F} , an input-independent, position-related artifact term \mathcal{G} , and an additional residual term Δ . (b) In the second stage, we train a generalizable denoiser to predict clean features from their original features. At inference time, only a single feedforward is needed to obtain denoised features.

4.1 Factorizing ViT Outputs

Our method is grounded in the principle that ideal visual features should be inherently translation and reflection invariant, *i.e.*, the features of an object should remain consistent, regardless of changes in the viewing window, size, and orientation. However, as indicated in Eqs. (1) to (4) and Fig. 3-(b), ViTs intermix patch embeddings with positional embeddings, thereby breaking the transformation invariance of visual features. This breach of invariance might not appear immediately problematic, but our investigations, illustrated in Fig. 3-(a) and (c), reveal a distinct correlation between the inclusion of positional embeddings and the emergence of undesirable artifacts in ViT outputs. Particularly, the middle row of Fig. 3-(c) shows that these artifacts persist with minor variation across different images, highlighting their consistency independent of the input content.

These observations motivate us to decompose ViT outputs into three terms: (1) an input-dependent, noise-free semantics term $f(\mathbf{x})^1$; (2) an input-independent artifact term related to spatial positions $g(\mathbf{E}_{pos})$; (3) and a residual term that accounts for the interdependence of semantics and positions $h(\mathbf{x}, \mathbf{E}_{pos})$. The decomposition is formally expressed as:

$$ViT(\mathbf{x}) \approx f(\mathbf{x}) + g(\mathbf{E}_{pos}) + h(\mathbf{x}, \mathbf{E}_{pos})$$
(5)

This factorization is universally applicable to all ViTs. For example, in scenarios where the output feature map is spatially invariant (e.g., no positional

¹ Throughout this paper, we use "noise" and "artifact" interchangeably.

embedding is used), the sum of g and h becomes a constant bias term that can be merged into f.

4.2 Per-image Denoising with Neural Fields

Directly addressing the above decomposition problem within a single forward pass in a ViT is impractical due to the intertwined nature of output features. To overcome this, we exploit the consistencies in cross-view features and artifacts: (1) Feature consistency refers to the transformation invariance of visual features, where despite the varied spatial transformations, the semantic content remains invariant; (2) Artifact consistency means that the input-independent artifact remains observable and constant across all transformations. Formally, consider an image \mathbf{x} and a set of its randomly transformed views $T(\mathbf{x}) = \{t_0(\mathbf{x}), t_1(\mathbf{x}), \cdots\}$, where each transformation t_i is drawn from a distribution of random augmentations \mathcal{T} , consisting of random resizing, cropping, and flipping. Our goal is to derive a mapping f such that the semantic features obtained from any transformed view, $f(t(\mathbf{x}))$, are equivalent to the transformed original semantic features, $t(f(\mathbf{x}))$; that is, $f(t(\mathbf{x})) = t(f(\mathbf{x}))$ with $t \sim \mathcal{T}$. Next, we describe our approach for learning the different terms in Eq. (5) in conjunction to derive f.

Neural fields as feature mappings. At the core of our approach is to have a holistic image semantics representation \mathcal{F} , for each individual image, alongside a spatial artifact feature representation, \mathcal{G} , shared by all transformed views. The holistic image feature representation \mathcal{F} is designed to capture spatially independent, artifact-free semantics, while \mathcal{G} should encode position-dependent but input-independent noise. We use coordinate networks, known as neural fields [15,17,21,29,31,40], to actualize \mathcal{F} and \mathcal{G} . Specifically, we define $f(t(\mathbf{x})) =$ $\mathcal{F}(\operatorname{coords}(t(\mathbf{x})))$, where $\operatorname{coords}(\cdot)$ extracts the pixel coordinates of the transformed views relative to the original image \mathbf{x} , and $g(\mathbf{E}_{pos}^i) = \mathcal{G}(i)$, with $i \in$ $\{0, \dots, N-1\}$ denoting the patch index. For simplicity, we use \mathcal{G} to denote the 2D artifact feature map reshaped from the 1D ordered sequence $\{\mathcal{G}(i)\}_{i=0}^{N-1}$. We refer to \mathcal{F} and \mathcal{G} as the semantics field and the artifact field, respectively.

