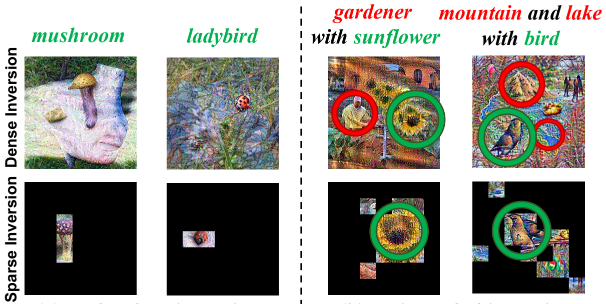
Sparse Model Inversion: Efficient Inversion of Vision Transformers for Data-Free Applications

Zixuan Hu[1 2] Yongxian Wei[1] Li Shen[3 4] Zhenyi Wang[5] Lei Li[1] Chun Yuan[1] Dacheng Tao[2]

Abstract

Model inversion, which aims to reconstruct the original training data from pre-trained discriminative models, is especially useful when the original training data is unavailable due to privacy, usage rights, or size constraints. However, existing dense inversion methods attempt to reconstruct the entire image area, making them extremely inefficient when inverting high-resolution images from large-scale Vision Transformers (ViTs). We further identify two underlying causes of this inefficiency: the redundant inversion of noisy backgrounds and the unintended inversion of spurious correlations—a phenomenon we term "hallucination" in model inversion. To address these limitations, we propose a novel sparse model inversion strategy, as a plug-and-play extension to speed up existing dense inversion methods with no need for modifying their original loss functions. Specifically, we selectively invert semantic foregrounds while stopping the inversion of noisy backgrounds and potential spurious correlations. Through both theoretical and empirical studies, we validate the efficacy of our approach in achieving significant inversion acceleration (up to ×3.79) while maintaining comparable or even enhanced downstream performance in data-free model quantization and data-free knowledge transfer. Code is available at https://github.com/Egg-Hu/SMI.



(a) Redundant inversion of noisy backgrounds

(b) Unintended inversion of spurious correlations

Figure 1. The inefficiency of dense inversion (*e.g.*, DeepInversion (Yin et al., 2020)) arises from (a) redundant inversion of noisy backgrounds, and (b) unintended inversion of spurious correlations between foregrounds (green) and backgrounds (red), which are improperly memorized in pre-trained models.

1. Introduction

Given a discriminative model $f : \boldsymbol{x} \to y$, model inversion aims to reconstruct inputs $\boldsymbol{x}$ from a target output y. This technique can be utilized to synthesize surrogate data when the original dataset is unavailable due to constraints like privacy concerns, usage rights, or dataset size. An illustrative application is data-free model quantization, which enables the quantization of a full-precision model to a low-precision one for lightweight deployment by using surrogate data inverted from the full-precision model (Li et al., 2023d; 2022b; Xu et al., 2020; Qin et al., 2023). Another application is data-free knowledge transfer, which enables knowledge transfer from a teacher model to a student model by using surrogate data inverted from the teacher model (Yin et al., 2020; Fang et al., 2021; Chundawat et al., 2023; Zhu et al., 2021). Overall, model inversion provides a practical solution in data-constrained scenarios by synthesizing surrogate data directly from the model itself.

However, existing inversion methods (Zhu et al., 2021; Fang et al., 2022; Zhang et al., 2022c; Yu et al., 2023; Braun et al., 2023; Patel et al., 2023) share a "dense" characteristic, meaning they attempt to reconstruct the entire image area. This becomes extremely inefficient when inverting high-resolution images from large-scale ViTs (see Tab. 1). As shown in Fig. 1, we further reveal two underlying causes,

[1]Tsinghua Shenzhen International Graduate School, Tsinghua University, Shenzhen, China [2]College of Computing & Data Science, Nanyang Technological University, Singapore [3]School of Cyber Science and Technology, Sun Yat-sen University, Shenzhen, China [4]JD Explore Academy, China [5]University of Maryland, College Park, USA. Correspondence to: Li Shen <mathshenli@gmail.com>, Chun Yuan <yuanc@sz.tsinghua.edu.cn>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

including the redundant inversion of noisy backgrounds and the unintended inversion of spurious correlations—a phenomenon we term "hallucination" in model inversion.

Based on our observations, we propose a novel strategy called sparse model inversion, as a plug-and-play extension to speed up existing dense inversion methods with no need for modifying their original loss functions. Our sparse inversion strategy enables efficient inversion from large-scale ViTs with less inversion of noisy backgrounds and potential spurious correlations. Specifically, we selectively invert semantic foregrounds while stopping the inversion of uninformative backgrounds. This is achieved by two components: *semantic patch identification*, utilizing attention weights from the preceding iteration to determine which patches to invert in the current iteration, and *early inversion stopping*, stopping the inversion of uninformative background patches in the early iterations. We implement "stopping" by discarding these background patches, no longer processing them forward or computing their backward gradients, thus excluding them from inversion. This stopping can be done progressively as the inversion process progresses, ensuring only the most informative foreground patches are retained.

To validate the efficacy of our approach, we perform a combination of theoretical and empirical studies. Empirically, we verify that our approach can achieve significant inversion acceleration up to 3.79×, while maintaining comparable or improved downstream performance in data-free model quantization and data-free knowledge transfer. In Sec. 4.4, we theoretically analyze that utilizing sparsely inverted data can effectively reduce the required number of training samples and iterations when training ViTs for downstream classification tasks (Li et al., 2023b), thereby stabilizing and accelerating convergence. In summary, our main contributions are outlined as follows:

- We reveal the limitations and underlying causes of existing dense inversion methods, *i.e.*, inefficiency of inverting high-resolution images from large-scale ViTs.
- We propose the sparse inversion strategy, as a plug-and-play extension of existing dense inversion, to achieve efficient inversion of ViTs with less inversion of noisy backgrounds and potential spurious correlations.
- We empirically and theoretically verify the efficacy of our sparse inversion strategy in achieving significant inversion acceleration while maintaining comparable or even enhanced downstream performance in data-free model quantization and data-free knowledge transfer.

2. Related Work

Model inversion aims to reconstruct the inputs given the outputs of a discriminative model. Research on model inversion is initially in the security community. Fredrikson et al. (2015) introduce model inversion attack to reconstruct private inputs. Subsequent works broaden this approach to new attack scenarios (He et al., 2019; Yang et al., 2019). More recently, model inversion has been used in data-inaccessible scenarios for tasks like data-free knowledge transfer (Lopes et al., 2017; Zhu et al., 2021; Fang et al., 2022; Zhang et al., 2022c; Yu et al., 2023; Braun et al., 2023; Patel et al., 2023; Shao et al., 2023) and data-free model quantization (Choi et al., 2021; Xu et al., 2020; Li et al., 2023d; Hu et al., 2023c). More applications of model inversion are introduced in App. C. However, previous inversion methods suffer from extreme inefficiency when inverting high-resolution images from large-scale ViTs. They typically employ model inversion as a tool to synthesize surrogate data, while our work is the first to enhance the scalability of model inversion for inverting high-resolution images from large-scale ViTs.

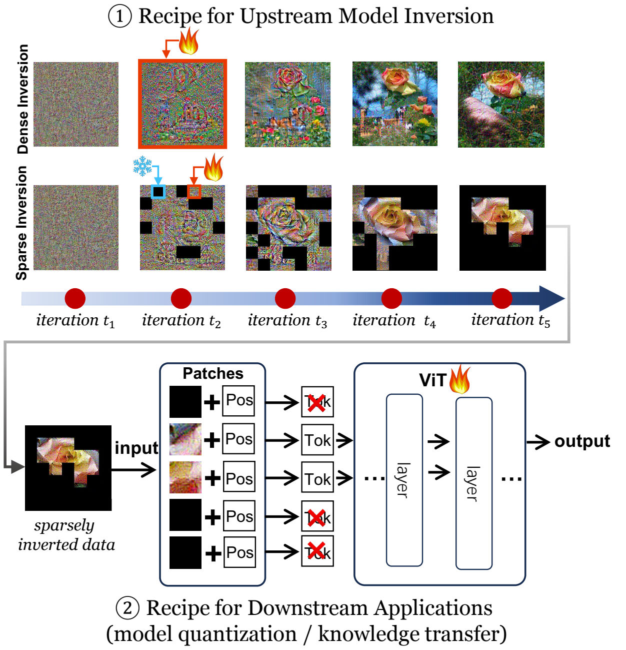


Figure 2. Recipe for model inversion and applications. Our approach selectively inverts semantic foreground patches while progressively stopping the inversion of uninformative background ones. When utilizing sparsely inverted data for downstream applications, we feed forward only the retained foreground patches.

Token sparsification. Recent advancements in token sparsification methods have proven effective in boosting the inference speed of ViTs, as seen in works of Wang et al. (2021); Rao et al. (2021); Meng et al. (2022); Xu et al. (2022); Liang et al. (2022); Bolya et al. (2023); Chang et al. (2023); Kim et al. (2024); Haurum et al. (2023); Chen et al. (2023a). These methods point out that uninformative patches occupy a significant portion of processing bandwidth but have minimal impact on the final prediction. However, the potential benefits of incorporating sparsity into the inversion process of ViTs still remain unexplored.

