Token Transformation Matters: Towards Faithful Post-hoc Explanation for Vision Transformer

Junyi Wu[1], **Bin Duan**[1], **Weitai Kang**[1], **Hao Tang**[2], **Yan Yan**[1,†]
[1]Department of Computer Science, Illinois Institute of Technology, USA
[2]Robotics Institute, Carnegie Mellon University, USA
{jwu125, bduan2, wkang11}@hawk.iit.edu, haotang2@cmu.edu, yyan34@iit.edu

Abstract

While Transformers have rapidly gained popularity in various computer vision applications, post-hoc explanations of their internal mechanisms remain largely unexplored. Vision Transformers extract visual information by representing image regions as transformed tokens and integrating them via attention weights. However, existing post-hoc explanation methods merely consider these attention weights, neglecting crucial information from the transformed tokens, which fails to accurately illustrate the rationales behind the models' predictions. To incorporate the influence of token transformation into interpretation, we propose ***TokenTM****, a novel post-hoc explanation method that utilizes our introduced measurement of token transformation effects. Specifically, we quantify token transformation effects by measuring changes in token lengths and correlations in their directions pre- and post-transformation. Moreover, we develop initialization and aggregation rules to integrate both attention weights and token transformation effects across all layers, capturing holistic token contributions throughout the model. Experimental results on segmentation and perturbation tests demonstrate the superiority of our proposed TokenTM compared to state-of-the-art Vision Transformer explanation methods.*

1. Introduction

The application of Transformer models has resulted in superior performance in computer vision [11, 17, 24, 30, 46, 47], paving the way for new breakthroughs in various tasks. However, the inherent complexity of Vision Transformers often renders them black-box models, making understanding the rationales behind their predictions a challenge. Such a lack of transparency undermines the trustworthiness of decision-making processes [2, 16, 26, 38, 49]. Therefore, it is required to interpret Transformer networks. In the visual domain, post-hoc explanation sheds light on the rationale behind a model's predictions using a heatmap over the input pixel space. This approach is both effective and efficient in providing interpretations that are easily understood by humans [13, 32, 45, 52].

There exist two types of post-hoc explanation methods, general traditional explanations [37, 45] and Transformer-specific attention-based explanations [1, 12]. Although traditional explanations excel in visualizing important pixels for predictions of MLPs and CNNs, their effectiveness markedly decreases when applied to Vision Transformers, due to the fundamental differences in model architectures. Therefore, attention-based explanations design new paradigms specific to Transformers, where attention weights play a dominant role [1, 12, 26, 35]. Integrating attention information, these explanation methods offer a promising approach toward Transformer interpretability.

Recent advancements in attention-based explanation techniques have outperformed traditional methods on Vision Transformers. However, the inherent complexity of these models, particularly due to their key components: Multi-Head Self-Attention (MHSA) and Feed Forward Network (FFN) [17], continues to pose unique challenges in terms of explainability. As elucidated in Section 3, we represent the outputs of both MHSA and FFN as weighted sums of transformed tokens, with each token scaled by an attention weight. Existing attention-based methods simply consider the scaling weights indicated by attention maps as the corresponding tokens' contributions, overlooking the impacts of token transformations. As shown in Figure 1, attention weights alone misrepresent the contributions from foreground objects or background regions, while transformation information offers a necessary counterbalance. For instance, even if certain background regions are scaled by high attention weights, their actual contribution can be diminished if they are transformed into smaller or divergent tokens. Conversely, a foreground object, despite receiving minimal attention weights, can play a pivotal role due to significant transformation within the model. Given the in-