Learning the decomposition. We learn the semantics field \mathcal{F} , the artifact field \mathcal{G} , and the residual term Δ by minimizing a regularized reconstruction loss:

$$\mathcal{L}_{\text{recon}} = \mathcal{L}_{\text{distance}} + \alpha \mathcal{L}_{\text{residual}} + \beta \mathcal{L}_{\text{sparsity}}$$
(6)

$$\mathcal{L}_{\text{distance}} = 1 - \cos(\mathbf{y}, \widehat{\mathbf{y}}) + \|\mathbf{y} - \widehat{\mathbf{y}}\|_2, \tag{7}$$

$$\mathcal{L}_{\text{residual}} = \|\text{sg}\left(\mathbf{y} - \widehat{\mathbf{y}'}\right) - \widehat{\Delta}\|_2, \quad \mathcal{L}_{\text{sparsity}} = \|\widehat{\Delta}\|_1 \tag{8}$$

where
$$\mathbf{y} = \operatorname{sg}\left(\operatorname{ViT}\left(t\left(\mathbf{x}\right)\right)\right), \qquad \widehat{\mathbf{y}} = \mathbf{y}' + \operatorname{sg}(\Delta)$$
 (9)

$$\mathbf{y}' = \mathcal{F}_{\theta}(\operatorname{coords}(t(\mathbf{x}))) + \mathcal{G}_{\xi}, \quad \Delta = h_{\psi}(\mathbf{y})$$
 (10)

Here, $\cos(\cdot, \cdot)$ denotes the cosine similarity, $\operatorname{sg}(\cdot)$ represents the stop-gradient operation, $t(\cdot)$ is a random transformation sampled from \mathcal{T} , and θ , ξ and ψ are the learnable parameters. Our loss function is designed to encourage $\widehat{\Delta}$ to remain

minimal by imposing a sparsity regularization, thereby allowing $\hat{\mathbf{y}'}$ to represent as much of the ViT output as possible. The use of stop-gradient operators is to avoid trivial solutions, such as identity mapping. The reconstructed feature from our method is $\hat{\mathbf{y}} = \mathcal{F}_{\theta} (\operatorname{coords} (t(\mathbf{x}))) + \mathcal{G}_{\xi} + \operatorname{sg} (h_{\psi} (\operatorname{ViT} (t(\mathbf{x})))))$, each term corresponding to f, g, and h as defined in Eq. (5).

Optimization. We break our optimization process into two phases, each spanning half of the total training iterations. In the first phase, we train \mathcal{F}_{θ} and \mathcal{G}_{ξ} using only $\mathcal{L}_{\text{distance}}$, allowing them to capture a significant portion of the ViT outputs. After completing half of the optimization iterations, we freeze \mathcal{G}_{ξ} and continue to train \mathcal{F}_{θ} alongside h_{ψ} using $\mathcal{L}_{\text{recon}}$ for the rest iterations. The coefficients α and β in $\mathcal{L}_{\text{recon}}$ balance loss scales and regulate the residual term to prevent $\widehat{\Delta}$ from over-explaining the outputs.

4.3 Generalizable Denoiser

Our per-image denoising method can already effectively remove artifacts from ViT outputs, yielding visually stunning denoised feature maps. The problems we are left with are run-time efficiency and distribution shifts. Specifically, the per-image denoising process is suitable for offline applications but undesired for real-time applications, and individually denoised feature maps can lead to feature distribution shifts due to sample bias, which hampers the feature coherence across images. To address these issues, we introduce a generalizable denoiser.

After applying per-image denoising, we accumulate a dataset of pairs consisting of noisy ViT outputs \mathbf{y} and their denoised counterparts \mathcal{F} , denoted as $\mathcal{B} = \{(\mathbf{y}_i, \mathcal{F}_i)\}_{i=1}^B$. We then train a denoiser network D_{ζ} to predict noise-free features from raw ViT outputs, *i.e.*, $\hat{\mathcal{F}} = D_{\zeta}(\mathbf{y})$. The loss function is:

$$\mathcal{L}_{\text{distance}}^{\text{DVT}} = 1 - \cos\left(D_{\zeta}\left(\mathbf{y}\right), \mathcal{F}\right) + \|D_{\zeta}\left(\mathbf{y}\right) - \mathcal{F}\|_{2}$$
(11)

Our generalizable denoiser is implemented as a single Transformer block, supplemented with additional learnable positional embeddings that are applied post the forward pass of a ViT. This design aims to mitigate the input-independent artifacts. To predict denoised features, the outputs from a pre-trained ViT are added with these positional embeddings and then processed through the Transformer block.