Spurious correlation refers to the statistical connection between foregrounds and non-predictive backgrounds, which is not necessarily causal (Bica et al., 2021; Hu et al., 2022; Ye et al., 2023; Kim et al., 2023; Ghosal & Li, 2023; Liu et al., 2022a). For example, the waterbird may spuriously correlate to the ocean background. This may cause a model to base its predictions on the background non rather than on the true relevant foreground, damaging its generalization during deployment when such correlation no longer holds. However, the potential risk that model inversion could unintentionally invert these spurious correlations from the pre-trained model is still unexplored. Our research is the first to identify and analyze this phenomenon in the context of model inversion.

3. Rethinking Dense Model Inversion

3.1. Problem Setup

Case study of dense inversion: DeepInversion (Yin et al., 2020). Given a classification model f_u, a randomly initialized input $\boldsymbol{X}^\mathrm{I} \in \mathbb{R}^{H\times W\times C}$ (height, width, and number of channels) and a target label y, the inversion process is optimizing a classification loss with a regularization term:

$$\min_{\boldsymbol{X}^\mathrm{I}} \mathcal{L}_\mathrm{inv} = \mathcal{L}_\mathrm{cls}\left(f_\mathrm{u}(\boldsymbol{X}^\mathrm{I}), y\right) + \alpha_\mathcal{R}\mathcal{R}(\boldsymbol{X}^\mathrm{I}), \tag{1}$$

where $\mathcal{L}_\mathrm{cls}(\cdot)$ is a classification loss (*e.g.*, cross-entropy loss) to ensure the label-conditional inversion, which desires $\boldsymbol{X}^\mathrm{I}$ could be predicted as y and exhibit discriminative features of y. $\mathcal{R}(\cdot)$ is an image regularization term widely used to penalize the total variance for local consistency (Braun et al., 2023; Hatamizadeh et al., 2022), with $\alpha_\mathcal{R}$ as the coefficient.

$$\mathcal{R}(\boldsymbol{X}^\mathrm{I}) = \sum_{i=2}^{H}\sum_{j=2}^{W}\left(\left\|\boldsymbol{X}^\mathrm{I}_{i,j} - \boldsymbol{X}^\mathrm{I}_{i-1,j}\right\|_2 + \left\|\boldsymbol{X}^\mathrm{I}_{i,j} - \boldsymbol{X}^\mathrm{I}_{i,j-1}\right\|_2 + \left\|\boldsymbol{X}^\mathrm{I}_{i,j} - \boldsymbol{X}^\mathrm{I}_{i-1,j-1}\right\|_2\right) + \sum_{i=2}^{H}\sum_{j=1}^{W-1}\left\|\boldsymbol{X}^\mathrm{I}_{i,j} - \boldsymbol{X}^\mathrm{I}_{i-1,j+1}\right\|_2, \tag{2}$$

where $\boldsymbol{X}^\mathrm{I}_{i,j}$ refers to a 3-dimensional vector containing the values of the pixel at position (i, j) across all channels. We omit the feature distribution regularization term due to the absence of batch normalization in ViTs. The main differences among various dense inversion methods (Yin et al., 2020; Li et al., 2023d; Hatamizadeh et al., 2022) mainly lie in the design of the regularization terms, which can be added to Eq. (1) compatibly.

3.2. Limitation & Cause

Limitation: Inefficiency of inverting high-resolution images from large-scale ViTs. Existing dense inversion methods (Lopes et al., 2017; Zhu et al., 2021; Fang et al., 2022; Zhang et al., 2022c; Yu et al., 2023; Braun et al., 2023; Patel et al., 2023; Shao et al., 2023) are mainly designed for small-scale convolutional networks. As indicated in Tab. 1, when inverting high-resolution images from large-scale ViTs, there is a notable increase in time and computational expenses. This is because inverting $\boldsymbol{X}^I$ in Eq. (1) requires multiple iterations of forward and backward propagation. As the image resolution or the model size grows, the number of learnable parameters in $\boldsymbol{X}^I$ rises, and the costs associated with forward and backward propagation through the large model also increase significantly.

Table 1. Inefficiency of dense inversion (*e.g.*, DeepInversion).

Resolution	**Model**	**Inversion**	**Throughput** (its/s)	**FLOPs** (G)
32 × 32	ResNet18	Dense	77.91	0.11
224 × 224	ResNet18	Dense	10.21	5.47
224 × 224	DeiT/16-Base	Dense	1.79	6475.63

Cause 1: Redundant inversion of noisy backgrounds. In Eq. (1), when targeting a specific label y, we aim to craft semantic features in $\boldsymbol{X}^\mathrm{I}$ by reducing the classification loss. However, from Fig. 1(a), we observe that these semantic features typically occupy only a small portion in $\boldsymbol{X}^\mathrm{I}$, while the backgrounds tend to be noisy. The reason is the backgrounds tend to contribute minimally to the decrease of $\mathcal{L}_\mathrm{cls}$ (see Tab. 2), thus maintaining characteristics similar to the initialized noise. Despite their uselessness, the uninformative and noisy backgrounds are equally included in the inversion process, resulting in wasted computational resources and time costs, thereby damaging the overall efficiency of inversion. Similarly, studies of token sparsification (Wang et al., 2021; Rao et al., 2021; Haurum et al., 2023; Chang et al., 2023; Kim et al., 2024; Chen et al., 2023a) suggest that background patches consume most of the processing bandwidth but contribute little to the final prediction.

Table 2. Backgrounds contribute minimally to reducing $\mathcal{L}_\mathrm{cls}$ during inversion. The initial loss value is evaluated on all patches.

Ablation	**Change of** $\mathcal{L}_\mathrm{cls}$
Identified Background Patches	10.78 → 10.69
Identified Foreground Patches	10.78 → 0.12

Cause 2: Unintended inversion of spurious correlations—a phenomenon we term "hallucination" in model inversion. As shown in Fig. 1(b), in addition to redundant inversion of noisy backgrounds, spurious correlations between the foregrounds and backgrounds can also be unintentionally inverted. For example, in the original training dataset, the waterbird may spuriously correlate to the ocean background. This statistical correlation can be improperly memorized by the model trained on it. When we attempt to invert waterbird images from this model, the ocean background can be unintentionally inverted, leading to co-inversion of both the foregrounds and connected backgrounds. The example in Fig. 1(b) illustrates such co-inversion, such as the co-occurrences of a gardener background when the target is a sunflower, or a mountainous and lacustrine background when the target is a bird. Several studies (Bica et al., 2021;

Hu et al., 2022; Ye et al., 2023; Kim et al., 2023; Ghosal & Li, 2023) have demonstrated that spurious correlations may cause a model to rely on the background rather than the true relevant foreground for predictions, thereby impairing its generalization during deployment when such correlations no longer hold. Moreover, when inverted data containing spurious correlations are utilized for knowledge transfer, these misleading correlations can be transferred to from the teacher model to the student model (Ojha et al., 2024).

4. Methodology

4.1. Preliminary of ViTs

ViTs (Dosovitskiy et al., 2020; Vaswani et al., 2017; Liu et al., 2024) first partition the input image $\boldsymbol{X}^{\mathrm{I}} \in \mathbb{R}^{H\times W\times C}$ into L non-overlapping patches, which are subsequently embedded into tokens of dimension D, *i.e.*, $\boldsymbol{X}^{\mathrm{I}} = [\boldsymbol{x}_{\texttt{[CLS]}}, \boldsymbol{x}_1, ..., \boldsymbol{x}_L]$ and $\boldsymbol{x}_i \in \mathbb{R}^D$. $\boldsymbol{x}_{\texttt{[CLS]}}$ is the class token inserted to the front before all image tokens to facilitate final classification. To integrate positional relationships, learnable position encodings are added to all tokens. The processed tokens are then fed into several stacked ViT layers. Each layer includes a multi-head self-attention (MHSA) layer and a feed-forward network (FFN). In MHSA, $\boldsymbol{X}^{\mathrm{I}}$ is projected to three matrices, namely query $\boldsymbol{Q}$, key $\boldsymbol{K}$, and value $\boldsymbol{V}$ matrices. The attention operation is defined as

$$\mathrm{Attention}(\boldsymbol{Q},\boldsymbol{K},\boldsymbol{V}) = \mathrm{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}\right)\boldsymbol{V}, \quad (3)$$

where d is the length of the query vectors in $\boldsymbol{Q}$. We define the square matrix $\boldsymbol{A} \triangleq \mathrm{Softmax}\left(\frac{\boldsymbol{Q}\boldsymbol{K}^T}{\sqrt{d}}\right)$, $\boldsymbol{A} \in \mathbb{R}^{(L+1)\times(L+1)}$, which is known as the attention map, representing attention weights of all token pairs. Further more, we define $\boldsymbol{a}_i \triangleq \boldsymbol{A}_{[i,:]}$, $\boldsymbol{a}_i$ indicating the attention weights from $\boldsymbol{x}_i$ to all tokens $[\boldsymbol{x}_{\texttt{[CLS]}}, \boldsymbol{x}_1, ..., \boldsymbol{x}_L]$. Particularly, $\boldsymbol{a}_{\texttt{[CLS]}}$ refers to $\boldsymbol{a}_0$. Based on Eq. (3), the i^{th} output token can be viewed as a linear combination of all tokens' value vectors $[\boldsymbol{v}_{\texttt{[CLS]}}, \boldsymbol{v}_1, ..., \boldsymbol{v}_L] = \boldsymbol{V}$, weighted by $\boldsymbol{a}_i$. Then, these output tokens are sent to FFN, consisting of two fully connected layers with an activation layer. At the final ViT layer, the output token $\boldsymbol{x}_{\texttt{[CLS]}}$, summarizing the entire image, is extracted as input to the classifier, generating the image's classification probability distribution.

