Visualization Comparison of Vision Transformers and Convolutional Neural Networks

Rui Shi, Tianxing Li, Liguo Zhang, Yasushi Yamaguchi

***Abstract*—Recent research has demonstrated that Vision Transformers (ViTs) are capable of comparable or even better performance than convolutional neural network (CNN) baselines. The differences in their structural designs are obvious, but our understanding of the differences in their feature representations remains limited. In this work, we propose several techniques to achieve high-quality visualization of representations in ViTs. Both qualitative and quantitative experiments show that our technical improvements can observably improve ViT visualization quality compared to previous studies. Furthermore, we conduct visualizations to explore the disparities between ViTs and CNNs pre-trained on ImageNet1K, revealing three intriguing properties of ViTs: (a) ViT feature propagation retains image detail information with minimal loss, whereas CNNs discard most image details for class discrimination. (b) Different from CNNs, object-related features do not show in ViT higher layers, suggesting that class-discriminative features may not be required for ViT classification. (c) Our visualization-assisted texture-bias experiment reveals that both ViTs and CNNs exhibit texture bias, of which ViTs seem to be more biased towards local textures.**

***Index Terms*—Vision Transformer, convolutional neural network, feature representation, optimization visualization.**

I. Introduction

OVER the past several years, convolutional layers have served as *de facto standard* building blocks for almost every vision tasks [1], [2]. This is mainly due to the powerful inductive bias of spatial locality encoded by convolutional operations and the low number of parameters. Recently, motivated by the tremendous success of attention-based Transformer networks in natural language processing tasks, researchers have developed Vision Transformers (ViTs) [3] which are capable of achieving equal or better performance than state-of-the-art ResNets [4] of similar capacity [5]. Subsequently, more advanced network architectures, such as **S**hifted **win**dows Transformer (SwinT) [6], have been proposed that integrate the fundamental principles commonly utilized in convolutional neural networks (CNNs) and Transformers. These networks operate nearly identical to Transformers deployed in language models–with an attention block followed by a multilayer perceptron (MLP) block, but this is obviously in contrast with many prior convolution-based studies focusing on incorporating image inductive biases explicitly.

Such a breakthrough motivates us to study the differences in feature representations between ViTs and CNNs. Are ViT feature propagations similar to CNNs? Or are ViTs developing novel feature representations? In this work, we investigate these questions and provide insights about several key differences between them by achieving *optimization visualization* in ViT. Although our research primarily focuses on the ViTs proposed in [3], we also assess the viability of the proposed method in our experiments using other sophisticated Transformer-based networks, such as SwinT. Generally, *optimization visualizations* are based on the fact that a vision neural network is differentiable with respect to its input image; thus, gradients can be used to iteratively update the input image to seek the kind of image whose feature representations are notable in some meaningful sense.

Specifically, in this work, we mainly focus on two visualization ideas (*i.e.*, *inversion* and *representation maximization*) that have been extensively studied to understand CNNs but are still challenging to apply to ViTs. In the first type, called **inversion** (Sec. III), we reconstruct feature representations of an input image to uncover the feature diversity of identical representations. We do so by computing a representation $\Phi_0 = \Phi(\mathbf{x}_0)$ of the image $\mathbf{x}_0$. Then, we update a randomly-initialized image to reconstruct the information of the representation Φ_0. Notably, a representation Φ is not invertible because it is commonly invariant to some nuisance factors such as perspectives and illuminations [7]. We can analyze the invariance by studying the representation reconstruction $\mathbf{x}^*$ that shares the (nearly) same feature representation with $\mathbf{x}_0$ and observe the loss of image information during feature propagation using the inversions of all layers.

In the second type, referred to as **representation maximization** (Sec. IV), also called "activation maximization" [8], we look for an image $\mathbf{x}^*$ that maximally excites a certain channel feature of the representation Φ. The updated image $\mathbf{x}^*$ is representative of the visual stimuli capable of expressing the implication of a channel or a selected component of feature representations. Differently from inversion, maximization visualization separates the certain channel implication from the meanings of representation combinations.

While conceptually simple, there are two particular challenges caused by the ViT structure when generating natural-looking visualizations. First, patch-based processing produces

Manuscript received September 21, 2022; revised April 23, June 18, 2023; accepted July 10, 2023. We would like to thank J. Zhan, Z. Li, Z. Qiao, and H. Deng for their valuable feedback and comments on the English usage. This work was supported supported in part by the National Natural Science Foundation of China [Grant No. U2233211] and in part by Japan Society for the Promotion of Science (JSPS KAKENHI) [Grant No. 20H04203]. *(Corresponding author: Tianxing Li.)*

Rui Shi, Tianxing Li, and Liguo Zhang are with the Faculty of Information Technology, Beijing University of Technology, Beijing 100124, China (e-mail: ruishi@bjut.edu.cn; litianxing@bjut.edu.cn; zhangliguo@bjut.edu.cn)

Yasushi Yamaguchi is with the Department of General Systems Studies, the University of Tokyo, Tokyo 153-8902, Japan (e-mail: yama@g.ecc.u-tokyo.ac.jp)

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the author. The material includes more experimental results and discussions on ablation studies.

Fig. 1. A visualization full of patch artifacts generated with [8]. This inversion visualization is generated using the feature representations extracted from the 11_{th} layer of ViT-B/16. The optimization strategy and regularization methods employed in this visualization are consistent with those described in the original paper. We can observe disjoint patch edges everywhere in the image as well as some noises indicated with red circles.

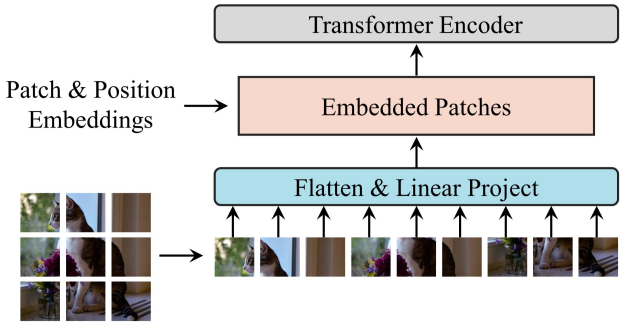


Fig. 2. An overview of the patch-based processing of ViTs. The input image is split into fixed-size patches, which are then linearly projected. Next, position and patch embeddings are added to these projected patches to form the input for ViT encoders. The illustration was inspired by [3].

many patch artifacts, *e.g.*, disjointed patch edges and noise marked by red circles in Fig. 1. Different from CNNs, ViTs first divide the input image into several non-overlapping patches, which are then linearly projected into embedded patches to serve as input to the Transformer encoder, as shown in Fig. 2. This patch-based processing enables ViTs to scale to higher resolutions, but it also results in a distinctive trend of receptive field variation that differs from CNNs. As discussed in [1], [5], ViT receptive fields show a strong dependence on a single patch and do not grow as gradually as CNNs, *i.e.*, the gradients of a single patch have little effect on the other patches, and therefore there is no guarantee that the edges between patches are consistent during optimization. Second, the attention-based structure of a ViT sub-encoder is very different from a convolutional block, as shown in Fig. 3. The self-attention mechanism is widely acknowledged to be highly informative, as it captures global contextual information from the entire image [1], [9]. Based on our experiments (Fig. 8), we observe that if the attention block information is not matched in inversion visualizations, some important information may be lost.

In this work, we address these challenges to achieve ViT visualization and uncover insights about feature representation differences between ViTs and ResNets. More specifically, our contributions are as follows:

- We propose to generate ViT visualizations in the frequency domain and design several indirect regularizations, both of which techniques eliminate patch artifacts.

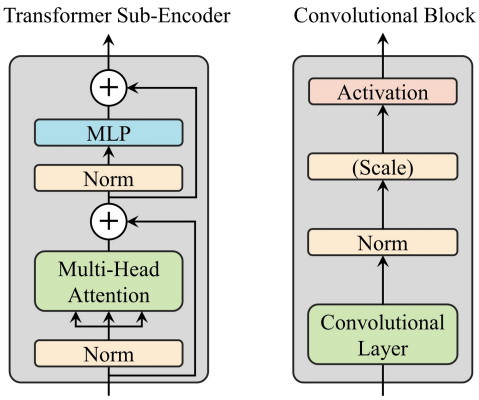


Fig. 3. Left: ViT sub-encoder structure; Right: convolutional block structure. The illustration was inspired by [3].

For inversion, we introduce attention matching into the common objective function to produce clear reconstructions of feature representations. For representation maximization, we visualize positive and negative representations separately and design a color space transform strategy to realize natural-looking visualizations. Ablation study in Sec. V-A demonstrates the importance of proposed techniques to achieve high-quality ViT visualizations. The programming implementation of our proposal is available online (https://bit.ly/net-vis-compare).

- By conducting visualization comparisons, we find three intriguing properties of ViT. First, the ViT inversions are highly consistent among all layers and retain a large amount of image detail information, which indicates that there is almost no information loss in ViT feature propagation. However, CNNs discard a significant amount of low-level image information to achieve discriminative representations (Sec. V-B). Second, our analysis of channel visualizations generated through representation maximization reveals that ViTs may have a lower number of object-related features in higher layers compared to CNNs, despite the fact that these class-discriminative features are typically considered crucial for predictions (Sec. V-C). Third, probably due to permutation invariance, ViTs show a stronger texture bias than ResNets and are more biased towards local textures. Even if the shape of the original object is completely disrupted, ViTs can still predict the true label from some patch-level texture information (Sec. V-D). Additionally, we also conduct visualization experiments to demonstrate the effectiveness of our proposed method on other newly developed networks based on Transformer such as SwinT in Sec. V-B–V-D and find that SwinT shows the characteristics both of ViTs and CNNs.

II. Related Work

Optimization visualization, a method of updating random noise by derivatives into an understandable image, can be applied to process a large variety of representations due to its high flexibility.

Inversion is a visualization method for recovering an image from specific encodings. The original idea of applying an energy minimization framework to invert neural networks was used to study non-image-processing neural networks [10]–[15] and extended to a variety of CNNs [16]–[19]. To understand feature connections among layers, Mahendran *et al.* [8], [20] explored several techniques of inverting deep CNNs and innovatively generalized them into a regularized energy-minimization framework. By exploiting the diversity contained in feature representations, Yin *et al.* [7] proposed DeepInversion that balances all-layer feature representations together to implement knowledge distillation and image generation. This unique processing approach allows for the synthesis of class-conditional input images without requiring any additional information from the original dataset.