Notably, this learned denoiser is lightweight, thus adding negligible latency to the original ViT and facilitating real-time applications. It also learns to generalize across samples, mitigating the distribution shift issue in the per-image denoising process.

5 Experiments

In this section, we first explore if ViTs trained with different objectives all have artifacts. Then, we evaluate the effectiveness of our generalizable denoiser on dense prediction tasks. For all experiments, we default to using ViT-*base* models



Fig. 5: Visual analysis of ViT output features and denoised features. (a) Visualizations of the feature maps from all layers of a DINOv2 [23] ViT-*base* model. Notably, the artifacts in the feature maps derived from the cat image exhibit a strong visual correlation with those from the zero-tensor inputs. (b) Visualizations of the decomposed artifacts, the original features, and the denoised features across various layers of DINOv2 ViTs. We observe similar patterns in differently-sized models.

with patch sizes of 14 or 16, depending on the availability of their implementations and model weights in PyTorch Image Models (timm [38]). We defer all the implementation details to the appendix.

5.1 Artifacts in ViTs

Positional artifacts in different ViTs. We visualize feature maps from differently pre-trained ViTs in Fig. 1. Among these, DINOv2 [23], a state-of-the-art vision foundation model with excellent performance on downstream tasks, displays clear position-related artifacts. Additionally, DeIT-III [32], trained with image class labels, and CLIP [24], trained by text-image alignment, also exhibit noticeable artifacts. Furthermore, EVA02 [11], which distills local patch features from a pre-trained CLIP model using masked image modeling, also has clear feature artifacts. In ViTs we have tested, our proposed DVT successfully mitigates these artifacts ("Original features" vs. "Denoised features" in Fig. 1).

Artifacts in different layers. In Fig. 5, we present a visual analysis of the artifact decomposition across various layers of DINOv2 ViTs of different sizes (b), alongside feature maps generated using only zero-tensors as input (a). Notably, the artifacts decomposed by our DVT show a strong visual resemblance to these zero-tensor-input feature maps. In addition, we observe that the artifacts vary across layers: the shallower layers predominantly exhibit low-frequency patterns, whereas the deeper layers are characterized by high-frequency patterns. Importantly, these patterns are consistent across ViTs of different sizes (*e.g.*, from

Table 1: Comparison of features correlation to spatial positions. We report the maximal information coefficient (MIC) between grid features and their coordinates.

	Before denoising	After denoising		
Method	Raw features	Artifacts term \mathcal{G}	Semantics term \mathcal{F}	
DINOv2 [23]	0.44	0.54	0.22	
DeiT-III [32]	0.34	0.32	0.06	
CLIP [24]	0.11	0.14	0.08	

ViT-*small* to ViT-*large*), diverging from the hypothesis in [6] that only large ViTs would display such patterns.

Correlation between artifacts and positions. Beyond visual qualitative inspection, we aim to quantitatively analyze the correlation between artifacts and their positions. Similar to [36], we use the maximal information coefficient (MIC) to measure the dependency between grid features and their normalized patch coordinates (See appendix for more details). This metric indicates how much patch features depend on their spatial positions and semantic content. As shown in Tab. 1, both the original ViT outputs and the decomposed artifacts exhibit a higher spatial correlation than the denoised semantic features, irrespective of the training methodology employed. These results support our hypothesis about the significant role of positional embeddings in the emergence of artifacts. Note that there is no "ground-truth" quantitative metric to to definitively quantify these patterns; hence, our reported numerical results should be viewed as empirical indicators, akin to the "high-norm" indicator used in [6].