4.2. Sparse Model Inversion (SMI)

When applying dense inversion methods to ViTs, all patches will undergo inversion. In contrast, we propose a sparse model inversion, which involves two key components: *semantic patch identification*, a method for identifying semantic patches to invert in subsequent iterations, and *early inversion stopping*, a technique to stop the inversion of uninformative background patches. The overall inversion process is illustrated in Fig. 3.

Semantic patch identification. The first question is how to identify the semantic patches that are crucial for inversion. At iteration t within the inversion process, we propose to identify semantic patches utilizing the attention weights $\boldsymbol{a}_{\texttt{[CLS]}}$ (defined in Sec. 4.1) from the preceding iteration $t-1$. Here, $\boldsymbol{a}_{\texttt{[CLS]}}$ is a $(L+1)$-dimension vector, representing the attention weights from token $\boldsymbol{x}_{\texttt{[CLS]}}$ to all tokens $[\boldsymbol{x}_{\texttt{[CLS]}}, \boldsymbol{x}_1, ..., \boldsymbol{x}_L]$. The interaction between $\boldsymbol{x}_{\texttt{[CLS]}}$ and all tokens is performed via attention:

$$\boldsymbol{x}_{\texttt{[CLS]}} = \boldsymbol{a}_{\texttt{[CLS]}} \cdot \boldsymbol{V}. \quad (4)$$

The output $\boldsymbol{x}_{\texttt{[CLS]}}$ is a linear combination of all tokens' value vectors, weighted by $\boldsymbol{a}_{\texttt{[CLS]}}$. Since $\boldsymbol{x}_{\texttt{[CLS]}}$ in the final layer serves for classification, it is rational to view $\boldsymbol{a}_{\texttt{[CLS]}}$ as an indicator, measuring the extent to which each token contributes label-relevant information to final predictions.

Moreover, we only use $\boldsymbol{a}_{\texttt{[CLS]}}$ from the final ViT layer, as it more precisely reflects the relationships among tokens. This is in contrast to shallower layers, where tokens interact to develop enhanced representations. Furthermore, within each ViT layer, the MHSA comprises H heads that execute parallel operations defined in 3. Consequently, there are H distinct attention weights represented as $[\boldsymbol{a}^{(1)}_{\texttt{[CLS]}}, ..., \boldsymbol{a}^{(H)}_{\texttt{[CLS]}}]$. To obtain more comprehensive relationships among all tokens, we compute the average of $\boldsymbol{a}_{\texttt{[CLS]}}$ across all heads (Fayyaz et al., 2022), *i.e.*, $\boldsymbol{a}_{\texttt{[CLS]}} = \frac{1}{H}\sum_{h=1}^{H} \boldsymbol{a}^{(h)}_{\texttt{[CLS]}}$. Note that this process to identify semantic patches requires no additional computational or informational demands, as it is an inherent part of the original ViTs' feed-forward process.

Early inversion stopping. The second question is how to stop the inversion of other uninformative patches. Suppose we have $L^{(t-1)}$ ($L^{(t-1)} \leq L$) patches remaining at the beginning of iteration t, and other tokens (if any) have been stopped previously. We start by evaluating the importance of each remaining token based on the attention weights from the preceding iteration $t-1$. Then, we stop the inversion of additional $p\%$ patches with the lowest attention weights, so that only $L^{(t)} = L^{(t-1)} \times (1-p\%)$ patches will be retained for subsequent inversion. We implement "stopping" by directly pruning patches with the lowest attention weights, which means they will no longer involve feed-forward processing and backward gradient calculations, and thus be excluded from inversion ever since. Those tokens will not be updated via Eq. (1) anymore. Patch pruning is performed after the addition of position embeddings to maintain the relative positional relationships among patches.

Moreover, we provide a progressive stopping strategy, *i.e.*, multi-stage stopping. Specifically, in the early iterations, when images are predominantly noisy and semantic patches are less discernible, our stopping strategy is conservative, *i.e.*, stopping inversion of a limited number of patches. As

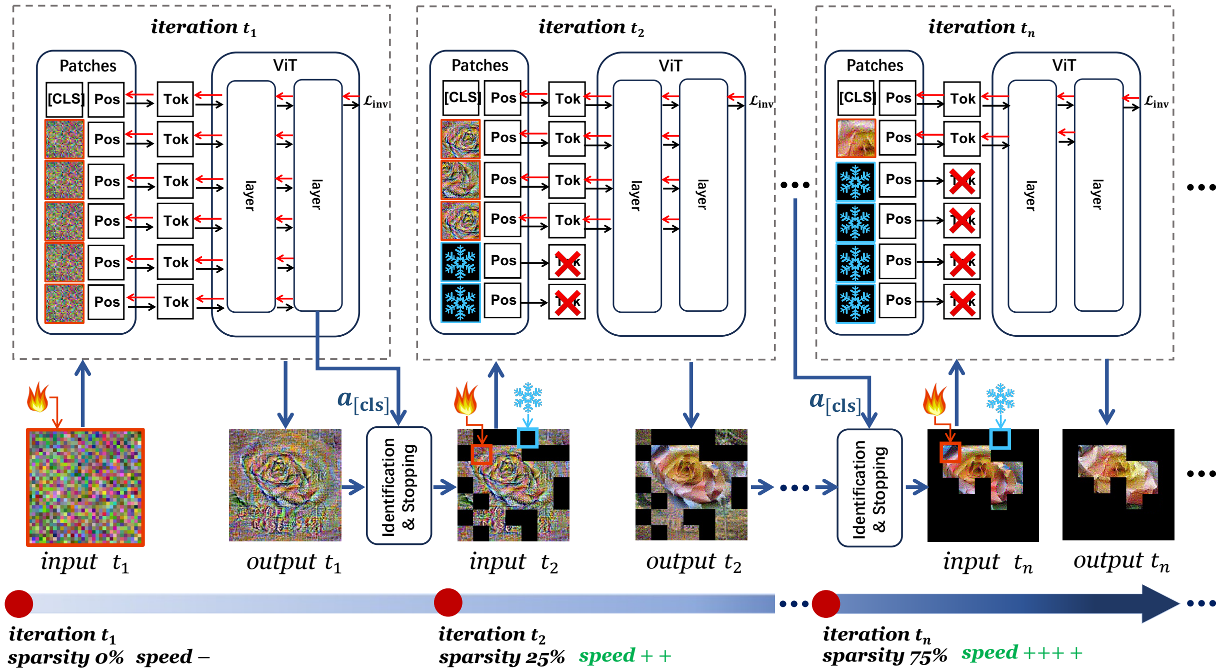


Figure 3. Overall process of sparse model inversion. As the inversion progresses, our approach selectively inverts semantic foreground patches while progressively stopping the inversion of uninformative background patches (marked as black blocks). Those stopped patches are directly discarded, with no further feed-forward processing and backward gradient computation, and thus are excluded from inversion ever since. The final inverted image only retains sparse patches with semantically meaningful information.

model inversion evolves, the clarity of inverted images increases, prompting us to stop inversion of more patches deemed to be uninformative, ensuring only the most critical patches are retained and processed further. The overall inversion process is illustrated in Fig. 3. As the inversion progresses, our approach selectively inverts semantic foreground patches while progressively stopping the inversion of other uninformative patches (marked as black blocks).

4.3. Applications of Model Inversion

Below, we introduce how to use sparsely inverted data to achieve data-free model quantization (Li et al., 2023d) and data-free knowledge transfer (Yin et al., 2020). As shown in Fig. 2, we adopt a specific recipe for using sparsely inverted data, only feeding forward the retained foreground patches while discarding other background patches (marked as black blocks). This can speed up downstream applications by reducing the number of tokens and improve performance by discarding noisy backgrounds (Li et al., 2023b) and avoiding potential spurious correlations (Ghosal & Li, 2023).

4.3.1. Data-Free Model Quantization

Data-free model quantization aims to quantize a full-precision (FP) model to a low-precision one for lightweight deployment by using surrogate data inverted from the full-precision model (Li et al., 2023d; 2022b; Xu et al., 2020; Qin et al., 2023). Following (Li et al., 2022b), the quantization is defined by the following equation:

$$\theta_{\mathrm{d}} = \left\lfloor \{\mathrm{clip}\,(\theta_{\mathrm{u}}; T_{\min}, T_{\max}) - T_{\min}\}/S \right\rceil, \qquad (5)$$
$$\text{where } S = \{T_{\max} - T_{\min}\}/\{2^k - 1\}.$$

Here, θ_{u} and θ_{d} denote the parameters of the FP model and its quantized variant, respectively. The round operator is represented by $\lfloor\cdot\rceil$. The term k refers to the bit precision for the quantized model, such as 4 or 8 bits. The scale factor S is calculated as described. Critically, $T_{\min}$ and $T_{\max}$ are the bounding values for quantization, which must be determined prior to the quantization process.

For weight quantization, $T_{\min}$ and $T_{\max}$ are directly determined by the minimum and maximum values of the FP weights. For activation quantization, following (Li et al., 2022b), we first invert surrogate data from the FP model and feed it to the FP model to obtain their activations. Then, we set $T_{\min}$ and $T_{\max}$ as the minimum and maximum values of these activations, respectively. Once $T_{\min}$ and $T_{\max}$ are set, we can perform activation quantization as Eq. (5). The rationale behind this approach is that the inverted data can provide prior information about original data distribution, helping to eliminate outliers and represent the majority of the FP activations more precisely.