Reconstructing the original data by inverting gradients is another important application scenario of inversion techniques. Built upon DeepInversion, Yin *et al.* [21] proposed GradInversion that inverts gradients and distribution statistics of normalization layers together to achieve clear input image reconstruction. More recently, Hatamizadeh *et al.* [22] introduced a gradient-based inversion method called GradViT, which successfully enables inversion generation for ViTs. This represents the first successful attempt at generating inversions for ViTs. They further proposed several additive regularization techniques that effectively improve the inversion quality. Despite producing remarkably good inversion results, we find it difficult to apply previous techniques to visualization directly. For example, image prior regularization designed based on batch normalization statistics obtained from an additional pre-trained CNN is a powerful way to generate natural-looking results; however, image prior implies the statistics of all layers, while visualization usually focuses on a specific layer only. In addition, utilizing an additional pre-trained CNN may interfere with the demonstrated diversity in ViT feature representation reconstructions. Our inversion method differs from the ones above as it does not require any hyperparameters of balancing regularization terms nor any information beyond the visualized layer.

Representation maximization, is used to maximize the response of representations in vision networks. We do not select the name "activation maximization" because there is no concept of activation in ViTs. By maximizing a class score, highly realistic images that are related to the output class from the perspective of a network can be generated [23]–[25]. To understand the implications of individual feature representations, Olah *et al.* [26], [27] visualized individual neurons and neuron combinations with different CNNs and discussed the meanings of sophisticated feature detectors. Maximization visualization was also applied to explore feature representation combinations related to a target output class [28]–[30]. Prior studies have significantly improved the visual quality and stability of maximization visualization on CNNs. Built upon these studies, we further develop several techniques based on the Transformer structure to achieve ViT maximization visualization.

Other technically relevant studies are **image synthesis** and **attribution visualization**. For **image synthesis**, Tancik *et al.* [31] extended neural tangent kernel to a stationary kernel by explicitly modeling Fourier features to achieve the image and 3D shape regression with high-frequency details. Their experimental results demonstrated the great potential of frequency domain processing for high-frequency feature learning in artificial neural networks, which directly inspired us to improve the visual quality by introducing frequency domain optimization into ViT visualization. Tesfaldet *et al.* [32] extended compositional pattern producing networks (CPPNs) to generate frequency coefficients to synthesize high-frequency details which cannot be captured by vanilla CPPNs. Their experiments also showed that frequency processing can effectively improve visual quality in image synthesis tasks.

Attribution visualization, also called saliency map [33], mainly concerned with generating heat maps corresponding to an output class [33], [34]. Simonyan *et al.* [35] used saliency information generated from signed gradients to explain network outputs. Bach *et al.* [36] designed several propagation rules according to hidden layer attributes, and thus proposed Layer-wise Relevance Propagation (LRP), which can propagate the relevance information layer by layer to the input features. More recently, several Shapley-value-based attribution methods have been proposed, which satisfy desirable properties like efficiency, symmetry, linearity, *etc* [37]–[41]. These attribution-based visualizations typically produce heat maps to highlight important image regions. In contrast, optimization-based visualization methods aim to generate images that reveal implicit semantic information. In this study, we adopt attributions to identify important channel features and generate heat maps as complementary visualizations to representation maximization.

III. Inversion

A. Frequency Domain Optimization

Inversion visualization can be formulated as an energy minimization problem where the goal is to find a visualization whose feature representations are close to the target representations Φ_0. Formally, we model feature representation as a function Φ mapping an input into ViT sub-encoder outputs and seek the visualization $\mathbf{x}^*$ that minimizes the objective function:

$$\mathbf{x}^* = \arg\min_{\mathbf{x}} \mathcal{L}(\Phi(\mathbf{x}), \Phi_0), \tag{1}$$

where $\mathbf{x} \in \mathbb{R}^{H\times W\times K}$ is initialized randomly and optimized to generate the visualization result $\mathbf{x}^*$. Φ_0 is the feature representation of the original image $\mathbf{x}_0$. Borrowing from previous studies [8], [22], the inversion loss function $\mathcal{L}$ is set to the L^2 distance:

$$\mathcal{L}(\Phi(\mathbf{x}), \Phi_0) = \frac{\|\Phi(\mathbf{x}) - \Phi_0\|^2}{\|\Phi_0\|^2}. \tag{2}$$

The optimization is reconstructing the target representation to reflect layer information contained in the original image from the perspective of the network [8].

The paradigm works well in CNNs; however, direct optimization could result in many patch artifacts in ViT visualizations, because of the low correlation among the gradients of different patches. The patch processing makes it difficult

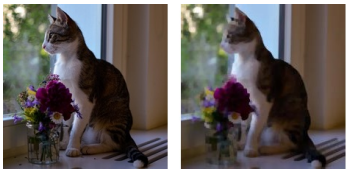


Fig. 4. Left: original image; Right: transformed image: closing followed by opening. Morphological operations have the advantage of better maintaining object contours while blurring internal textures, which contributes to edge-consistent optimization results.

to solve this problem directly in the spatial domain. To solve this problem, we note the property that the Fourier transform can easily access geometric characteristics. More specifically, modifying a point in the frequency domain can change all the information in the spatial domain, which allows us to avoid updating pixels around patch edges individually, thus alleviating patch artifacts. Instead of manually designing the Fourier mapping as in [31], which can adjust the spectral width and thus affect the reconstruction of different frequencies, we adopt the original two-dimensional Fourier transform to ensure that the inversion is not disturbed by the manual selection of high and low frequencies outside the network. With this idea, we transform visualization optimization from the spatial domain to the frequency domain. The inverse discrete Fourier transform along the color channel is:

$$\mathbf{x}(h,w,k) = \frac{1}{HW}\sum_{u,v}\mathbf{F}(u,v,k)e^{\jmath 2\pi(\frac{uh}{H}+\frac{vw}{W})}, \tag{3}$$

where h, w, k are indices of height, width, and color channel, respectively. $\mathbf{F}(u,v,k)$ is the Fourier coefficient for the given frequencies u, v under the channel k. $\jmath = \sqrt{-1}$. Note that, since visualization is real-valued, we just ignore the imaginary part of the result computed by the inverse Fourier transform. For simplicity, we will use f_{ift} to represent the inverse transform operation. Then, the objective function Eq. (1) is transformed to:

$$\mathbf{F}^* = \arg\min_{\mathbf{F}} \mathcal{L}\big(\Phi(f_{ift}(\mathbf{F})), \Phi_0\big), \tag{4a}$$

$$\mathbf{x}^* = f_{ift}\left(\mathbf{F}^*\right). \tag{4b}$$

The introduction of the Fourier transform can also be interpreted as adding an image prior that forces the optimizer to produce a structurally ordered visualization, such that the objective can be more easily minimized.

B. Image Processing Regularization

Permutation invariance is an advantage of ViTs, but the lack of integrated constraints among patches causes patch artifacts in visualizations. To further solve this problem, we propose to add some perturbations with randomness to the image inner contours, forcing the optimization process to find edge-consistent results and thus avoiding the patch artifacts. Adding regularization terms such as total variation and α-norm is a common strategy to achieve constraints, but it introduces additional hyperparameters and cannot achieve good patch consistency in ViT visualization. More discussion and examples can be found in the Sec. D of the supplementary materials.

Inspired by the discussion of regularization techniques [42], we discard direct regularization and choose to develop an indirect regularization strategy for ViTs instead. In particular, we first design two morphology-based regularizers, *i.e.*, random closing and opening, that apply morphological transformations to the image with random structuring elements before forwarding it to the network. As an example shown in Fig. 4, closing followed by opening can smooth the internal textures while maintaining the shape, thus contributing to edge-consistent optimization results. Our morphological structuring element set is defined as follows:

$$\begin{aligned}&\{B(0), B(1), \ldots, B(4)\}\\ &= \left\{\begin{bmatrix}0&1&0\\1&1&1\\0&1&0\end{bmatrix}, \begin{bmatrix}1&1&0\\1&1&1\\0&1&0\end{bmatrix}, \ldots, \begin{bmatrix}0&1&0\\1&1&1\\0&1&1\end{bmatrix}\right\}.\end{aligned}$$

We also select random cropping, a common image augmentation operation, as another indirect regularizer that adds randomness by shifting the image to enforce the optimizer to generate crisper visualization. With these image processing regularizers, the objective function is converted to:

$$\mathbf{F}^* = \arg\min_{\mathbf{F}} \boldsymbol{E}_{\tau,\epsilon}\big[\mathcal{L}\big(\Phi(f_{proc}(\mathbf{F};\tau,\epsilon)), \Phi_0\big)\big], \tag{5a}$$

$$\begin{aligned}&f_{proc}\left(\mathbf{F};\tau,\epsilon\right)\\ &= f_{rc}\Big(f_{open}\Big(f_{close}\big(f_{ift}(\mathbf{F}); B(\tau)\big); B(\tau)\Big); \epsilon\Big),\end{aligned} \tag{5b}$$

where $\boldsymbol{E}_{\tau,\epsilon}[\cdot]$ indicates the expectation of two random variables τ and ϵ. For morphological operations, we select a structuring element from the set B using τ, where τ is a discrete random variable uniformly distributed in the set $\{0, 1, \ldots, 4\}$. ϵ is also a random variable uniformly distributed in the set $\{-T, \ldots, T\}$ which means ϵ pixels are cropped along width and height dimension, and the space is padded in reflect mode. Empirically, T can be set at the greatest integer less than $(W+H)/100$ for ViT inversion. The ablation study in V-A shows that both image processing regularization and frequency domain optimization have significant effects on artifact removal.

C. Attention Matching

Each sub-encoder is considered as a ViT layer, but different from a convolutional block, the sub-encoder contains two important blocks, *i.e.*, the multi-head self-attention (MSA) block and the MLP block. In CNN inversion, the feature representation usually refers to the output of the activation layer in a convolutional block. However, we find that considering only the final output of the sub-encoder makes the inversion optimization difficult and does not reconstruct the feature representation even in shallow layers. Without an additional constraint, the optimization process cannot construct the accurate quantities of outputs of these two blocks; therefore,

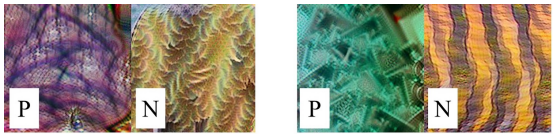


Fig. 5. Examples of positive and negative representation maximization. The positive and negative visualizations are generated with the same channel but show completely different patterns.