5.2 Evaluation on Downstream Task Performance

We evaluate our method in dense recognition tasks, including semantic segmentation, monocular depth estimation, object detection, and object discovery. It is important to note that there is no direct competitor for these tasks in our study. Instead, our focus is on comparing the performance of pre-trained ViTs before and after applying our DVT. For all the models in the main experiments, we use 10k denoised samples randomly selected from the VOC2012 and the VOC2007 datasets, excluding their validation samples, to train generalizable denoisers.

Semantic segmentation. We follow [6, 23] to evaluate our approach in two semantic segmentation datasets: VOC2012 [10] and ADE20k [41], using a linear probing protocol, *i.e.*, a linear layer is trained to predict pixels' class from patch tokens. Tab. 2 presents the main results. We observe significant and consistent enhancements in all pre-trained ViTs across datasets. Notably, the DINOv2-*giant*, with an 83.0 mIoU on VOC2012 as reported in [23], is outperformed by our DVT-denoised DINOv2-*base* model (84.84 mIoU). This improvement is also evident in the ADE20k dataset, where the DINOv2-*giant* and DINOv2-*large* models attain mIoUs of 49.0 and 47.7, respectively, as reported in [23], while our denoised *base* model achieves a 48.66 mIoU. Remarkably, the *giant* model, which is $\mathbf{13} \times$ larger than the *base* model, is outperformed by or on par with

	VOC2012 Segmentation ADE20k S		Segmentation NYUv		2 Depth Estimation	
Method	mIoU(†)	$mAcc(\uparrow)$	mIoU(†)	$mAcc(\uparrow)$	$\delta_1(\uparrow)$	abs rel(↓)
(a1) DINO [3]	63.00	76.35	31.03	40.33	73.19	0.1701
(a2) DINO [3] + DVT	66.22	78.14	32.40	42.01	73.53	0.1731
(b1) DeiT-III [32]	70.62	81.23	32.73	42.81	72.16	0.1788
(b2) DeiT-III [32] + DVT	73.36	83.74	36.57	49.01	71.36	0.1802
(c1) EVA02 [11]	71.52	82.95	37.45	49.74	63.68	0.1989
(c2) EVA02 [11] + DVT	73.15	83.55	37.87	49.81	68.52	0.1964
(d1) CLIP [24]	77.78	86.57	40.51	52.47	73.95	0.1679
(d2) CLIP [24] + DVT	79.01	87.48	41.10	53.07	74.61	0.1667
(e1) DINOv2-reg [6]	83.64	90.67	48.22	60.52	87.88	0.1190
(e2) DINOv2-reg [6] + DVT	84.50	91.45	49.34	61.70	88.26	0.1157
(f1) DINOv2 [23]	83.60	90.82	47.29	59.18	86.88	0.1238
(f2) DINOv2 [23] + DVT	84.84	91.70	48.66	60.24	87.58	0.1200

Table 2: Qualitative performance of DVT. DVT improves differently pre-trained ViTs for dense prediction tasks. We report performance on semantic segmentation (VOC2012, ADE20K) and depth prediction (NYUd) tasks.

our denoised *base* model. This indicates that the performance gains primarily stem from effective artifact removal rather than the *minor* increase in model parameters of our denoiser network.

Our DVT also increases the performance of the concurrent DINOv2-reg model [6], where a ViT is trained with dummy learnable register tokens. As evidenced in Tab. 2, our DVT enhances the performance of both DINOv2 ((f1) vs. (f2)) and DINOv2-reg ((e1) vs. (e2)). When applying DVT only, DINOv2 shows more improvements compared to using registers ((f2) vs. (e1)); for instance, DINOv2 denoised by DVT achieves 84.84 mIoU in VOC2012 and 48.66 mIoU in ADE20k, surpassing the performance of DINOv2-reg, which achieves 83.64 mIoU and 48.22 mIoU on the respective benchmarks. Furthermore, DVT can further enhance the performance of DINOv2-reg ((e1) vs. (e2)) on both datasets (+0.86 in VOC2012 and +1.12 in ADE20k). In addition, DINOv2-reg [6] requires retraining entire models from scratch using 142M images, while our approach requires training a single Transformer block using 10k denoised samples.