4.3.2. Data-Free Knowledge Transfer

Data-free knowledge transfer enables knowledge transfer from a teacher model to a student model by using surrogate

Table 3. Model-quantization results on ImageNet. Sparsity refers to the fraction of remaining patches. Gaussian Noise refers to calibrating the quantization configuration using Gaussian noise. W4/A8 refers to the bit precision for weight and activation quantization, respectively. The changes in blue refer to the comparison with DeepInversion.

Model	Method	Model Inversion (Upstream)				Quantization (Downstream)			
		Sparsity	Throughput (its/s) ↑	FLOPs (G) ↓	GPU Mem (MB) ↓	Prec.	Top-1	Prec.	Top-1
DeiT/16 -Tiny	Original	—	—	—	—	FP	72.14	FP	72.14
	Gaussian Noise	—	—	—	—	W4/A8	7.80	W8/A8	10.55
	PSAQ-ViT (Dense)	0	0.74	414.20	1648.08	W4/A8	64.97	W8/A8	70.54
	DeepInversion (Dense)	0	7.33	414.20	1118.69	W4/A8	64.28	W8/A8	70.27
	DeepInversion (Sparse)	77%	18.82 (×2.57)	107.32(−74.09%)	476.32(−57.42%)	W4/A8	64.04	W8/A8	70.13
DeiT/16 -Base	Original	—	—	—	—	FP	81.85	FP	81.85
	Gaussian Noise	—	—	—	—	W4/A8	11.09	W8/A8	14.72
	PSAQ-ViT (Dense)	0	0.46	6475.63	9327.12	W4/A8	76.73	W8/A8	78.93
	DeepInversion (Dense)	0	1.19	6475.63	4096.96	W4/A8	75.99	W8/A8	78.58
	DeepInversion (Sparse)	77%	4.51 (×3.79)	1578.97 (−75.62%)	1516.64 (−62.98%)	W4/A8	77.51	W8/A8	79.63

data inverted from the teacher model (Yin et al., 2020; Fang et al., 2021; Chundawat et al., 2023; Zhu et al., 2021). We begin with a teacher model f_{u}, which has been trained on a specific dataset $\mathcal{D}_{\mathrm{u}}$. We invert surrogate data $\hat{\mathcal{D}}_{\mathrm{u}}$ from the teacher and utilize it to transfer the teacher's specific knowledge on $\mathcal{D}_{\mathrm{u}}$ to a vanilla student model f_{d}. The knowledge transfer is implemented by minimizing the disparity in the prediction outputs on $\hat{\mathcal{D}}_{\mathrm{u}}$ between f_{u} and f_{d}, formulated as:

$$\theta_{\mathrm{d}} = \min_{\theta_{\mathrm{d}}} \frac{1}{|\mathcal{D}_{\mathrm{u}}|} \sum_{\boldsymbol{x} \in \mathcal{D}_{\mathrm{u}}} \mathrm{KL}\left(f_{\mathrm{u}}\left(\boldsymbol{x}; \boldsymbol{\theta}_{\mathrm{u}}\right)/\tau; f_{\mathrm{d}}\left(\boldsymbol{x}; \boldsymbol{\theta}_{\mathrm{d}}\right)/\tau\right), \tag{6}$$

where KL denotes the Kullback–Leibler divergence, and τ is the temperature parameter. Data-free knowledge transfer can be used to transfer knowledge from a large-scale model to a smaller one for model compression (Fang et al., 2021), transfer selective knowledge from the original model to a new model for machine unlearning (Chundawat et al., 2023), or transfer knowledge from client models to a server model for federated learning (Zhu et al., 2021; Zhang et al., 2022a).

4.4. Analytical Study

How does sparse model inversion achieve significant acceleration? Given an image split into L patches, each with an embedding dimension of D, the computational complexity of self-attention (SA) and feed-forward network (FFN) are (Chen et al., 2023a):

$$O(\mathrm{SA}) = 3LD^2 + 2L^2D, \;\; O(\mathrm{FFN}) = 8LD^2. \tag{7}$$

Since the complexities of SA and FFN scale quadratically and linearly with L, our approach can significantly reduce the cost by decreasing the input patch number.

How does sparse model inversion benefit downstream applications? As shown in Fig. 2, it can speed up downstream applications by reducing the number of input tokens. Furthermore, for quantization, using sparsely inverted data can achieve more precise bounding values in Eq. (5) by reducing potential outliers in the noisy backgrounds. For knowledge transfer, we theoretically analyze how the noise and sparsity of inverted data affect the convergence conditions in the context of training ViTs for classification. Our analysis draws upon the framework established by Li et al. (2023b) (refer to App. D for details). To begin, we define several key factors:

(i) Patch sparsity setup: Consider N inverted samples $\{(\boldsymbol{X}^n, y^n)\}_{n=1}^N$, where each $\boldsymbol{X}^n$ comprises L' retained patches $[\boldsymbol{x}_1^n, ..., \boldsymbol{x}_{L'}^n]$, $(L' \leq L)$. Let $\mathcal{S}^n \subseteq [L']$ represent the indices of label-relevant[1] patches in $\boldsymbol{X}^n$. We define the average fraction of label-relevant patches as $\alpha = \sum_{n=1}^N \frac{|\mathcal{S}^n|}{N \cdot L'}$.

(ii) Patch noise setup: Label-relevant patches in $\boldsymbol{x}^n$ correspond to specific patterns $\boldsymbol{\mu}_{y^n}$ of its label y^n with a noise level τ, satisfying $\left\|\boldsymbol{x}^n - \boldsymbol{\mu}_{y^n}\right\|_2 \leq \tau$. Other patches in $\boldsymbol{X}^n$, however, correspond to patterns of other labels or just noise.

(iii) Convergence condition: We denote the required number of training samples and iterations as N and T, respectively.

Remark 4.1. Compared to real data, using densely inverted data makes the convergence process more challenging. This is primarily due to the inherent higher noise level τ in inverted data[2]. With the noise level τ increasing, the number of required training samples N increases by a factor of $1/(\Theta(1) - \tau)^2$ (Li et al., 2023b). This aligns with the observed difficulty in achieving convergence when training ViTs with densely inverted data (see Fig. 4).

Remark 4.2. Compared to densely inverted data, using sparsely inverted data can stabilize convergence by reducing the number of required training samples N and iterations T. This is because both N and T are negatively correlated with the fraction of label-relevant patches in α and α^2, respectively (Li et al., 2023b). Using sparsely inverted data can increase α by maintaining foreground patches while discard-

[1] Label-relevant patches refer to the ground-truth outcome of our semantic patch identification.

[2] It is easy to derive that the noise level of inverted data is upper bounded by the sum of the L2-norm distance between inverted and real data and the noise level of the real data.

ing background patches with potential spurious correlations. Furthermore, using sparsely inverted data can decrease the noise level τ by pruning noisy background patches. This inference also aligns well with our experiments (as presented in Fig. 4 and Tab. 5 in App. E).

5. Empirical Study

We conduct comprehensive experiments to validate the efficacy of our approach in reducing time, computational, and memory costs required for inversion. We also verify that our method either maintains or even enhances performance when employing sparsely inverted data for downstream tasks such as model quantization (Sec. 5.1) and knowledge transfer (Sec. 5.2). In Secs. 5.3 and 5.4, we provide the visualization results with ablation studies.

Baselines of dense model inversion. DeepInversion (Yin et al., 2020), a method for dense model inversion, aims to invert entire image areas, as detailed in Sec. 3.1. PSAQ-ViT (Li et al., 2022a) is a variant of DeepInversion tailored for data-free ViT quantization. Its primary distinction from DeepInversion lies in the introduction of extra regularization terms in the inversion loss function, resulting in significantly increased time consumption.

Metrics. To evaluate the efficiency of our model inversion approach, we selecte three key metrics: Throughput, FLOPs, and GPU Memory Usage. Note that the results we present are the average values obtained during the model inversion process on one NVIDIA GeForce RTX 3090 GPU.

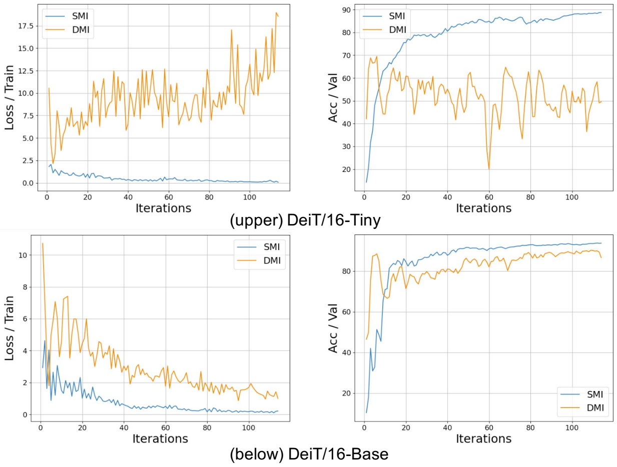
5.1. Experiments on Data-Free Model Quantization

Overview. We aim to verify that sparse inversion can accelerate the inversion process while maintaining or enhancing the downstream performance of model quantization.