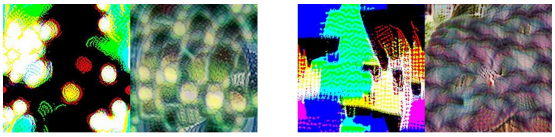


Fig. 6. The left side of each pair shows the maximization visualization without the color space transform, and the right side shows the case with the color space transform. Although all of them can highly activate the particular channel, results with the color space transform are more natural to human eyes.

we add an attention matching term using the output of the MSA block and modify the loss function as follows:

$$\mathcal{L}\big(\Phi(f_{proc}(\mathbf{F};\tau,\epsilon)),\Phi_0,\Phi_0^{attn}\big)=\frac{\|\Phi(f_{proc}(\mathbf{F};\tau,\epsilon))-\Phi_0\|^2}{\|\Phi_0\|^2}+\frac{\|\Phi^{attn}(f_{proc}(\mathbf{F};\tau,\epsilon))-\Phi_0^{attn}\|^2}{\|\Phi_0^{attn}\|^2},\tag{6}$$

where Φ^{attn} means the representation function of the attention block, and Φ_0^{attn} is the representation of the attention block of the original image. It seems that this newly introduced attention term needs a hyperparameter to balance, but the range of differences between two residual-connected blocks is small that we therefore directly add this term in practice. With these two references, we can reconstruct representations in a sub-encoder more accurately, such that clearer inversion can be generated.

IV. Representation Maximization

A. Maximization Loss

Representation maximization can express the channel implication by finding a visualization that highly activates the channel. In previous CNN visualization studies, the maximization loss function is defined by summing a channel or several neurons thereof [23]. In ViTs, it can be obtained similarly by considering the summation of all patches of one channel:

$$\mathcal{L}\big(\Phi(f_{ift}(\mathbf{F}));d\big)=\sum_{i=1}^{N}\Phi(f_{ift}(\mathbf{F}))(i,d),\tag{7}$$

where $\Phi(\cdot)(i,d)$ stands for the i_{th} patch in the d_{th} channel of the feature representation. N is equal to one (class token) plus the patch number (*e.g.*, 1+196 in ViT-B/16). For maximization visualization, the class token does not contain valid channel feature information, thus the accumulation starts from $i=1$. In addition, there is no activation function like ReLU before ViT sub-encoder outputs, *i.e.*, maximizing and minimizing the loss function can both produce meaningful visualizations in the channel d. Fig. 5 shows positive and negative visualizations of two channels in a middle layer of ViT-B/16, and we can find that the positive and negative maximization values of one channel correspond to completely different patterns. On the other hand, maximization visualizations are not directly correlated with a particular network output; thus, when analyzing the channel features corresponding to a target output, we use an Aumann-Shapley-based attribution method [40] to compute channel contribution scores and detect the channels contributing to the particular output as stated in Sec. V-C.

B. Color Space Transform

Different from the inversion with constrained optimization objectives, we observe that ViTs cannot maintain reasonable correlations between image channels during maximization optimization. The representation maximization in the frequency domain produces results that can highly activate the specific representation but with over-saturated colors, as shown in Fig. 6. Visually, these results are highly similar to the channel-decorrelated image after the Karhunen-Loève transform (KLT) [43], which motivates us to apply the color space transform inversely to constrain the pixels in a range more suitable for human eyes. The KLT matrix $\boldsymbol{A}$ is defined as:

$$\boldsymbol{C}=\begin{bmatrix}C_{RR}&C_{RG}&C_{RB}\\C_{GR}&C_{GG}&C_{GB}\\C_{BR}&C_{BG}&C_{BB}\end{bmatrix}\tag{8a}$$

$$\boldsymbol{A}=\begin{bmatrix}\boldsymbol{v}_1&\boldsymbol{v}_2&\boldsymbol{v}_3\end{bmatrix},\tag{8b}$$

where $\boldsymbol{C}$ is the covariance matrix of the image color channels. The transform matrix $\boldsymbol{A}$ consists of eigenvectors of the covariance matrix $\boldsymbol{C}$. Although KLT is data-dependent, the transform matrices of most images are correlated. Therefore, to improve optimization efficiency, we randomly select 5000 images from the ImageNet1K validation dataset [44] to calculate the covariance matrices and their eigenvectors, after which we heuristically average the upper and lower 25% of the median interval and then orthogonalize the result to construct the estimated transform matrix $\hat{\boldsymbol{A}}$ and the mathematical expectation $\hat{\boldsymbol{\mu}}$. Then the loss function with color space transform is $\mathcal{L}\big(\Phi(f_{ift}(\mathbf{F})\hat{\boldsymbol{A}}+\hat{\boldsymbol{\mu}});d\big)$. Note that, after the color space transform, image processing regularization still needs to be performed, though we simplify it here for the clarity of notations.

V. Experiments

Considering network throughput capacity and representativeness, we mainly use ViT-B/16 and ResNet-50 pre-trained on ImageNet1K [44] in our qualitative experiments, which have become backbone networks in many vision tasks. We also introduce another Transformer-based network pre-trained on ImageNet1K, *i.e.*, SwinT. A significant distinction between ViTs and SwinTs is the shifted windowing scheme utilized in SwinTs, which limits self-attention computation to non-overlapping local windows while allowing for cross-window connection. As the SwinT stage deepens, the region influenced by a patch increases, as shown in Fig. 7. SwinTs are considered to combine the advantages of Transformers and CNNs, as they

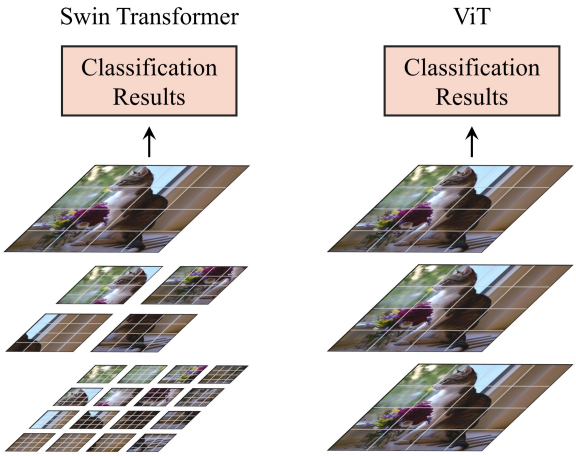


Fig. 7. One major distinction between SwinT and ViT architectures is that SwinT builds hierarchical representations by merging image patches (left), while ViT produces representations with the same resolution (right). The illustration was inspired by [6].

TABLE I
QUANTITATIVE COMPARISONS OF INVERSION OF THE FIRST LAYER. OUR PROPOSAL OUTPERFORMS OTHER APPROACHES IN ALL THREE METRICS.

Inversion Method	PSNR ↑	FFT2D ↓	LPIPS ↓
w/o frequency domain	12.712	0.043	0.334
w/o regularizations	17.123	0.017	0.236
w/o attention matching	18.971	0.019	0.276
InvertRepr	13.326	0.039	0.368
GradInversion	13.537	0.041	0.328
GradViT	15.214	0.035	0.292
ours	**20.054**	**0.015**	**0.196**

incorporate a hierarchical construction approach commonly used in CNNs. Our subsequent experiments also provide evidence that SwinTs exhibit characteristics of both architectures.

For quantitative evaluations in Sec. V-A and Sec. V-D, to obtain comprehensive results, we take a representative set of three types of networks–ViT-S/16, ViT-S/32, ViT-B/16, ViT-B/32, ViT-L/16, ResNet-34, ResNet-50, ResNet-101, ResNet-152, SwinT-S, SwinT-B, and SwinT-L. In our experiments, input image resolution is set at 224×224 pixels.

The optimization algorithm is Adamax [45], and we decay the learning rate exponentially according to loss change. For inversion, we activate the regularization only in the first $3/4$ epochs and then disable the regularization to eliminate possible blurring.

A. Visualization Quality Evaluation

In this section, we conduct an ablation study of the proposed techniques and compare our proposal with related studies. We mainly focus on inversion visualization, which is generated based on the original image and therefore allows for quantitative evaluation. The ablation study includes: inversion generated in the spatial domain (w/o frequency domain); direct optimization without image processing regularization (w/o regularizations); reconstruction of sub-encoder representation without attention matching (w/o attention matching). In addition, we compare our method with InvertRepr [8], GradInversion [21], and GradViT [22]. The fidelity regularization in GradInversion and the image prior in GradViT are not implemented in our experiments, because these regularization methods require an additional CNN to obtain statistics of normalization layers, which may introduce unknown errors when only ViT information is desired. When reproducing the methods being compared, we use the optimization strategy employed in the original papers, because we find that Adamax used in our proposal does not show superior performance in spatial domain optimization. The visualization comparisons (Fig. 8) in layer 1, 7, and 12 of ViT-B/16 show that our method produces the clearest and artifact-free inversion results.

As a quantitative evaluation, we adopt three commonly-used inversion quality metrics as in [22], *i.e.*, peak signal-to-noise ratio (PSNR), cosine similarity in the Fourier space (FFT2D), and learned perceptual image patch similarity (LPIPS) [46], to measure the similarity between inversions of the first layer and the original images. We select these three image quality metrics because they are capable of assessing the accuracy of feature representations reflected by the inversion method. Additionally, we choose to focus on the first layer as its inversion can be viewed as a complete reconstruction of the original images as discussed in [8]. The image quality assessment of the first layer allows us to evaluate the inverting performance of the inversion method, *i.e.*, the ability to accurately reconstruct representations. We randomly select 2000 images from the ImageNet1K validation dataset to generate feature representations and inversions at the first layer of all ViTs listed in Sec. V. Then, these three quality metrics are applied to evaluate the similarity between these inversions and the original images, as shown in Table I. The result shows that frequency domain optimization can significantly improve the inversion quality, and with our regularization and attention matching, the quality outperforms the previous benchmarks by a large margin.

Indeed, our method benefits from the inclusion of attention matching, which provides an additional advantage compared to previous methods that do not incorporate attention matching. When comparing the PSNR results of our method with and without attention matching, we observe an improvement of 1.083 on the metric. This indicates that attention matching plays an important role in enhancing the quality of the inversion results, leading to better image reconstructions with higher fidelity and accuracy.

We have not included a comparison of maximization visualization results in our study, as we find that the results optimized in the spatial domain tend to exhibit unordered and unstable patterns. We have presented several examples of such failure results in Fig. 9. Previous research has demonstrated that MLPs correspond to kernels of fast frequency falloff and is therefore difficult to produce high-frequency information in the spatial domain [31]. Considering the similarity between ViTs and MLPs, we speculate that ViTs may be more susceptible to the frequency falloff property than CNNs and therefore difficult to be processed in the spatial domain.