Depth estimation. Following [23], we evaluate our method on the NYUv2-Depth dataset [22] using a linear evaluation protocol (more details in appendix). As shown in Tab. 2, our method clearly enhances the performance of most pretrained ViTs. For context, the DINOv2-*large* model exhibits a 0.01 RMSE improvement over the DINOv2-*base* model with $3.5 \times$ more parameters. Our denoiser achieves similar performance gains with $0.08 \times$ the parameters of the base model. These results highlight our method's efficiency, achieving marked performance gains with minimal increases in parameter count.

Object detection. In this experiment, we train ViTDet detectors [19] on the frozen features following the Faster RCNN framework [28] (more details in appendix). We train all models on the VOC trainval07+12 subset and report their mAP metrics on the test2007 subset. Results are reported in Tab. 3. Our approach

Table 3: **Object detection with frozen features.** We report the mAP metric on the VOC object detection benchmark.



Fig. 6: **Emerged object discovery ability.** The features denoised by our DVT show higher feature norms on objects of interest.

Table 4: **Unsupervised object discovery using LOST [30].** We report the corloc score across three datasets. Our DVT significantly improves existing models. [†]: results quoted from [6]; these models are ViT-*large* trained on the ImageNet-22k dataset while our reported results are based on the publicly available ViT-*base*.

Method	VOC2007	VOC2012	COCO20k
(a) [†] DINOv2 [23]	35.3	40.2	26.9
(b) [†] DINOv2-reg [6]	55.4	60.0	42.0
(c) DINOv2-reg	38.0	41.5	26.9
(d) DINOv2-reg + DVT	56.1 (+18.1)	59.3 (+17.8)	45.5 (+18.6)
(e) DINOv2	30.8	35.9	23.4
(f) $DINOv2 + DVT$	58.0 (+27.2)	60.3 (+24.4)	46.7 (+23.3)

shows consistent improvements over the studied ViTs. Notably, DINOv2-reg [6] shows a slight decrease in object detection performance when compared to the original DINOv2 [23], while our approach improves it.

Object discovery. Unsupervised object discovery has been a long-standing problem of interest. An intriguing finding from our experiments is the emerging capability of object discovery in denoised ViTs. Fig. 6 illustrates this through PCA visualizations and L_2 norms of the feature maps. Post-denoising, not only are the artifacts removed, but also the objects of interest become more distinctly visible from the feature norm values. This enhancement in object clarity is *not* a goal of DVT but emerges as the outcome of our method.

To quantitatively assess these enhancements, we follow [6] to use LOST [30] for evaluating object discovery efficacy before and after applying our DVT. We use feature norms as an indicator of object prominence. We conduct object discovery experiments on PASCAL VOC 2007 [9] and 2012 [10] and COCO20k datasets [20]. Tab. 4 presents the results. Our DVT significantly improves both DINOv2 [23] and DINOv2-reg [6] in all the evaluated datasets. In particular, while the publicly available DINOv2-reg shows some improvements ((c) vs. (e)), we find that it falls short of the performance levels reported in [6] ((c) vs. (b)).

Despite this, our DVT achieves more substantial enhancements in object discovery capabilities, even compared to the numbers reported in [6] ((f) vs. (b)).

Table 5: Ablation study on perimage denoising using KNN segmentation protocol on VOC12val.

Table 6: Ablation study on the architectural design of generalizable denoiser. We report the mIoU of the VOC2012 validation set.

Representations	mIoU	Denoiser architectures	mIoU
(a) DINOv2	65.35	(a) DINOv2 (reproduced)	83.60
(b) <i>F</i>	67.81	(b) conv1x1	82.15
(c) $\mathcal{F} + \mathcal{G}$	70.82	(c) conv3x3	83.27
(d) $\mathcal{F} + \mathcal{G} + \hat{\Delta}$	70.94	(d) Single Transformer Block + PE.	84.84
		(e) Single Transformer Block	84.81

Ablation Study 5.3

In this section, we provide ablation studies to understand the importance of different components in our proposed DVT.