Experimental setup. We adopt DeiT/16-Base and DeiT/16-Tiny as the models to be quantized, which are pre-trained on ImageNet for 1000-class classification. All models are accessible from timm. The resolution of inverted images is 224×224. We perform 4000 iterations for inversion using the Adam optimizer with a learning rate of 0.25 (Yin et al., 2020). $\alpha_{\mathcal{R}}$ is set as 1e-4 (Yin et al., 2020). For SMI, we empirically stop 30% of the retained patches at the 50^{th}, 100^{th}, 200^{th}, and 300^{th} iterations, leading to an overall sparsity level of about 77%. The size of the calibration dataset is 32 (Li et al., 2022a). We evaluate different quantization precision for weights and activations, including W4/A8 and W8/A8. The accuracy of the quantized model is reported on the validation set of ImageNet.

Table 4. Knowledge-transfer results on CIFAR10/100 datasets.

Model	Method	Knowledge Transfer (Downstream)			
		Dataset	Top-1	Dataset	Top-1
DeiT/16 -Tiny	Teacher	CIFAR-10	90.23	CIFAR-100	71.66
	DeepInversion (Dense)	CIFAR-10	69.51	CIFAR-100	70.32
	DeepInversion (Sparse)	CIFAR-10	90.08	CIFAR-100	70.48
DeiT/16 -Base	Teacher	CIFAR-10	95.36	CIFAR-100	79.41
	DeepInversion (Dense)	CIFAR-10	90.02	CIFAR-100	74.88
	DeepInversion (Sparse)	CIFAR-10	95.10	CIFAR-100	74.53



(upper) DeiT/16-Tiny

(below) DeiT/16-Base

Figure 4. Impact of utilizing sparsely (blue curve) versus densely (orange curve) inverted data on training loss (left) and validation accuracy (right) throughout the knowledge transfer process.

Results. Tab. 3 illustrates the results. In evaluating efficiency, compared with dense inversion, our approach achieves a range of 2.57 to 3.79-fold speed increase, accompanied by a 74.09%-75.62% reduction in FLOPs and 57.42%-62.98% less GPU memory usage. Importantly, we observe performance gains when using sparsely inverted data compared with using densely inverted data. The reason is that sparsely inverted data allows for a focus on the foregrounds while disregarding noisy backgrounds, thus avoiding potential outliers that detrimentally affect the determination of bounding values ($T_{\min}$ and $T_{\max}$ in Eq. (5)).

5.2. Experiments on Data-Free Knowledge Transfer

Overview. We further examine the effectiveness of sparse model inversion for data-free knowledge transfer.

Experimental setup. We adopt timm-sourced DeiT/16-Tiny and DeiT/16-Base fine-tuned on CIFAR10 and CIFAR100 as teachers, containing knowledge of these specific datasets. We use the vanilla DeiT/16-Tiny and DeiT/16-Base (pre-trained on ImageNet) as student models. The setup of inversion is the same as mentioned in Sec. 5.1. For knowledge transfer, we alternately perform inversion and knowledge transfer at each iteration with a batch size of 128. We implement Eq. (6) with linear probing, using an SGD optimizer with a learning rate of 0.1 and a temperature coefficient of 20. We evaluate the student on the validation sets of CIFAR10 or CIFAR100.

Results. Tab. 4 verifies the superiority of our approach

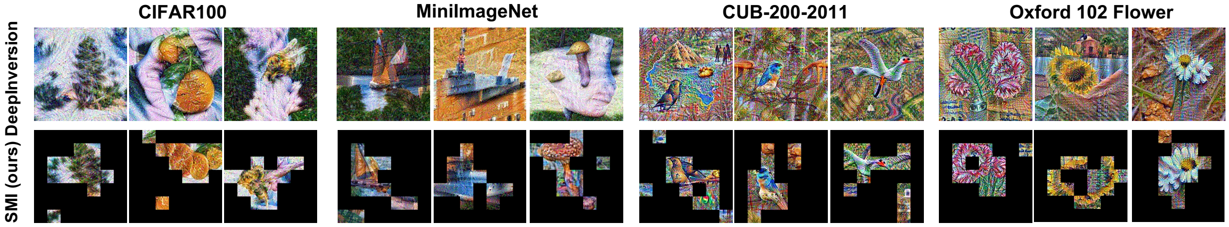
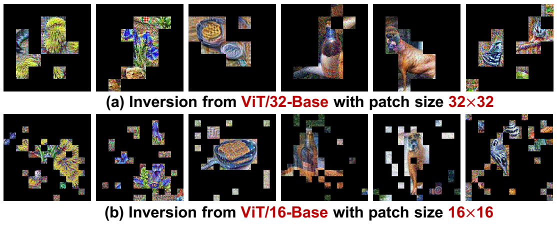


Figure 5. Our inverted images of 224 × 224 pixels from ViT/32-Base encompass a wide range of datasets, from natural images (CIFAR100 and MiniIma- geNet) to more specialized categories (Oxford 102 Flower for various flower species and CUB-200-2011 for bird species).



(a) Inversion from ViT/32-Base with patch size 32×32

(b) Inversion from ViT/16-Base with patch size 16×16

Figure 6. Inversion with different patch-size settings.

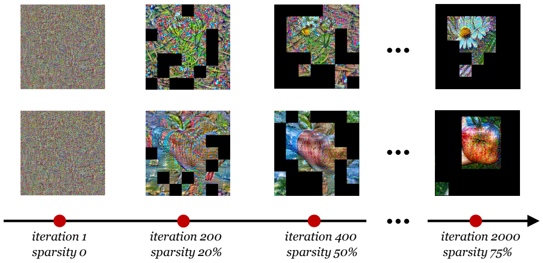


Figure 7. Visualization of the inversion process.

when applied to data-free knowledge transfer. Apart from the acceleration benefits shown in Tab. 3, using sparsely inverted data from our approach can maintain or even enhance the performance of knowledge transfer compared to using densely inverted data. Let us take a deeper look at the convergence process of knowledge transfer on CIFAR10 illustrated in Fig. 4. A critical observation is that using densely inverted data (referring to the orange curve) markedly damages the convergence of the student model, causing a decelerated convergence rate (for DeiT/16-Base) or even a training failure (for DeiT/16-Tiny). Remarkably, switching to sparsely inverted data (referring to the blue curve), without modifying any other settings, results in stable and faster convergence. This finding aligns well with our previous convergence analysis in Sec. 4.4, suggesting that using inverted data is prone to issues of slow- or non-convergence, yet using sparsely inverted data can significantly stabilize and speed up convergence.

5.3. Visualization

Inversion of multiple datasets. To validate the versatility of our approach in inverting images for a broad spectrum of datasets, we visualize the images inverted from CLIP-based ViT/32-Base[3] because features inverted from such large-scale models tend to align more closely with human perception (Ilyas et al., 2019). These models are seperately fine-tuned on CIFAR100 (Bertinetto et al., 2018), MiniImageNet (Vinyals et al., 2016), Oxford 102 Flower (Nilsback & Zisserman, 2008), and CUB-200-2011 (Wah et al., 2011). Fig. 5 visually demonstrates how our method effectively retains the semantic foregrounds while excluding the noisy backgrounds and potential spurious correlations.

Inversion with different patch-size configurations. Pretrained ViTs typically have a fixed patch size setting. In Fig. 6, we perform inversion from ViT/16-Base and ViT/32-Base with patch sizes of 16 × 16 and 32 × 32, respectively. The visualization showcases our approach's adaptability to different patch-size settings, effectively focusing on semantic foregrounds and discarding uninformative backgrounds.

Inversion process. Fig. 7 visualize the process of sparse inversion. As the inversion progresses, our approach selectively inverts semantic patches while progressively stopping inverting uninformative patches (marked as black blocks).

5.4. Ablation Studies

Effect of sparsity level. Here, we evaluate the performance of data-free knowledge transfer on CIFAR10 and the teacher model is DeiT/16-Tiny. Fig. 8 shows that as the sparsity level of the inverted data increases, the inversion process speeds up considerably, and the performance of knowledge transfer significantly improves. Besides, we also find the convergence becomes more stable and quicker, in alignment with our analysis detailed in Sec. 4.4.

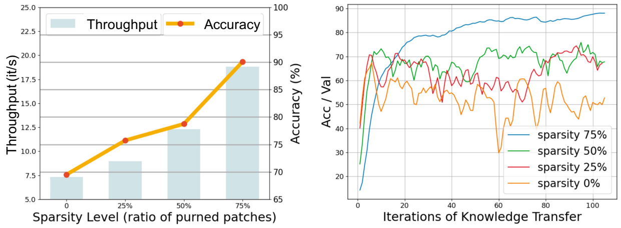


Figure 8. Effect of the sparsity level of inverted data on inversion speed (*i.e.*, throughput), performance and convergence of data-free knowledge transfer.

[3] https://huggingface.co/openai/clip-vit-base-patch32

6. Conclusion

In this paper, we reveal the limitations of existing dense inversion methods, *i.e.*, the inefficiency of inverting high-resolution images from large-scale ViTs. We further identify two underlying causes: the redundant inversion of uninformative backgrounds and the unintended inversion of spurious correlations—a phenomenon we term "hallucination" in model inversion. To address these limitations, we propose the sparse model inversion strategy, as a plug-and-play extension to speed up existing dense inversion with no need for modifying the original loss functions. Specifically, it selectively inverts semantic foregrounds while stopping the inversion of noisy backgrounds and potential spurious correlations. Comprehensive theoretical and empirical studies validate our efficacy in achieving significant inversion acceleration (up to $\times 3.79$) while maintaining comparable or even enhanced downstream performance in data-free model quantization and data-free knowledge transfer.