B. Feature Representation Reconstruction

In this section, we implement a comparison of feature representation reconstruction between ViT-B/16, ResNet-50,

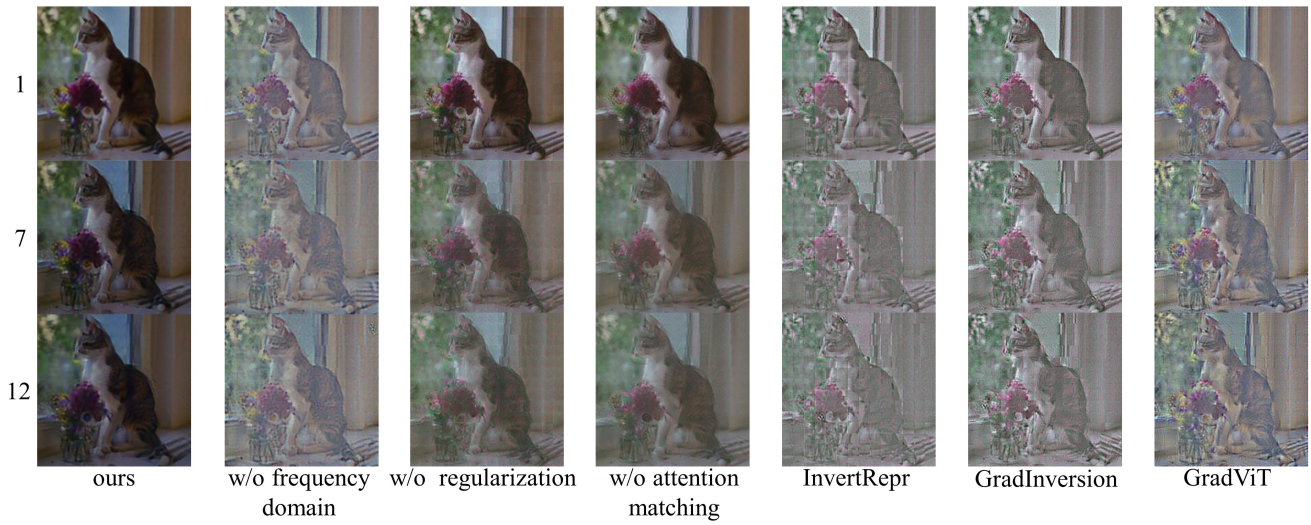


Fig. 8. Qualitative comparisons of inversions generated in layers 1, 7, and 12 (from top to bottom) of ViT-B/16. Other ablations and competing methods appear with artifacts or distortion of details. In contrast, our regularization method and attention matching demonstrate a significant impact on the sharpness of the inversion results and the removal of patch noise, as observed in these results. Furthermore, frequency-domain optimization appears to implicitly improve color consistency in the generated images.

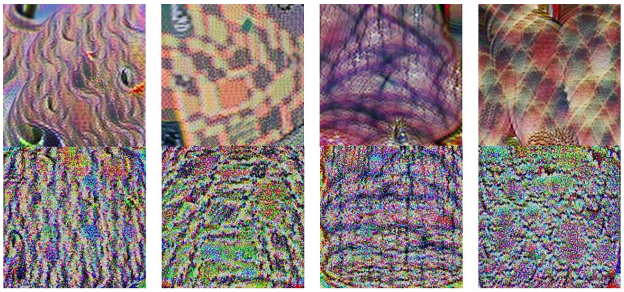


Fig. 9. Channel visualization examples in the 7_{th} layer of ViT-B/16. Top: optimization results in the frequency domain. Bottom: optimization results in the spatial domain. The results generated in the spatial domain show some unordered and unstable patterns.

and SwinT-B, as shown in Fig. 10. More qualitative experiments using other networks can be found in supplementary materials. The visualization comparison demonstrates that the consistency of representation reconstruction of the ViT is stronger, especially in higher layers where the ViT can always preserve most of the image details. On the other hand, the image detail information in the CNN and SwinT is gradually lost as the perceptual field increases in higher convolutional layers.

Lower layers of CNNs are considered to possess general-purpose feature extraction capability, *i.e.*, basic image features such as lines, colors, patterns, *etc*. Therefore, the low-layer inversions have a lot of image details, *e.g.*, the inversion of conv1 almost reconstructs the original image. In contrast, feature representations of higher layers of CNNs are considered to be class-discriminative. It is difficult to reproduce the details given these object-related feature representations, because a significant amount of low-level image features have been discarded. For example, the inversion of res5b of ResNet-50 shown in Fig. 10b looks like an abstract illusion of the objects in the original image. For the ViT, we find its inversions are very different from the ResNet. Even in higher layers, ViT inversions still contain object details, which indicates that the ViT feature propagation almost does not lose image information. For SwinT, we find that its results exhibit similarities with ResNet, *e.g.*, starting from the 7_{th} layer of stage 3, SwinT inversion results also show a substantial loss of image detail information.

We then apply the image similarity evaluation metrics (*i.e.*, PSNR and LPIPS) to all layers of these three networks by averaging the results generated using the selected 2000 images. As shown in Fig. 11, the inversion results of the ViT's different layers are highly consistent with the original images. While for ResNet-50, the ability to reconstruct original images degrades obviously in higher layers. SwinT further strengthens this tendency, and it is difficult to clearly reconstruct the original image from the 4_{th} layer of stage 3. We further adopt a factorization-accelerated centered kernel alignment (CKA) computation technique [47]–[49] to assess correlation scores among layers as another supporting evidence for the consistency of ViT representations. Similar correlation results can also be found in previous related work [1]. As shown in Fig. 12, correlation scores of feature representations between lower and higher layers of ResNet-50 are much lower than those of ViT-B/16. Notably, the results of SwinT in different stages are similar to the results of ResNet in different residual modules. These qualitative and quantitative results indicate that the original ViT feature propagation can better preserve image information.

C. Channel Feature Visualization

In this section, we generate channel representation maximization visualizations, and use attribution scores to detect channels that are relevant to the targeted output, to create a connection between maximization visualizations and the particular output. The choice of positive or negative visualization is determined by the sign of the sum of the channel

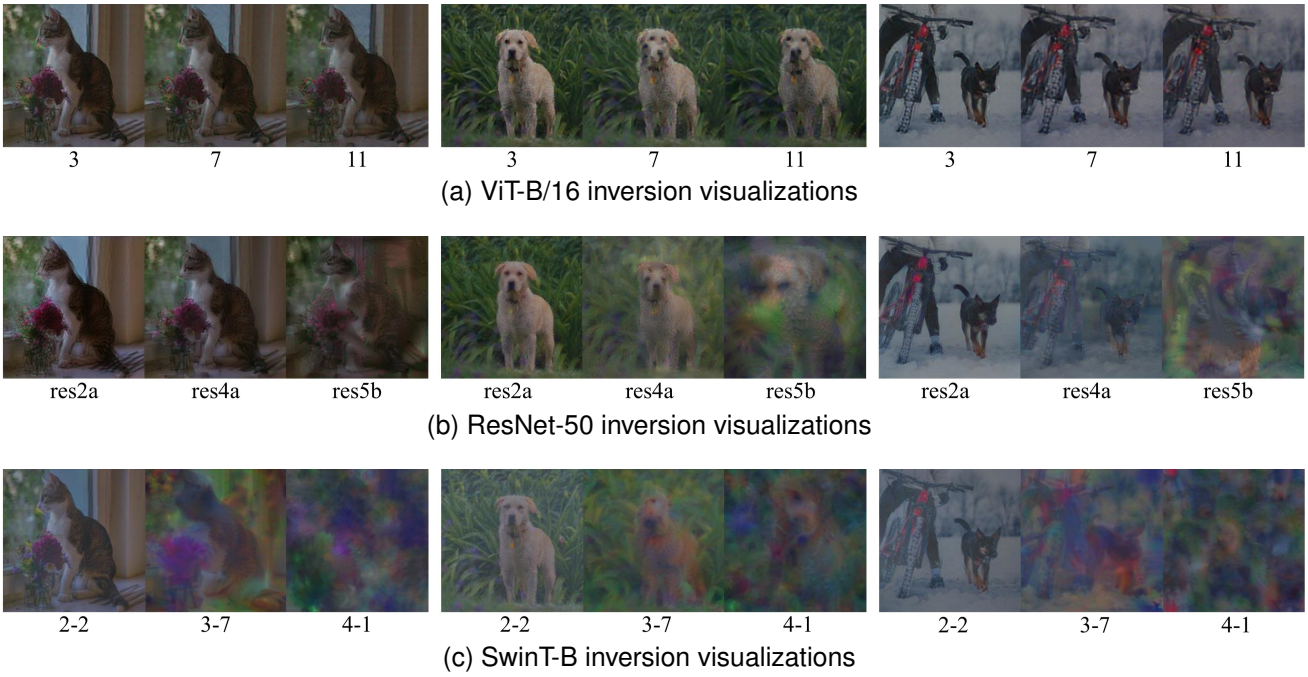


Fig. 10. Inversion comparison among ViT-B/16, ResNet-50, and SwinT-B. The visualizations of ViT consistently exhibit a high degree of consistency and effectively reconstruct a large number of image details even in the higher layers. This result suggests that ViTs are capable of capturing and maintaining meaningful image features throughout the network layers. In contrast, both CNN and SwinT visualizations show noticeable variability as the layer index increases, indicating that the feature representations in these architectures may undergo more significant changes with increasing depth. This observation highlights the potential advantage of ViTs in preserving and leveraging meaningful image features across different network layers.

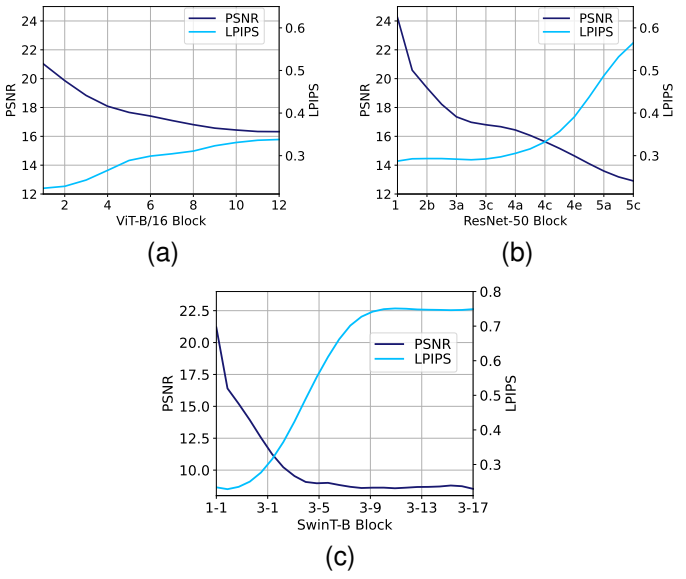


Fig. 11. Layer-wise inversion quality comparison. The horizontal axis represents the layer index of the network, the vertical axis on the left corresponds to the PSNR score, and the vertical axis on the right corresponds to the LPIPS score. Lower LPIPS scores indicate better quality, while higher PSNR scores indicate better quality. The results reveal that ViT inversions exhibit higher consistency among all layers, while both CNN and SwinT show lower consistency in high-layer inversion results.