Factorization. We ablate our per-image denoising method using a K-Nearest-Neighbor (KNN) pixel segmentation evaluation protocol on the VOC2012 dataset. Specifically, we collect class centroids from each training image by masked pooling to construct a memory bank using ground truth annotations. Then, for each pixel in a validation image, we classify it based on its 20 nearest neighbors in the memory bank. We report the mIoU on the validation set. Tab. 5 shows the results. We observe that combining the artifact field \mathcal{G} and the residual term Δ yields the best result (d). Omitting both these elements reduces our approach to merely utilizing a neural field \mathcal{F} to learn multi-crop ensembled image features, without addressing artifacts (b). While this variant shows improvement, it falls behind our proposed method by a large margin, underscoring the importance of removing artifacts.

Generalizable denoiser. We explore alternative architectural designs for our generalizable denoiser in Tab. 6. We study four variations: 1) our default setting, which incorporates a single Transformer Block with new learnable position embeddings; 2) our default setting but without position embeddings; 3) a multilayer convolution denoiser with a Conv1x1-ReLu-Conv1x1-ReLu-Conv1x1 structure, and 4) a multilayer convolution denoiser with a Conv3x3-ReLu-Conv3x3-ReLu-Conv3x3 structure. We observe that denoisers based on convolutional structures (b, c) do not yield good results, with the conv1x1 setting performing the worst (c). Moreover, we note that our default setting with a Transformer block and learnable positional embeddings achieves the best result (d), and removing the learnable position embeddings obtains very similar numerical performance (e). We empirically find that the design of (d) leads to better qualitative visualizations, and thus we use this setting.

Scaling behaviors. We study how DVT scales with model sizes and data scales in Fig. 7. In (a), we see DVT boosts differently-sized ViTs, even allowing ViT-base to match or exceed ViT-giant performance in semantic segmentation. Overall, DVT's scaling behaviors closely align with those of baseline models. In (b), we



Fig. 7: **DVT's Scaling Behaviors.** We study the generalizable denoiser's performance for (a) different model sizes, (b) the number of denoised samples used for training denoisers, and (c) the number of views used when performing per-image denoising.

study the impact of the number of denoised training samples on task performance, where DVT shows promising results even with limited training samples $(e.g., 100\sim1000)$. Note that our denoiser never sees ADE20k and NYU-depth datasets during training, yet generalizes effectively. In (c), we plot the task performance vs. the number of views used for the denoising. DVT benefits from more views in first-stage denoising. When training neural fields, more views enhance performance, while fewer views lead to overfitting. In particular, aggregating views is itself an approach to denoising, which still aligns with our motivation. We also demonstrate that a denoiser trained on samples denoised solely by aggregating views via neural fields (\mathcal{F} -only in (c)) surpasses baselines but underperforms the full DVT, which further confirms the effectiveness of our proposed denoising procedure.

6 Discussion and Future Works

Denoising Vision Transformers (DVT) introduces a robust method leveraging neural fields to eliminate feature artifacts from ViTs. This work additionally pinpoint positional embeddings as the primary source of these artifacts, despite their importance in various vision tasks. Using a neural field optimization process, DVT efficiently extracts clean features from the noise-riddled feature maps of existing ViTs. And using a scalable feature denoiser model, DVT eliminates the need for individual image optimizations. When learned from individually denoised samples, our denoiser generalizes well to unseen data and improves pre-trained ViTs by large margins in dense vision tasks. More broadly, our research suggests several avenues for future exploration: (1) understanding the role of positional embeddings in ViT could inform the design of next-generation deep learning architectures, and (2) redefining positional embeddings within ViTs and transformers is also an imperative problem. Lastly, combining the insights from our work and those of [6] could lead to a more complete picture of how these artifacts emerge. We hope that the results presented in this work contribute to a deeper understanding of artifacts in vision transformers and beyond.

Acknowledgements

We are grateful to many friends, including Jiageng Mao, Junjie Ye, Justin Lovelace, Varsha Kishore, and Christian Belardi, for their fruitful discussions on this work and follow-ups. Katie Luo is supported by an Nvidia Graduate Fellowship. Leonidas Guibas acknowledges the support from a Vannevar Bush Faculty Fellowship. We also acknowledge an unrestricted gift from Google in support of this project.