Acknowledgements

This work was supported by the National Key R&D Program of China (2022YFB4701400/4701402), SSTIC Grant(KJZD20230923115106012), Shenzhen Key Laboratory (ZDSYS20210623092001004), and Beijing Key Lab of Networked Multimedia. Dr Tao's research is partially supported by NTU RSR and Start Up Grants.

Impact Statement

A potential concern is the risk of model inversion inadvertently revealing sensitive information embedded in the original data. However, since we do not focus on synthesizing high-quality images like the works in the generation community, the inverted data dose not accurately reproduce the original images.

References

Bertinetto, L., Henriques, J. F., Torr, P. H., and Vedaldi, A. Meta-learning with differentiable closed-form solvers. *arXiv preprint arXiv:1805.08136*, 2018.

Bica, I., Jarrett, D., and van der Schaar, M. Invariant causal imitation learning for generalizable policies. *Advances in Neural Information Processing Systems*, 34:3952–3964, 2021.

Bolya, D., Fu, C.-Y., Dai, X., Zhang, P., Feichtenhofer, C., and Hoffman, J. Token merging: Your ViT but faster. In *International Conference on Learning Representations*, 2023.

Braun, S., Mundt, M., and Kersting, K. Deep classifier mimicry without data access. *arXiv preprint arXiv:2306.02090*, 2023.

Carta, A., Cossu, A., Lomonaco, V., and Bacciu, D. Ex-model: Continual learning from a stream of trained models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, pp. 3790–3799, June 2022.

Chang, S., Wang, P., Lin, M., Wang, F., Zhang, D. J., Jin, R., and Shou, M. Z. Making vision transformers efficient from a token sparsification view. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6195–6205, 2023.

Chaudhuri, A., Bhunia, A. K., Song, Y.-Z., and Dutta, A. Data-free sketch-based image retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 12084–12093, June 2023.

Chawla, A., Yin, H., Molchanov, P., and Alvarez, J. Data-free knowledge distillation for object detection. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3289–3298, 2021.

Chen, H., Wang, Y., Xu, C., Yang, Z., Liu, C., Shi, B., Xu, C., Xu, C., and Tian, Q. Data-free learning of student networks. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 3514–3522, 2019.

Chen, M., Lin, M., Li, K., Shen, Y., Wu, Y., Chao, F., and Ji, R. Cf-vit: A general coarse-to-fine method for vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7042–7052, 2023a.

Chen, X., Wang, Y., Yan, R., Liu, Y., Guan, T., and He, Y. Texq: Zero-shot network quantization with texture feature distribution calibration. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023b.

Choi, K., Hong, D., Park, N., Kim, Y., and Lee, J. Qimera: Data-free quantization with synthetic boundary supporting samples. *Advances in Neural Information Processing Systems*, 34:14835–14847, 2021.

Choi, Y., Choi, J., El-Khamy, M., and Lee, J. Data-free network quantization with adversarial knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pp. 710–711, 2020.

Chundawat, V. S., Tarun, A. K., Mandal, M., and Kankanhalli, M. Zero-shot machine unlearning. *IEEE Transactions on Information Forensics and Security*, 2023.

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Fang, G., Song, J., Shen, C., Wang, X., Chen, D., and Song, M. Data-free adversarial distillation. *arXiv preprint arXiv:1912.11006*, 2019.

Fang, G., Song, J., Wang, X., Shen, C., Wang, X., and Song, M. Contrastive model inversion for data-free knowledge distillation. *arXiv preprint arXiv:2105.08584*, 2021.

Fang, G., Mo, K., Wang, X., Song, J., Bei, S., Zhang, H., and Song, M. Up to 100x faster data-free knowledge distillation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 6597–6604, 2022.

Fayyaz, M., Koohpayegani, S. A., Jafari, F. R., Sengupta, S., Joze, H. R. V., Sommerlade, E., Pirsiavash, H., and Gall, J. Adaptive token sampling for efficient vision transformers. In *European Conference on Computer Vision*, pp. 396–414. Springer, 2022.

Finn, C., Abbeel, P., and Levine, S. Model-agnostic meta-learning for fast adaptation of deep networks. In *International conference on machine learning*, pp. 1126–1135. PMLR, 2017.

Fredrikson, M., Jha, S., and Ristenpart, T. Model inversion attacks that exploit confidence information and basic countermeasures. In *Proceedings of the 22nd ACM SIGSAC conference on computer and communications security*, pp. 1322–1333, 2015.

Ghosal, S. S. and Li, Y. Are vision transformers robust to spurious correlations? *International Journal of Computer Vision*, pp. 1–21, 2023.

Hatamizadeh, A., Yin, H., Roth, H. R., Li, W., Kautz, J., Xu, D., and Molchanov, P. Gradvit: Gradient inversion of vision transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10021–10030, 2022.

Haurum, J. B., Escalera, S., Taylor, G. W., and Moeslund, T. B. Which tokens to use? investigating token reduction in vision transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 773–783, 2023.

He, X., Lu, J., Xu, W., Hu, Q., Wang, P., and Cheng, J. Generative zero-shot network quantization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 3000–3011, 2021.

He, Z., Zhang, T., and Lee, R. B. Model inversion attacks against collaborative inference. In *Proceedings of the 35th Annual Computer Security Applications Conference*, pp. 148–162, 2019.

Hu, Z., Zhao, Z., Yi, X., Yao, T., Hong, L., Sun, Y., and Chi, E. Improving multi-task generalization via regularizing spurious correlation. *Advances in Neural Information Processing Systems*, 35:11450–11466, 2022.

Hu, Z., Shen, L., Lai, S., and Yuan, C. Task-adaptive feature disentanglement and hallucination for few-shot classification. *IEEE Transactions on Circuits and Systems for Video Technology*, 2023a.

Hu, Z., Shen, L., Wang, Z., Liu, T., Yuan, C., and Tao, D. Architecture, dataset and model-scale agnostic data-free meta-learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7736–7745, 2023b.

Hu, Z., Shen, L., Wang, Z., Wu, B., Yuan, C., and Tao, D. Learning to learn from apis: Black-box data-free meta-learning. In *International Conference on Machine Learning*, 2023c.

Ilyas, A., Santurkar, S., Tsipras, D., Engstrom, L., Tran, B., and Madry, A. Adversarial examples are not bugs, they are features. *Advances in neural information processing systems*, 32, 2019.

Kim, J. M., Koepke, A., Schmid, C., and Akata, Z. Exposing and mitigating spurious correlations for cross-modal retrieval. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2584–2594, 2023.

Kim, M., Gao, S., Hsu, Y.-C., Shen, Y., and Jin, H. Token fusion: Bridging the gap between token pruning and token merging. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 1383–1392, 2024.

Li, H., Wang, M., Liu, S., and Chen, P. A theoretical understanding of shallow vision transformers: Learning, generalization, and sample complexity. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net, 2023a. URL https://openreview.net/pdf?id=jClGv3Qjhb.

Li, H., Wang, M., Liu, S., and Chen, P.-Y. A theoretical understanding of shallow vision transformers: Learning, generalization, and sample complexity. *arXiv preprint arXiv:2302.06015*, 2023b.

Li, X., Wang, S., Sun, J., and Xu, Z. Variational data-free knowledge distillation for continual learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023c.

Li, Z., Ma, L., Chen, M., Xiao, J., and Gu, Q. Patch similarity aware data-free quantization for vision transformers. In *European Conference on Computer Vision*, pp. 154–170. Springer, 2022a.

Li, Z., Ma, L., Chen, M., Xiao, J., and Gu, Q. Patch similarity aware data-free quantization for vision transformers. In *European Conference on Computer Vision*, pp. 154–170, 2022b.

Li, Z., Chen, M., Xiao, J., and Gu, Q. Psaq-vit v2: Toward accurate and general data-free quantization for vision transformers. *IEEE Transactions on Neural Networks and Learning Systems*, 2023d.

Liang, Y., Ge, C., Tong, Z., Song, Y., Wang, J., and Xie, P. Not all patches are what you need: Expediting vision transformers via token reorganizations. *CoRR*, abs/2202.07800, 2022. URL https://arxiv.org/abs/2202.07800.

Liu, Y., Wei, Y.-S., Yan, H., Li, G.-B., and Lin, L. Causal reasoning meets visual representation learning: A prospective study. *Machine Intelligence Research*, 19 (6):485–511, 2022a.

Liu, Y., Wu, Y.-H., Sun, G., Zhang, L., Chhatkuli, A., and Van Gool, L. Vision transformers with hierarchical attention. *Machine Intelligence Research*, pp. 1–14, 2024.

Liu, Z., Shen, Z., Long, Y., Xing, E., Cheng, K.-T., and Leichner, C. Data-free neural architecture search via recursive label calibration. In *European Conference on Computer Vision*, pp. 391–406. Springer, 2022b.

Liu, Z., Oguz, B., Zhao, C., Chang, E., Stock, P., Mehdad, Y., Shi, Y., Krishnamoorthi, R., and Chandra, V. Llm-qat: Data-free quantization aware training for large language models. *arXiv preprint arXiv:2305.17888*, 2023.

Lopes, R. G., Fenu, S., and Starner, T. Data-free knowledge distillation for deep neural networks. *arXiv preprint arXiv:1710.07535*, 2017.

Meng, L., Li, H., Chen, B.-C., Lan, S., Wu, Z., Jiang, Y.-G., and Lim, S.-N. Adavit: Adaptive vision transformers for efficient image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12309–12318, 2022.