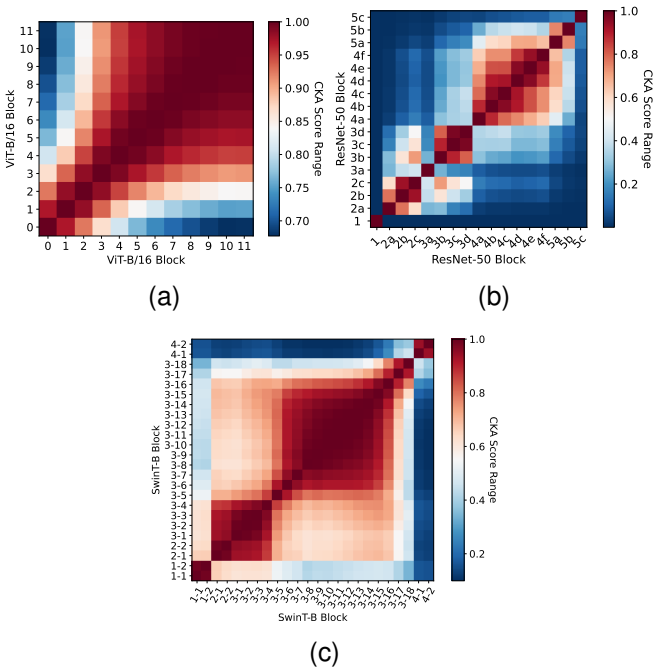
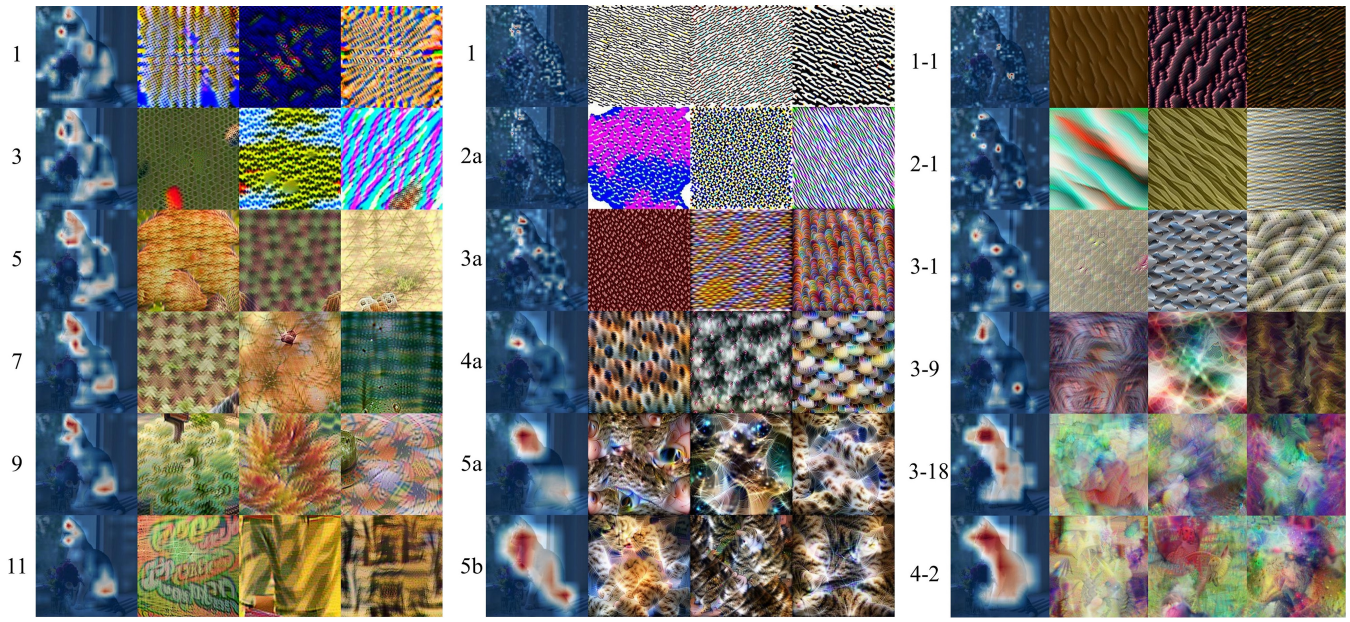


Fig. 12. CKA layer-wise similarities of ViT-B/16, ResNet-50, and SwinT-B. Both axes represent the layer index of the network, with red indicating the high similarity of feature representations and blue indicating low similarity. The ViT architecture exhibits highly consistent representations across all layers, while ResNet-50 shows noticeable differences in similarities between lower and higher layers. The SwinT architecture shows similar results to CNN in different stages (stages 1–4), but within the same stage, the SwinT results are more similar to ViT.

representation, which makes no significant difference for the discussion in this section.

The lower layers of a vision network are usually considered to learn general-purpose features such as lines, contours,



(a) ViT-B/16 channel visualizations (b) ResNet-50 channel visualizations (c) SwinT-B channel visualizations

Fig. 13. Representation maximization visualizations of the top three channels related to the "tiger cat" class. The heat maps are obtained by accumulating the three attribution values along channels. These images are best viewed on screen.

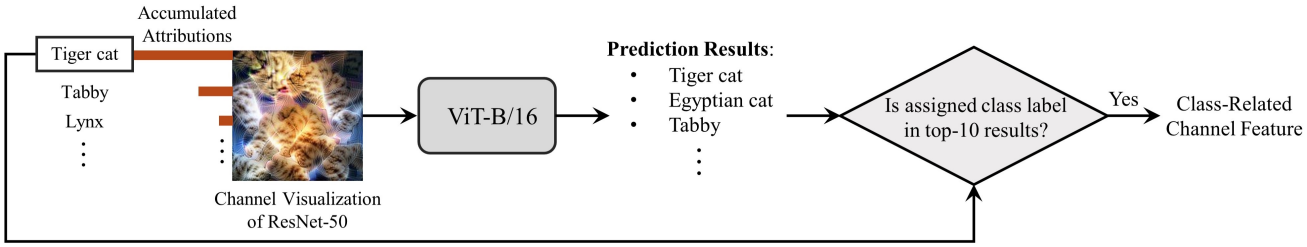


Fig. 14. A simple flowchart of the experiment for evaluating the class-related property. If a channel consistently contributes to a specific class label, we assign that class label to this channel. Then, if the network also recognizes this channel visualization, we consider the channel to be class-related.

textures, patterns, *etc*. While the higher layers are considered to be activated by specific objects, *e.g.*, the residual block5 of ResNet-50 can learn features associated with a specific output class. However, because of the consistency demonstrated by the ViT inversion experiment, we suppose that the high-layer channel features of ViT may be very different from ResNet. As an example, we calculate attributions with the "tiger cat" label using the original image in Fig. 4, and detect the contributing channel features of ViT-B/16, ResNet-50, and SwinT-B, respectively. Fig. 13 shows the visualizations of the top three channels that contribute the most to the target class in different layers. More channel feature visualization results using other networks can be found in supplementary materials. The heat maps are generated by summing these attributions along the channel dimension. We can find that the feature implications reflected by the ViT are still complex patterns even in higher layers, *e.g.*, the 11_{th} layer's results; however, the implications of residual block5 of ResNet appear to correspond to some objects such as cat's eyes, claws, and fur. Compared to ResNet, the high-layer features of ViT do not show a clear object-related property. Although SwinT's high-layer results are expected to be similar to ResNet theoretically, their results are highly abstract and may be challenging to effectively observe from a single visualization image.

To comprehensively assess the non-object/class-related property of ViT, we test whether the classification results are associated with class labels by swapping the channel visualizations of higher layers of ViT-B/16 and ResNet-50 and forwarding visualizations into each other's networks. First, we calculate attributions using 5000 randomly selected images from the ImageNet1K validation dataset, and assign the sum of the channel attribution values under a particular class to this channel as its class point, thus assigning a fixed class label for each channel based on its class points. For instance, to attribute one channel in a hidden layer, we accumulate the attribution scores obtained from all selected images under their respective labels. Because the attribution scores are class-targeted, we assign 1000-class points to this channel. Then, the class label corresponding to the highest point is associated with this channel. In other words, if a channel consistently contributes to a specific class label, we assign that class label to this channel. Subsequently, when the feature visualization

Fig. 15. Examples of the texture-bias dataset include original, grayscale, silhouette, edge, style-transferred, occluded, and shuffled images.

TABLE II
PREDICTION RESULTS OF VIT-B/16, RESNET-50, AND SWINT-B USING FIG. 15.

		Original	Grayscale	Silhouette	Edge	ST1	ST2	Occluded	Shuffled
ViT-B/16	Top-1 Class	tabby	tabby	hook	envelope	disk brake	wood rabbit	tabby	tabby
	True Label Ranking	1	1	441	157	39	3	1	1
ResNet-50	Top-1 Class	tabby	tabby	tabby	envelope	disk brake	ice bear	crossword puzzle	quilt
	True Label Ranking	1	1	1	91	377	23	110	7
SwinT-B	Top-1 Class	tabby	tabby	tabby	paper towel	mask	meerkat	tabby	tabby
	True Label Ranking	1	1	1	12	71	5	1	1

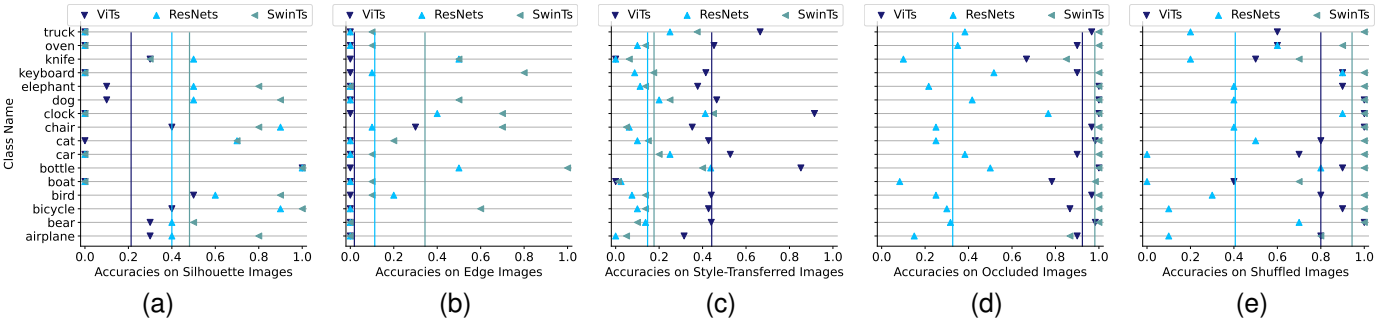


Fig. 16. Classification results for ViTs, ResNets, and SwinTs using five types of texture-bias data, *i.e.*, silhouette, edge, style-transferred, occluded, and shuffled images. The vertical axis represents different classes of images. The horizontal axis is the prediction accuracy of the networks. The triangular data points represent the prediction accuracy for each image class, and the vertical line of the same color as the data points represents the average score. In cases (a), (b), and (c), higher accuracy indicates that the network is less affected by texture bias, whereas in cases (d) and (e), higher accuracy indicates that the network may exploit local texture information.

of this channel is forwarded into the network, we expect it to activate the assigned class label. After confirming the class labels, the channel visualizations of residual block 5a and 5b of ResNet-50 are forwarded to the ViT to compute the probability that the visualization label is in the top-10 results. A simple flowchart of the experimental process is shown in Fig. 14. The result predicted by ViT-B/16 is 22%, which indicates that the high-layer visualization of ResNet-50 does correspond to some specific class labels. In contrast, we feed the channel visualizations of layers 10 and 11 of ViT-B/16 into the ResNet and find the prediction result is 1%, indicating that channel features of the ViT are not associated with specific class labels. This experiment demonstrates that ViTs may not necessarily learn object/class-related features even in higher layers, which are traditionally considered crucial for CNN-based classification. However, this observation also suggests that ViTs may possess stronger generalization capabilities, as they are still able to achieve high classification accuracy despite potentially not relying on explicit object/class-related features in their high-level representations.