References

- Brown, T.B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D.M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., Amodei, D.: Language models are few-shot learners (2020)
- Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., Zagoruyko, S.: Endto-End Object Detection with Transformers, p. 213–229. Springer International Publishing (2020). https://doi.org/10.1007/978-3-030-58452-8_13, http://dx.doi.org/10. 1007/978-3-030-58452-8_13
- Caron, M., Touvron, H., Misra, I., Jégou, H., Mairal, J., Bojanowski, P., Joulin, A.: Emerging properties in self-supervised vision transformers. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 9650–9660 (2021)
- 4. Chen, X., Hsieh, C.J., Gong, B.: When vision transformers outperform resnets without pre-training or strong data augmentations. arXiv preprint arXiv:2106.01548 (2021)
- 5. Chowdhery, A., Narang, S., Devlin, J., Bosma, M., Mishra, G., Roberts, A., Barham, P., Chung, H.W., Sutton, C., Gehrmann, S., Schuh, P., Shi, K., Tsvyashchenko, S., Maynez, J., Rao, A., Barnes, P., Tay, Y., Shazeer, N., Prabhakaran, V., Reif, E., Du, N., Hutchinson, B., Pope, R., Bradbury, J., Austin, J., Isard, M., Gur-Ari, G., Yin, P., Duke, T., Levskaya, A., Ghemawat, S., Dev, S., Michalewski, H., Garcia, X., Misra, V., Robinson, K., Fedus, L., Zhou, D., Ippolito, D., Luan, D., Lim, H., Zoph, B., Spiridonov, A., Sepassi, R., Dohan, D., Agrawal, S., Omernick, M., Dai, A.M., Pillai, T.S., Pellat, M., Lewkowycz, A., Moreira, E., Child, R., Polozov, O., Lee, K., Zhou, Z., Wang, X., Saeta, B., Diaz, M., Firat, O., Catasta, M., Wei, J., Meier-Hellstern, K., Eck, D., Dean, J., Petrov, S., Fiedel, N.: Palm: Scaling language modeling with pathways (2022)
- Darcet, T., Oquab, M., Mairal, J., Bojanowski, P.: Vision transformers need registers (2023)
- 7. Devlin, J., Chang, M.W., Lee, K., Toutanova, K.: Bert: Pre-training of deep bidirectional transformers for language understanding (2019)
- Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., Houlsby, N.: An image is worth 16x16 words: Transformers for image recognition at scale (2021)
- Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results. http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html

- 16 J. Yang, K. Luo et al.
- Everingham, M., Van Gool, L., Williams, C.K.I., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes Challenge 2012 (VOC2012) Results. http://www.pascal-network.org/challenges/VOC/voc2012/workshop/index.html
- 11. Fang, Y., Sun, Q., Wang, X., Huang, T., Wang, X., Cao, Y.: Eva-02: A visual representation for neon genesis. arXiv preprint arXiv:2303.11331 (2023)
- Fang, Y., Wang, W., Xie, B., Sun, Q., Wu, L., Wang, X., Huang, T., Wang, X., Cao, Y.: Eva: Exploring the limits of masked visual representation learning at scale (2022)
- 13. Goyal, S., Ji, Z., Rawat, A.S., Menon, A.K., Kumar, S., Nagarajan, V.: Think before you speak: Training language models with pause tokens (2023)
- He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. pp. 16000–16009 (2022)
- Kerr, J., Kim, C.M., Goldberg, K., Kanazawa, A., Tancik, M.: Lerf: Language embedded radiance fields (2023)
- Kirillov, A., Mintun, E., Ravi, N., Mao, H., Rolland, C., Gustafson, L., Xiao, T., Whitehead, S., Berg, A.C., Lo, W.Y., Dollár, P., Girshick, R.: Segment anything (2023)
- 17. Kobayashi, S., Matsumoto, E., Sitzmann, V.: Decomposing nerf for editing via feature field distillation (2022)
- Li, N., Liu, S., Liu, Y., Zhao, S., Liu, M., Zhou, M.: Close to human quality tts with transformer. arXiv preprint arXiv:1809.08895 (2018)
- Li, Y., Mao, H., Girshick, R., He, K.: Exploring plain vision transformer backbones for object detection. In: European Conference on Computer Vision. pp. 280–296. Springer (2022)
- Lin, T.Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., Zitnick, C.L.: Microsoft coco: Common objects in context. In: Computer Vision– ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13. pp. 740–755. Springer (2014)
- Müller, T., Evans, A., Schied, C., Keller, A.: Instant neural graphics primitives with a multiresolution hash encoding. ACM Transactions on Graphics (ToG) 41(4), 1– 15 (2022)
- 22. Nathan Silberman, Derek Hoiem, P.K., Fergus, R.: Indoor segmentation and support inference from rgbd images. In: ECCV (2012)
- Oquab, M., Darcet, T., Moutakanni, T., Vo, H., Szafraniec, M., Khalidov, V., Fernandez, P., Haziza, D., Massa, F., El-Nouby, A., et al.: Dinov2: Learning robust visual features without supervision. arXiv preprint arXiv:2304.07193 (2023)
- Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision (2021)
- 25. Radford, A., Narasimhan, K., Salimans, T., Sutskever, I., et al.: Improving language understanding by generative pre-training (2018)
- Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., Sutskever, I., et al.: Language models are unsupervised multitask learners. OpenAI blog 1(8), 9 (2019)
- Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., Liu, P.J.: Exploring the limits of transfer learning with a unified text-to-text transformer (2023)
- Ren, S., He, K., Girshick, R., Sun, J.: Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems 28 (2015)