Nayak, G. K., Rawal, R., and Chakraborty, A. Dad: Data-free adversarial defense at test time. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 3562–3571, 2022.

Nilsback, M.-E. and Zisserman, A. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pp. 722–729. IEEE, 2008.

Ojha, U., Li, Y., Sundara Rajan, A., Liang, Y., and Lee, Y. J. What knowledge gets distilled in knowledge distillation? *Advances in Neural Information Processing Systems*, 36, 2024.

Patel, G., Mopuri, K. R., and Qiu, Q. Learning to retain while acquiring: Combating distribution-shift in adversarial data-free knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7786–7794, 2023.

Qin, H., Ding, Y., Zhang, X., Wang, J., Liu, X., and Lu, J. Diverse sample generation: Pushing the limit of generative data-free quantization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2023.

Rao, Y., Zhao, W., Liu, B., Lu, J., Zhou, J., and Hsieh, C.-J. Dynamicvit: Efficient vision transformers with dynamic token sparsification. *Advances in neural information processing systems*, 34:13937–13949, 2021.

Sanyal, S., Addepalli, S., and Babu, R. V. Towards data-free model stealing in a hard label setting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15284–15293, 2022.

Shao, R., Zhang, W., Yin, J., and Wang, J. Data-free knowledge distillation for fine-grained visual categorization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 1515–1525, 2023.

Smith, J., Hsu, Y.-C., Balloch, J., Shen, Y., Jin, H., and Kira, Z. Always be dreaming: A new approach for data-free class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 9374–9384, 2021.

Truong, J.-B., Maini, P., Walls, R. J., and Papernot, N. Data-free model extraction. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 4771–4780, 2021.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Vinyals, O., Blundell, C., Lillicrap, T., Wierstra, D., et al. Matching networks for one shot learning. *Advances in neural information processing systems*, 29, 2016.

Wah, C., Branson, S., Welinder, P., Perona, P., and Belongie, S. *The Caltech-UCSD Birds-200-2011 Dataset*. 7 2011.

Wang, Y., Huang, R., Song, S., Huang, Z., and Huang, G. Not all images are worth 16x16 words: Dynamic vision transformers with adaptive sequence length. *arXiv preprint arXiv:2105.15075*, 2(3):8, 2021.

Wang, Z., Wang, X., Shen, L., Suo, Q., Song, K., Yu, D., Shen, Y., and Gao, M. Meta-learning without data via wasserstein distributionally-robust model fusion. In *Uncertainty in Artificial Intelligence*, pp. 2045–2055. PMLR, 2022.

Wang, Z., Yang, E., Shen, L., and Huang, H. A comprehensive survey of forgetting in deep learning beyond continual learning. *arXiv preprint arXiv:2307.09218*, 2023.

Wang, Z., Li, Y., Shen, L., and Huang, H. A unified and general framework for continual learning. *arXiv preprint arXiv:2403.13249*, 2024.

Wei, Y., Hu, Z., Wang, Z., Shen, L., Yuan, C., and Tao, D. Free: Faster and better data-free meta-learning. *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2024.

Xu, S., Li, H., Zhuang, B., Liu, J., Cao, J., Liang, C., and Tan, M. Generative low-bitwidth data free quantization. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XII 16*, pp. 1–17. Springer, 2020.

Xu, Y., Zhang, Z., Zhang, M., Sheng, K., Li, K., Dong, W., Zhang, L., Xu, C., and Sun, X. Evo-vit: Slow-fast token evolution for dynamic vision transformer. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pp. 2964–2972, 2022.

Yang, E., Wang, Z., Shen, L., Yin, N., Liu, T., Guo, G., Wang, X., and Tao, D. Continual learning from a stream of apis. *arXiv preprint arXiv:2309.00023*, 2023.

Yang, Z., Chang, E.-C., and Liang, Z. Adversarial neural network inversion via auxiliary knowledge alignment. *arXiv preprint arXiv:1902.08552*, 2019.

Ye, H., Zou, J., and Zhang, L. Freeze then train: Towards provable representation learning under spurious correlations and feature noise. In *International Conference on Artificial Intelligence and Statistics*, pp. 8968–8990. PMLR, 2023.

Yin, H., Molchanov, P., Alvarez, J. M., Li, Z., Mallya, A., Hoiem, D., Jha, N. K., and Kautz, J. Dreaming to distill: Data-free knowledge transfer via deepinversion. In *The IEEE/CVF Conf. Computer Vision and Pattern Recognition (CVPR)*, 2020.

Yu, S., Chen, J., Han, H., and Jiang, S. Data-free knowledge distillation via feature exchange and activation region constraint. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24266–24275, 2023.

Zhang, J., Chen, C., Li, B., Lyu, L., Wu, S., Ding, S., Shen, C., and Wu, C. Dense: Data-free one-shot federated learning. *Advances in Neural Information Processing Systems*, 35:21414–21428, 2022a.

Zhang, J., Li, B., Xu, J., Wu, S., Ding, S., Zhang, L., and Wu, C. Towards efficient data free black-box adversarial attack. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15115–15125, 2022b.

Zhang, L., Shen, L., Ding, L., Tao, D., and Duan, L.-Y. Fine-tuning global model via data-free knowledge distillation for non-iid federated learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10174–10183, 2022c.

Zhang, Y., Chen, H., Chen, X., Deng, Y., Xu, C., and Wang, Y. Data-free knowledge distillation for image super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7852–7861, 2021.

Zhu, Z., Hong, J., and Zhou, J. Data-free knowledge distillation for heterogeneous federated learning. In *International conference on machine learning*, pp. 12878–12889. PMLR, 2021.

Appendix

A. Model Access and Data Processing

The pre-trained models used in our experiments are all publicly accessible via the Pytorch code or link shown below:

DeiT/16-Tiny (pytorch): `timm.create_model("deit_tiny_patch16_224",pretrained=True)`

DeiT/16-Base (pytorch): `timm.create_model("deit_base_patch16_224",pretrained=True)`

ViT/32-Base: `https://huggingface.co/openai/clip-vit-base-patch32`

ViT/16-Base: `https://huggingface.co/openai/clip-vit-base-patch16`

We implement data augmentation, including Random Horizontal Flip and normalization, to process inverted data. For the patches discarded in the sparsely inverted data, we simply ignore them and do not perform any additional processing on them. We only use resize and normalization to process test data. All images are resized to the resolution of 224×224.

B. More Discussions on Experimental Results

Different quantization performance gains across model scales. In Tab. 3, we observe that using sparsely inverted data for model quantization can achieve greater performance gains on DeiT-Base than on DeiT-Tiny. This trend is also found when using densely inverted data[4] (Li et al., 2022a;b). A plausible explanation for this phenomenon is the better foreground extraction capabilities of larger models. When inverting images from such larger models, the inversion process can target foregrounds more precisely. This more precise focus on the foregrounds can allow for the more accurate determination of bounding values ($T_{\min}$ and $T_{\max}$ in Eq. (5)), and consequently leading to greater performance gains. Moreover, our approach with sparsely inverted data goes a step further by explicitly discarding those noisy backgrounds. This removal can effectively reduce the distraction of outliers to the determination of bounding values, thus further amplifying the beneficial effect of foreground extraction.

C. More Applications of Model Inversion

Model inversion is often utilized to synthesize surrogate data directly from the discriminative model, proving highly useful in data-constrained real-world scenarios. Besides **model quantization** (Liu et al., 2023; Choi et al., 2020; He et al., 2021; Chen et al., 2023b) and **knowledge transfer** (Yin et al., 2020; Fang et al., 2019; Chen et al., 2019), which we discussed in Sec. 2 of the main paper, data-constrained situations also arise in meta-learning, continual learning, federated learning, and other applications. In these contexts, model inversion can offer an effective solution.

Meta-learning. Meta-learning (Finn et al., 2017; Bertinetto et al., 2018; Hu et al., 2023a) necessitates meta-training on a vast array of related tasks, typically represented by task-specific training and testing sets. However, in the real world, acquiring a large number of meta-training tasks with labelled data is challenging due to issues like data privacy or annotation costs. Based on this real-world setting, data-free meta-learning (Wang et al., 2022) aims to conduct meta-training on tasks using only pre-trained models, without access to corresponding datasets. Existing work (Hu et al., 2023b;c; Wei et al., 2024) has also employed model inversion for data-free meta-learning to address the issue of data inaccessibility.

Federated learning. Federated learning is a method to train a global server model without accessing the training data stored on each client. A typical approach involves each client uploading their self-trained model, which the server then merges into a single global model. Thus, we can use model inversion to invert data from client models, aiding the server in better integrating these models. For specific methodologies, please refer to (Zhu et al., 2021; Zhang et al., 2022a;c)

Continual learning. Continual learning (Wang et al., 2024; 2023) aims to learn new tasks while retaining knowledge of old tasks. A classic approach is to store training data from old tasks and re-learn these along with new tasks. However, the data for old tasks may be inaccessible due to reasons like privacy concerns or storage costs. In such cases, we can employ model inversion to infer data from old tasks from the model. For specific methodologies, please refer to (Smith et al., 2021; Li et al., 2023c; Carta et al., 2022; Yang et al., 2023).

[4] The performance gains of using sparsely inverted data are based on the comparison with using densely inverted data, while the performance gains of using densely inverted data are based on the comparison with using real data.

Other applications include image retrieval (Chaudhuri et al., 2023), neural architecture search (Liu et al., 2022b), model extraction (Truong et al., 2021; Sanyal et al., 2022), adversarial attack (Zhang et al., 2022b), adversarial defense (Nayak et al., 2022), object detection (Chawla et al., 2021), and image super-resolution (Zhang et al., 2021).