D. Texture Bias Evaluation

Geirhos *et al.* [50] designed a texture-bias dataset and observed that CNNs pre-trained on ImageNet1K tend to make decisions based on textures more than shapes. They have 160 images of objects with white backgrounds and generated grayscale, silhouette, edge, and style transferred (ST) images based on 160 original images to form the texture bias dataset. In this section, we perform a similar experiment to evaluate the texture bias of ViTs and SwinTs. In the texture-bias dataset, the original and grayscale images can be accurately recognized by all networks, thus we do not discuss them in particular. In addition, we find that style transferred images generated using VGG networks [51] may not be appropriate for assessing ViTs, because experimental results show that CNN-based ResNets are more easily fooled by these style-transferred images than ViTs. Considering the permutation invariance of ViTs and SwinTs, we further add two new types of texture-bias data, *i.e.*, randomly occluded and shuffled images. If a network is able to recognize occluded or shuffled images, it is more likely to be biased towards texture. Randomly occluded or shuffled patch size in our experiment was randomly set under the uniform distribution at 8, 16, or 32. For randomly occluded

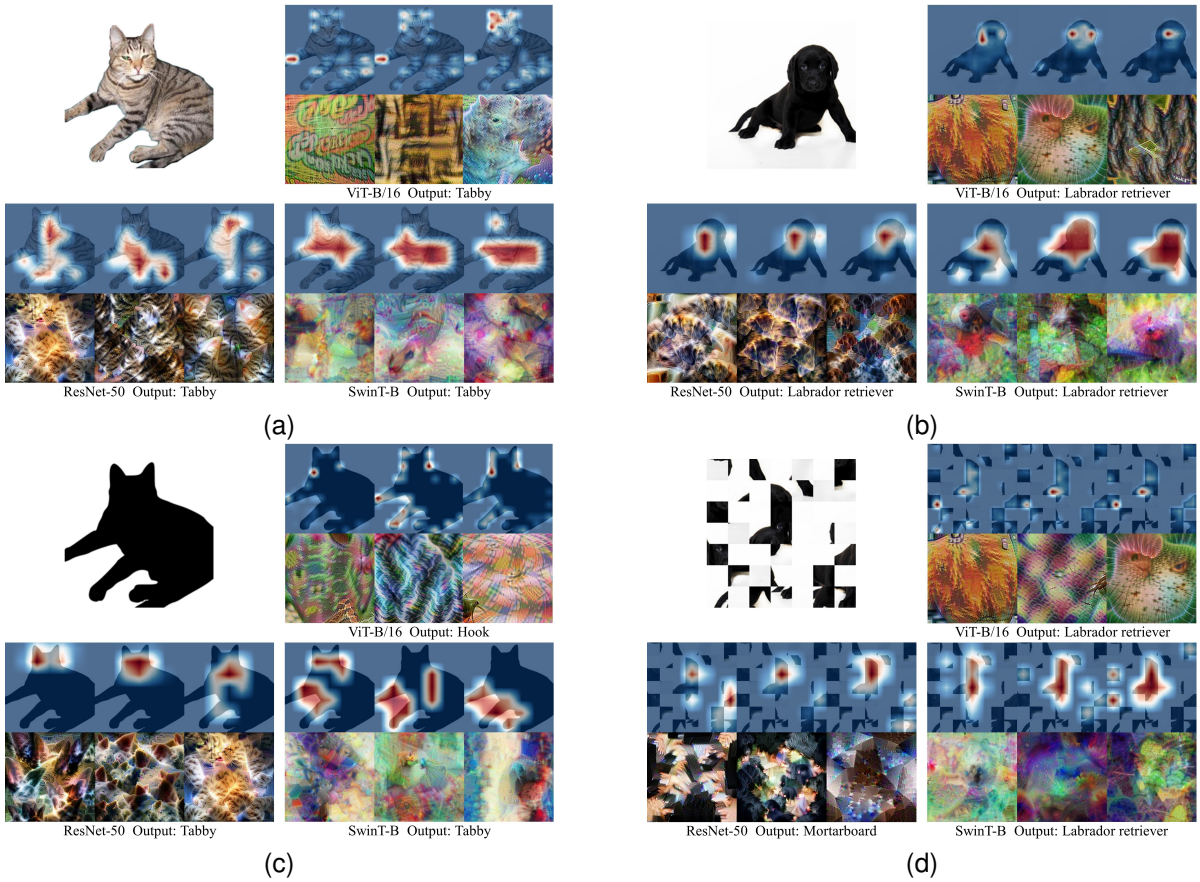


Fig. 17. Visualizations are generated using the image on the top left. Heat maps are generated using attribution values in the contributing channel which are also determined by attribution scores. Maximization visualizations below correspond to these contributing channels. These images are best viewed on screen.

images, the percentage of the masked region was randomly set in the range of 50%–70%. Some examples of the texture-bias images are shown in Fig. 15. Prediction results using these images (*i.e.*, the texture-bias images in Fig. 15) are shown in Table II, where "top-1 class" means the top-1 output label and "true label ranking" refers to the ranking of the "tabby" class (true label) in the prediction result. The original and grayscale cat images are accurately recognized by ViT-B/16, ResNet-50, and SwinT-B. The ResNet and SwinT recognize the silhouette cat image, but the ViT incorrectly recognizes it as a hook. For style-transferred images, the ViT performs better than ResNet and SwinTs. Interestingly, both ViT and SwinT are not disturbed by patch occlusion or permutation at all and successfully predict the tabby cat.

In terms of experimental implementation, differently from [50] where all images are evaluated together, we divide our test images into several parts to evaluate biases under different types of texture-bias data separately. In this experiment, all ViTs, CNNs, and SwinTs are used to evaluate texture bias. The classification results are shown in Fig. 16. We can find that ViTs are more biased towards recognizing textures than CNNs on edge and silhouette images. But SwinTs exhibits impressive performance on silhouette and edge images, implying its superior shape recognition ability. We also find that ViTs and SwinTs appear to be less affected by random occlusion or shuffle, indicating that they may more focus on the local textures of the object.

To make our texture-bias experiment comprehensive, we use the representation maximization visualization for further validation, as shown in Fig. 17. The visualization layer of the ViT is 11, while for ResNet it is res5b, and for SwinT it is layer 4-1. These heat maps are generated by resizing the attribution values of the channel to the image size. The representation maximization visualizations are selected according to channel contributions to the output class. Fig. 17(a) and 17(b) show two examples using original images in the texture-bias dataset. The top left image in Fig. 17(c) is the silhouette image of the original cat image above, and the visualizations show that ResNet-50 can recognize the cat by its shape in the silhouette. However, the top-1 classification result of the ViT is the "hook" class, while the "tabby" class ranks 441_{st}, indicating that the ViT cannot recognize the cat's shape. The visualization results also correspond to the lines formed by the silhouette of the cat and the background. This result is compatible with the observation in [3] where they described that using positional encoding results only in a 4% improvement in ImageNet 5-shot performance. The visualizations generated by SwinT are highly abstract and contain substantial target feature information. However, the visualization images do not clearly differentiate this information. For the Labrador retriever image, as shown in Fig. 17(d), the ViT and the SwinT can recognize the object even if all patches are randomly

shuffled, indicating that they pay more attention to local textures. Based on the experimental results, we hypothesize that shuffle invariance could be a significant challenge in generating visualizations with distinct object shapes in the higher layers of Transformers. Addressing this issue could be an intriguing direction for future research. On the other hand, the shuffled image is classified as a mortarboard by ResNet based on the black square in the middle region. The maximization visualizations also show colors and shapes very similar to those of a mortarboard. These experimental results show that all these networks are biased towards texture features, of which ViTs and SwinTs are more local-texture-biased.

VI. Conclusion

There is an increasing interest in comparing ViTs and CNNs to help researchers further understand these deep networks. In this work, we have utilized optimization visualization to intuitively understand representations of ViTs. To remove patch artifacts caused by the Transformer structure, we have introduced frequency domain optimization and image processing regularization into visualization generation. For inversion visualization, we leveraged attention matching to further improve visual quality. For maximization visualization, we redefined the loss function according to the Transformer structure and proposed to transform color space to alleviate the problem of inter-channel correlations due to inverse Fourier transform.

More important results emerged from visualization comparisons between these two types of networks. The comparison of representation reconstructions shows that ViTs are able to retain a large amount of image detail information up to very deep layers, a property that has not been observed in CNNs. We then used channel feature visualizations to demonstrate that ViT high-layer features do not exhibit object/class correlations as ResNets. Although the high-layer class discrimination is commonly considered necessary for CNN classification, ViTs achieve accurate classification with a different paradigm that the high-layer features are still dominated by complex patterns rather than class-discriminative features. Furthermore, we conducted a texture bias experiment with the support of visualizations and found that all tested networks are more biased towards textures than shapes, of which ViTs prefer local textures. We believe that optimization visualization can be a useful tool in understanding and comparing deep networks, particularly ones with different structures or mechanisms.

Our work still has some limitations that warrant further exploration in future research. Firstly, given the plethora of possibilities for indirect regularization, our proposal may not be deemed optimal, and identifying further ways to improve the regularization method remains a challenging task. Secondly, while our research has explored several applications of inversion and representation maximization, we acknowledge that there are numerous other potential applications for further research in these fields, *e.g.*, the exploration of the implicit conflict between the general architectural capabilities of ViTs and representation maximization. Both inversion and representation maximization hold significant promise for advancing our understanding of network feature representations. We believe that conducting further investigation into these methods can yield valuable insights into the intricate workings of neural networks. Finally, the impact of the data augmentation policy on texture bias remains unknown. There are various data augmentation techniques that could significantly influence network performance, we believe that systematically investigating and assessing their effects on texture bias would be a valuable undertaking. Such research could help disentangle the impact of network architectures and training approaches on texture bias. As research in this area continues, we plan to delve deeper into these questions.