- Shen, W., Yang, G., Yu, A., Wong, J., Kaelbling, L.P., Isola, P.: Distilled feature fields enable few-shot language-guided manipulation. arXiv preprint arXiv:2308.07931 (2023)
- Siméoni, O., Puy, G., Vo, H.V., Roburin, S., Gidaris, S., Bursuc, A., Pérez, P., Marlet, R., Ponce, J.: Localizing objects with self-supervised transformers and no labels. arXiv preprint arXiv:2109.14279 (2021)
- Tancik, M., Srinivasan, P.P., Mildenhall, B., Fridovich-Keil, S., Raghavan, N., Singhal, U., Ramamoorthi, R., Barron, J.T., Ng, R.: Fourier features let networks learn high frequency functions in low dimensional domains. NeurIPS (2020)
- 32. Touvron, H., Cord, M., Jégou, H.: Deit iii: Revenge of the vit (2022)
- 33. Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., Lample, G.: Llama: Open and efficient foundation language models (2023)
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems **30** (2017)
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, L., Polosukhin, I.: Attention is all you need (2023)
- Voita, E., Ferrando, J., Nalmpantis, C.: Neurons in large language models: Dead, n-gram, positional. arXiv preprint arXiv:2309.04827 (2023)
- Wang, Y., Skerry-Ryan, R., Stanton, D., Wu, Y., Weiss, R.J., Jaitly, N., Yang, Z., Xiao, Y., Chen, Z., Bengio, S., Le, Q., Agiomyrgiannakis, Y., Clark, R., Saurous, R.A.: Tacotron: Towards end-to-end speech synthesis. In: Interspeech 2017. ISCA (2017). https://doi.org/10.21437/interspeech.2017-1452, http://dx.doi.org/10.21437/ Interspeech.2017-1452
- Wightman, R.: Pytorch image models. https://github.com/rwightman/pytorch-imagemodels (2019). https://doi.org/10.5281/zenodo.4414861
- Xiao, G., Tian, Y., Chen, B., Han, S., Lewis, M.: Efficient streaming language models with attention sinks (2023)
- 40. Yang, J., Ivanovic, B., Litany, O., Weng, X., Kim, S.W., Li, B., Che, T., Xu, D., Fidler, S., Pavone, M., et al.: Emernerf: Emergent spatial-temporal scene decomposition via self-supervision. In: The Twelfth International Conference on Learning Representations (2024)
- 41. Zhou, B., Zhao, H., Puig, X., Xiao, T., Fidler, S., Barriuso, A., Torralba, A.: Semantic understanding of scenes through the ade20k dataset (2018)
- Zhou, J., Wei, C., Wang, H., Shen, W., Xie, C., Yuille, A., Kong, T.: ibot: Image bert pre-training with online tokenizer. arXiv preprint arXiv:2111.07832 (2021)