Previous methods typically employ model inversion as a tool to synthesize surrogate data, while our work is the first to enhance the scalability of model inversion for inverting high-resolution images from large-scale ViTs.

D. Detailed Theoretical Analysis

Here, we go into more detail about our analysis study to investigate how sparsely inverted data affect the convergence conditions, including the number of required training samples N and iterations T. Our study is based on the setting of Li et al. (2023b), in the context of training ViTs for classification. Below, we first introduce some specific setups unique to our sparsely inverted data.

Setup. (i) Patch sparsity setup: Consider N inverted samples $\{(\boldsymbol{X}^n, y^n)\}_{n=1}^N$, where each $\boldsymbol{X}^n$ comprises L' retained patches $[\boldsymbol{x}_1^n, ..., \boldsymbol{x}_{L'}^n]$, $(L' \leq L)$. Let $\mathcal{S}^n \subseteq [L']$ represent the indices of label-relevant patches in $\boldsymbol{X}^n$. Label-relevant patches refer to the ground-truth outcome of our semantic patch identification. We define the average fraction of label-relevant patches as $\alpha = \sum_{n=1}^N \frac{|\mathcal{S}^n|}{N \cdot L'}$. (ii) Patch noise setup: Label-relevant patches in $\boldsymbol{x}^n$ correspond to specific patterns $\boldsymbol{\mu}_{y^n}$ of its label y^n with a noise level τ, satisfying $\left\|\boldsymbol{x}^n - \boldsymbol{\mu}_{y^n}\right\|_2 \leq \tau$. Other patches in $\boldsymbol{X}^n$, however, correspond to patterns of other labels or just noise. (iii) Convergence condition: We denote the required number of training samples and iterations as N and T, respectively.

Lemma D.1. *(Li et al., 2023a) Under certain assumptions, a ViT with initial errors σ and δ for value and query/key vectors respectively, and trained via SGD with step size η, can achieve zero generalization error (i.e., population risk achieves zero) with a probability of at least 0.99. This outcome is conditioned upon the sample complexity N and iteration numbers T:*

$$\alpha \geq \frac{1-\alpha}{e^{-(\delta+\tau)}(1-(\sigma+\tau))}, \quad T = \Theta\left(\eta^{-3/5}\alpha^{-1}\right), \tag{8a}$$

$$N \geq \Omega\left(\frac{1}{(\alpha - c'(1-\zeta) - c''(\sigma+\tau))^2}\right), \tag{8b}$$

where $c', c'' > 0$ are constants, and $\zeta \gtrsim 1 - \eta^{10}$.

Compared to real data, using densely inverted data makes the convergence process more challenging. This is primarily due to the inherent higher noise level τ in inverted data. It is easy to derive that the noise level of inverted data is upper bounded by the sum of the distance between inverted and real data and the noise level of the real data. With the noise level τ increasing, the number of required training samples N increases by a factor of $1/(\Theta(1) - \tau)^2$ (Li et al., 2023b). This aligns with the observed difficulty in achieving convergence when training ViTs with densely inverted data (see Fig. 4).

Compared to densely inverted data, using sparsely inverted data can stabilize convergence by reducing the number of required training samples N and iterations T. This is because both N and T are negatively correlated with the fraction of label-relevant patches in α and α^2, respectively (Li et al., 2023b). Using sparsely inverted data can increase α by maintaining foreground patches while discarding background patches with potential spurious correlations. Furthermore, using sparsely inverted data can decrease the noise level τ by pruning noisy background patches. This inference also aligns well with our experiments (as presented in Fig. 4 and Tab. 5 in App. E).

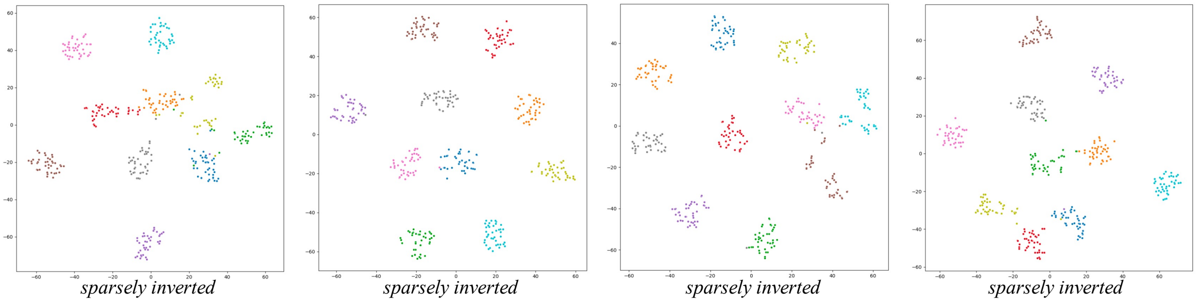
E. More Experiments

Effect of sparsely inverted data on convergence conditions of knowledge transfer. In addition to Figs. 4 and 8, which illustrates how using sparsely inverted data stabilizes and accelerates the convergence process in knowledge transfer, this section provides a quantitative analysis of the impact of using sparsely inverted data on the convergence conditions in knowledge transfer, including the required number of training samples (N) and iterations (T). Tab. 5 compares the number of inverted samples and iterations needed to achieve the same accuracy (as per the maximum test accuracy achieved using densely inverted data) when using sparsely versus densely inverted data. In this experiment, we invert 128 CIFAR-10 data from DeiT/16-Base per iteration, which is used to perform knowledge transfer (Eq. (6)). Tab. 5 demonstrate that sparsely inverted data leads to faster convergence and requires fewer training samples to reach the same level of accuracy. These results also align well with our analytical study in Sec. 4.4.

Table 5. Impact of using sparsely versus densely inverted data on convergence conditions of knowledge transfer. We compare the number of inverted samples and iterations required to achieve the same accuracy. For each iteration of knowledge transfer, we invert 128 CIFAR-10 data from DeiT/16-Base, and then use them to perform knowledge transfer (Eq. (6)).

Training Data	**Knowledge Transfer**		
	Test Accuracy	**Sample Complexity** (N)	**Iteration Counts** (T)
Densely Inverted Data (DeepInversion)	90.02	14080	110
Sparsely Inverted Data (SMI)	90.02	5540	43

T-SNE visualization. Fig. 9 presents the t-SNE visualizations of pseudo images inversed from ViT/32-Base on diverse datasets, namely CIFAR100, MiniImageNet (which is a subset of ImageNet featuring diverse images), VGG-Flower (dedicated to detailed flower species classification), and CUB (focused on fine-grained bird categorization). These visualizations effectively highlight our method's ability to invert essential discriminative features.



sparsely inverted CIFAR100 *sparsely inverted MiniImageNet* *sparsely inverted CUB-200-2011* *sparsely inverted Oxford 102 Flower*

Figure 9. T-SNE visualization of sparsely inverted data with the sparsity level of 77%.

Effect of progressive stopping. In Tab. 6, we compare the effects of one-stage and progressive multi-stage stopping on the downstream performance of data-free knowledge distillation. With the same sparsity level in the inverted data, multi-stage pruning provides greater performance gains for data-free knowledge transfer due to its progressive refinement in identifying semantic patches.

Table 6. Effect of progressive stopping. Under the same sparsity goal, we implement one-stage stopping at the 100 th iteration, and multi-stage with the same inversion setting in Sec. 5.1. We evaluate the performance of knowledge transfer on CIFAR10 and the pre-trained model is DeiT/16-Tiny.

Variants	**Sparsity**	**Data-free knowledge transfer**
one-stage	77%	88.82
multi-stage	77%	90.08

Sensitivity analysis of varied inversion stopping strategies. Tab. 7 shows that the downstream performance of data-free knowledge transfer is not very sensitive to variations in stopping strategy, if the overall sparsity level keep consistent, highlighting the practical adaptability of our approach.

Table 7. Sensitivity analysis of different inversion stopping strategies. We evaluate the performance of knowledge transfer on CIFAR10 and the pre-trained model is DeiT/16-Tiny.

Inversion stopping strategy	**Sparsity**	**Data-free knowledge transfer**
{100: 77%}	77%	88.82
{200: 77%}	77%	89.02
{400: 77%}	77%	89.13
{50: 40%, 150: 60%}	77%	89.26
{100: 40%, 200: 60%}	77%	89.38
{250: 40%, 300: 60%}	77%	90.02
{30: 30%, 70: 30%, 150: 30%, 200: 30%}	77%	89.90
{50: 30%, 100: 30%, 200: 30%, 300: 30%}	77%	90.08
{200: 30%, 250: 30%, 350: 30%, 450: 30%}	77%	90.14

Effect of batch size and model scale on speed gains. Tab. 8 illustrates that (i) increasing the batch size for each iteration of model inversion enhances processing speed, and (ii) enlarging the model scale further amplifies this speed gain.

Table 8. Effect of batch size and model scale on speed gains.

Model Scale	**Batch Size**	**Throughout** (its/s) ↑		
		DeepInversion (Dense)	**DeepInversion** (Sparse)	**Speed Gains**
DeiT/16-Base	32	4.45	14.29	×3.21
	64	2.36	8.30	×3.52
	128	1.19	4.51	×3.79
DeiT/16-Tiny	32	24.78	43.26	×1.75
	64	14.04	33.58	×2.39
	128	7.33	18.82	×2.57
	256	3.76	9.78	×2.60