References

[1] S. Bhojanapalli, A. Chakrabarti, D. Glasner, D. Li, T. Unterthiner, and A. Veit, "Understanding robustness of transformers for image classification," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2021, pp. 10 211–10 221.

[2] X. Cui, D. Wang, and Z. J. Wang, "Feature-flow interpretation of deep convolutional neural networks," *IEEE Trans. Multimedia*, vol. 22, no. 7, pp. 1847–1861, Oct. 2020.

[3] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby, "An image is worth 16x16 words: Transformers for image recognition at scale," in *Proc. Int. Conf. Learn. Representations*, 2021.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2016, pp. 770–778.

[5] M. Raghu, T. Unterthiner, S. Kornblith, C. Zhang, and A. Dosovitskiy, "Do vision transformers see like convolutional neural networks?" in *Proc. Adv. Neural Inf. Process. Syst.*, 2021, pp. 12 116–12 128.

[6] Z. Liu, Y. Lin, Y. Cao, H. Hu, Y. Wei, Z. Zhang, S. Lin, and B. Guo, "Swin Transformer: Hierarchical vision transformer using shifted windows," in *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021.

[7] H. Yin, P. Molchanov, J. M. Alvarez, Z. Li, A. Mallya, D. Hoiem, N. K. Jha, and J. Kautz, "Dreaming to distill: Data-free knowledge transfer via deepinversion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2020, pp. 8712–8721.

[8] A. Mahendran and A. Vedaldi, "Visualizing deep convolutional neural networks using natural pre-images," *Int. J. Comput. Vis.*, vol. 120, no. 3, pp. 233–255, Mar. 2016.

[9] D. Zhou, Z. Yu, E. Xie, C. Xiao, A. Anandkumar, J. Feng, and J. M. Alvarez, "Understanding the robustness in vision transformers," in *Proceedings of the 39th International Conference on Machine Learning*, vol. 162. PMLR, 2022, pp. 27 378–27 394. [Online]. Available: https://proceedings.mlr.press/v162/zhou22m.html

[10] R. J. Williams, "Inverting a connectionist network mapping by back-propagation of error," in *8th Annual Conf. Cognitive Sci. Soc.*, 1986.

[11] S. Lee and R. M. Kil, "Inverse mapping of continuous functions using local and global information," *IEEE Trans. Neural Netw.*, vol. 5, no. 3, pp. 409–423, 1994.

[12] B.-L. Lu, H. Kita, and Y. Nishikawa, "Inverting feedforward neural networks using linear and nonlinear programming," *IEEE Trans. Neural Netw.*, vol. 10, no. 6, pp. 1271–1290, 1999.

[13] G. Joshi, R. Natsuaki, and A. Hirose, "Neural network model for multi-sensor fusion and inverse mapping dynamics for the analysis of significant factors," in *IGARSS 2022 - 2022 IEEE International Geoscience and Remote Sensing Symposium*, 2022, pp. 473–476.

[14] L. Xie and Z.-Q. Liu, "Realistic mouth-synching for speech-driven talking face using articulatory modelling," *IEEE Trans. Multimedia*, vol. 9, no. 3, pp. 500–510, 2007.

[15] G. Joshi, R. Natsuaki, and A. Hirose, "Neural network fusion processing and inverse mapping to combine multisensor satellite data and analyze the prominent features," *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, vol. 16, pp. 2819–2840, 2023.

[16] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv:1312.6034*, 2013.

[17] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.*, vol. 8689, 2014, pp. 818–833.

[18] É. Protas, J. D. Bratti, J. F. de Oliveira Gaya, P. D. Jr., and S. S. C. Botelho, "Visualization methods for image transformation convolutional neural networks," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 7, pp. 2231–2243, 2019.

[19] J. Huang, J. Liao, and S. Kwong, "Unsupervised image-to-image translation via pre-trained stylegan2 network," *IEEE Trans. Multimedia*, vol. 24, pp. 1435–1448, 2022.

[20] A. Mahendran and A. Vedaldi, "Understanding deep image representations by inverting them," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2015, pp. 5188–5196.

[21] H. Yin, A. Mallya, A. Vahdat, J. M. Alvarez, J. Kautz, and P. Molchanov, "See through gradients: Image batch recovery via GradInversion," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 16 337–16 346.

[22] A. Hatamizadeh, H. Yin, H. Roth, W. Li, J. Kautz, D. Xu, and P. Molchanov, "GradViT: Gradient inversion of vision transformers," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2022, pp. 10 021–10 030.

[23] A. Nguyen, J. Yosinski, and J. Clune, "Understanding neural networks via feature visualization: A survey," *arXiv:1904.08939*, 2019.

[24] S. Santurkar, A. Ilyas, D. Tsipras, L. Engstrom, B. Tran, and A. Madry, "Image synthesis with a single (robust) classifier," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 1260–1271.

[25] F. Hohman, H. Park, C. Robinson, and D. H. P. Chau, "Summit: Scaling deep learning interpretability by visualizing activation and attribution summarizations," *IEEE Trans. Vis. Comput. Graphics*, vol. 26, no. 1, pp. 1096–1106, 2020.

[26] C. Olah, A. Mordvintsev, and L. Schubert, "Feature visualization," *Distill*, 2017, https://distill.pub/2017/feature-visualization.

[27] C. Olah, N. Cammarata, L. Schubert, G. Goh, M. Petrov, and S. Carter, "An overview of early vision in inceptionv1," *Distill*, 2020, https://distill.pub/2020/circuits/early-vision.

[28] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2019, pp. 2950–2958.

[29] R. Shi, T. Li, and Y. Yamaguchi, "Group visualization of class-discriminative features," *Neural Netw.*, vol. 129, pp. 75–90, 2020.

[30] S. Singla, B. Nushi, S. Shah, E. Kamar, and E. Horvitz, "Understanding failures of deep networks via robust feature extraction," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 12 853–12 862.

[31] M. Tancik, P. P. Srinivasan, B. Mildenhall, S. Fridovich-Keil, N. Raghavan, U. Singhal, R. Ramamoorthi, J. T. Barron, and R. Ng, "Fourier features let networks learn high frequency functions in low dimensional domains," in *Proc. Adv. Neural Inf. Process. Syst.*, 2020.

[32] M. Tesfaldet, X. Snelgrove, and D. Vázquez, "Fourier-CPPNs for image synthesis," in *Proc. IEEE Int. Conf. Comput. Vis. Workshop*. IEEE, 2019, pp. 3173–3176.

[33] Y. Wang, H. Su, B. Zhang, and X. Hu, "Learning reliable visual saliency for model explanations," *IEEE Trans. Multimedia*, vol. 22, no. 7, pp. 1796–1807, 2020.

[34] H. Chefer, S. Gur, and L. Wolf, "Transformer interpretability beyond attention visualization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2021, pp. 782–791.

[35] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," in *Proceedings of the International Conference on Learning Representations*, 2014. [Online]. Available: http://dblp.uni-trier.de/db/conf/iclr/iclr2014w.html#SimonyanVZ13

[36] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, and W. Samek, "On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation," *PloS one*, vol. 10, no. 7, 2015. [Online]. Available: https://doi.org/10.1371/journal.pone.0130140

[37] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/file/8a20a8621978632d76c43dfd28b67767-Paper.pdf

[38] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *Proceedings of the 34th International Conference on Machine Learning*, 2017, p. 3319–3328.

[39] H. Chen, S. M. Lundberg, and S.-I. Lee, "Explaining a series of models by propagating shapley values," *Nat. Commun.*, vol. 13, p. 4512, 2022.

[40] R. Shi, T. Li, and Y. Yamaguchi, "Output-targeted baseline for neuron attribution calculation," *Image Vis. Comput.*, vol. 124, p. 104516, 2022.

[41] J. Ren, Z. Zhou, Q. Chen, and Q. Zhang, "Can we faithfully represent absence states to compute shapley values on a DNN?" in *The Eleventh International Conference on Learning Representations*, 2023. [Online]. Available: https://openreview.net/forum?id=YV8tP7bW6Kt

[42] F. Eitel, A. Melkonyan, and K. Ritter, "Feature visualization for convolutional neural network models trained on neuroimaging data," *arXiv:2203.13120*, 2022.

[43] X. Tang, "Texture information in run-length matrices," *IEEE Trans. Image Process.*, vol. 7, no. 11, pp. 1602–1609, 1998.

[44] J. Deng, W. Dong, R. Socher, L. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2009, pp. 248–255.

[45] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv:1412.6980*, 2014.

[46] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 2018, pp. 586–595.

[47] C. Cortes, M. Mohri, and A. Rostamizadeh, "Algorithms for learning kernels based on centered alignment," *J. Mach. Learn. Res.*, vol. 13, pp. 795–828, 2012.

[48] S. Kornblith, M. Norouzi, H. Lee, and G. E. Hinton, "Similarity of neural network representations revisited," in *Proc. Int. Conf. Mach. Learn.*, vol. 97, 2019, pp. 3519–3529.

[49] T. Li, R. Shi, and T. Kanai, "Detail-aware deep clothing animations infused with multi-source attributes," *Computer Graphics Forum*, vol. 42, no. 1, pp. 231–244, 2023.

[50] R. Geirhos, P. Rubisch, C. Michaelis, M. Bethge, F. A. Wichmann, and W. Brendel, "Imagenet-trained CNNs are biased towards texture; increasing shape bias improves accuracy and robustness," in *Proc. Int. Conf. Learn. Representations*, 2019.

[51] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Representations*, 2015.

Supplementary Materials of Visualization Comparison of Vision Transformers and Convolutional Neural Networks

I. More Experimental Results

A. Inversions using Different Images

The images in Fig. S1 are downloaded from the URL: https://unsplash.com/, licensed under Creative Commons Attributions CC-BY 4.0. We report more inversion results on ViT-B/16 and ResNet-50 using these images, as shown in Fig. S2. The highly consistent ViT inversions indicate that even in high layers, the feature representation of the ViT may correspond to image details like textures and shapes. ResNet, on the other hand, discards most of the image information in high layers which may not be necessary for classification in ResNet.

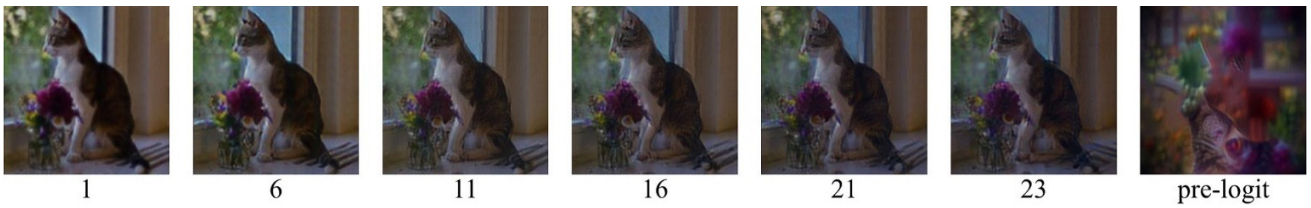
Fig. S1. Original images for generating visualizations.

Fig. S2. Inversions generated with different images. Top row of each pair is generated using ViT-B/16; Bottom row of each pair is generated using ResNet-50.

B. Inversions using Other Networks

We report inversion experiments on ViT-S/16, ViT-L/16, and ResNet-152, as shown in Fig. S3. Both ViTs achieve highly consistent representation reconstructions of the input image. However, ResNet-152 has difficulty in achieving such clear reconstruction in high layers. We also find that the optimization in the last few layers of the very deep ResNet-152 seems to be intractable and cannot steadily produce a recognizable result.

(a) ViT-S/16.



(b) ViT-L/16.

(c) ResNet-152.

Fig. S3. Inversions on different networks.

We further report inversion experiments on ViT-B/16, ResNet-50, and SwinT-B pre-trained on ImageNet21K, as shown in Fig. S4. The ImageNet21K results are similar to their ImageNet1K counterpoints. For instance, ViT continues to exhibit consistent inversion results, whereas ResNet and SwinT progressively lose image details.

(a) ViT-B/16-21K.

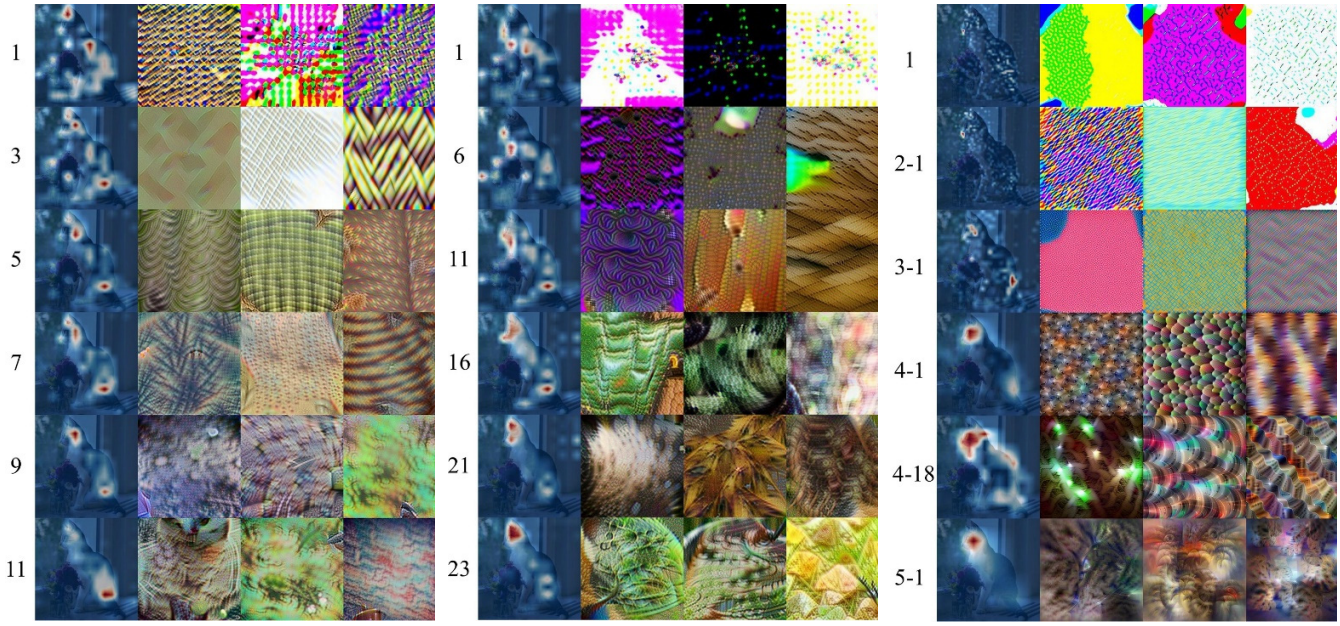
(b) ResNet-50-21K.

(c) SwinT-B-21K.

Fig. S4. Inversions on the networks pre-trained on ImageNet21K.

C. Maximization Visualizations using Other Networks

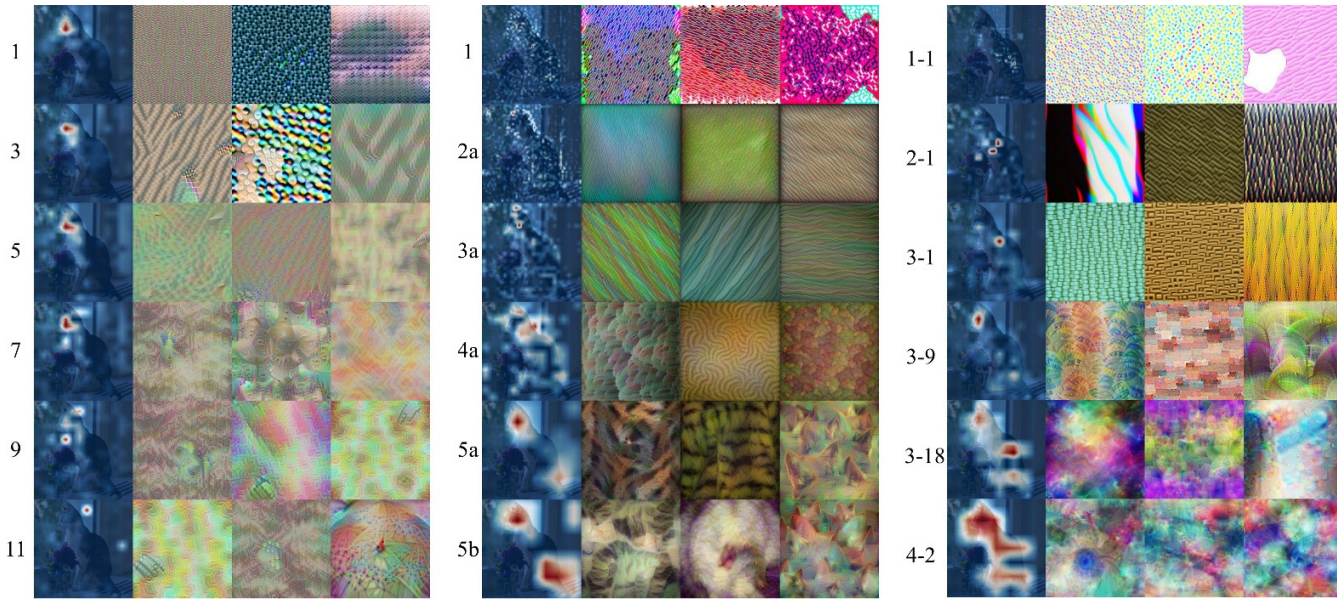
We report maximization visualization experiments on ViT-S/16, ViT-L/16, and ResNet-152, as shown in Fig. S5. The high layers of ViTs seem similar to the residual block 4 of the ResNet. The visualizations of ViTs are not explicitly related to a specific object/class, but the visualizations of block 5 in ResNet show clearly cat-related details.



(a) ViT-S/16. (b) ViT-L/16. (c) ResNet-152.

Fig. S5. Representation maximization results of the top three channels related to the "tiger cat" class on different networks. These images are best viewed on screen.

We also report maximization visualization experiments on ViT-B/16, ResNet-50, and SwinT-B pre-trained on ImageNet21K, as shown in Fig. S6. Compared with the networks pre-trained on ImageNet1K, the networks pre-trained on ImageNet21K seem to show more abstraction, which may imply stronger feature extraction ability. But we still cannot observe distinct shape semantics in ViT visualizations.



(a) ViT-B/16-21K. (b) ResNet-50-21K. (c) SwinT-B-21K.

Fig. S6. Representation maximization results of the top three channels related to the "tiger cat" class on the networks pre-trained on ImageNet21K. These images are best viewed on screen.

D. Influence of Regularization Selections

In addition to the indirect regularization methods we devised, commonly used direct regularization techniques, such as total variation and α-norm, lack the ability to maintain patch consistency. Another intuitive option, similar to morphological operations, is image smoothing filters like the mean filter. Compared to image smoothing, one advantage of morphological operations is that

they offer better preservation of contours while blurring internal textures. In addition, their flexible structuring element selection may contribute to further extension.

We also conducted experiments with classical regularization methods, such as a combination of total variation and α-norm, and with mean filter regularization (replacing morphological operations with mean filter). As shown in Fig. S7, the regularization methods we selected demonstrate slightly better visual performance. The quantitative results are shown in Table S-I. Both experimental results show that our proposal outperforms the other variations. But we also believe that exploring improved filters or morphological approaches could yield further improvements in results. However, due to the possibility of regularization, determining the optimal combination of regularization methods remains challenging. Further exploration in this direction could be an interesting avenue for future research.

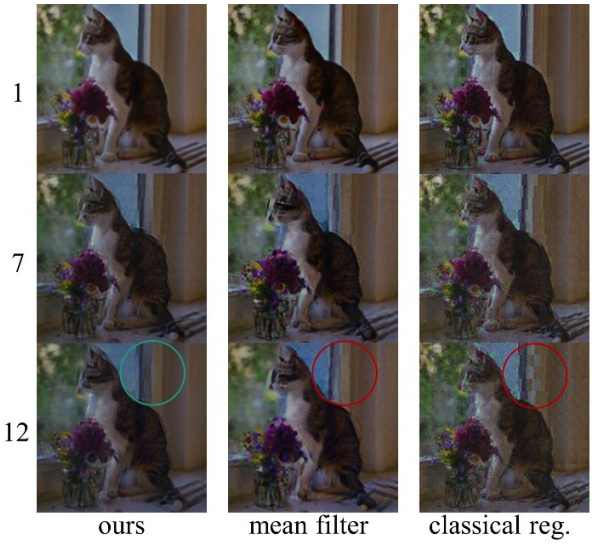


Fig. S7. An effect of different regularization methods.

Table S-I

Quantitative comparisons of inversion generated with different regularization techniques. Our proposal outperforms other alternatives in all three metrics.

Regularization Selection	PSNR ↑	FFT2D ↓	LPIPS ↓
mean filter	19.015	0.018	0.226
classical reg.	18.731	0.019	0.235
ours	**20.054**	**0.015**	**0.196**