[†]Corresponding author

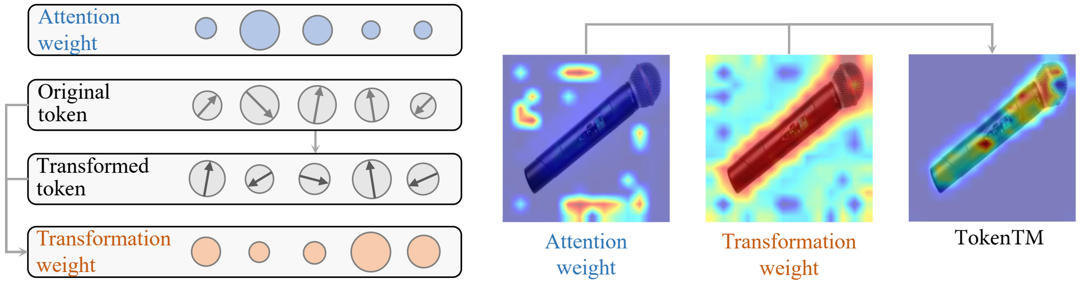

Figure 1. Visualization of attention and transformation weights, and the result of our TokenTM that integrates both of them. Circle sizes signify weight magnitudes or token lengths, and arrows indicate directions. Transformation weights are derived by our proposed measurement, which evaluates the transformation effects by gauging changes in length and direction. Both weights are visualized by heatmaps. Solely using attention weights often fails to localize foreground objects and inaccurately highlights noisy backgrounds as rationales. In contrast, leveraging additional information from transformation, our method produces object-centric post-hoc interpretations.

tertwined dynamics of attention weights and token transformations, there is an imperative need for a comprehensive explanation method that cohesively addresses both factors.

In this paper, we propose **TokenTM**, a novel post-hoc explanation method for Vision Transformers using token transformation measurement. We first revisit Vision Transformer layers from a generic perspective, which conceives both MHSA and FFN's outputs as weighted linear combinations of original and transformed tokens. Intuitively, tokens with increased magnitude and aligned orientations will significantly influence the linear combination outcomes. Therefore, to quantify the effects of transformations, we introduce a measurement that focuses on two fundamental token attributes: length and direction. Using the proposed measure, we integrate transformation effects, quantified by transformation weights, with the attention information, ensuring a faithful assessment of token contributions. Moreover, Vision Transformers consist of sequentially stacked layers, each performing vital token transformations and globally mixing different tokens, a process termed contextualization [47]. Recognizing the accumulative nature of these mechanisms, a single-layer analysis remains insufficient [1, 10, 26]. To holistically evaluate all layers, our solution encompasses an aggregation framework. Employing rigorous initialization and update rules, our framework yields a comprehensive contribution map, which reveals token contributions over the entire model. Experiments on segmentation and perturbation tests show that our TokenTM outperforms state-of-the-art methods.

In summary, our contributions are as follows: **(i)** We explore post-hoc interpretation for Vision Transformer and identify a primary issue: the lack of comprehensive consideration for token transformations and attention weights, which can result in misleading rationales of the model's prediction. **(ii)** We introduce TokenTM, a novel post-hoc explanation method employing token transformation measurement. This measurement regards changes in length and directional correlation, which faithfully assesses the impact of transformations on token contributions. **(iii)** Our approach establishes an aggregation framework that integrates both attention and token transformation effects across multiple layers, thereby capturing the cumulative nature of Vision Transformers. **(iv)** Using the proposed measurement and framework, TokenTM demonstrates superior performance in comparison to existing state-of-the-art methods.

2. Related Work

General Traditional Explanations. Traditional post-hoc explanation methods mainly fall into two groups: gradient-based and attribution-based. Examples of gradient-based methods are Gradient*Input [40], SmoothGrad [41], Deconvolutional Network [50], Full Grad [43], Integrated Gradients [45], and Grad-CAM [37]. These methods produce saliency maps using the gradient. On the other hand, attribution-based methods propagate classification scores backward to the input [5, 20, 22, 25, 31, 39], based on Deep Taylor Decomposition [33]. There are other approaches beyond these two types, such as saliency-based [15, 32, 51], Shapley additive explanation (SHAP) [31], and perturbation-based methods [19]. Although initially designed for MLPs and CNNs, some traditional methods have been adapted for Transformers in recent works [3, 13]. However, without attention weights, these methods still yield suboptimal performance on Vision Transformers.

Transformer-specific Attention-based Explanations. A growing line of work in interpretability develops new paradigms specifically for Transformers. Attention maps are widely used in this direction, as they are intrinsic distributions of scaling weights over tokens. Representative methods include Raw Attention [49], which regards attention as an explanation, Rollout [1], which linearly accumulates attention maps, GAE [12], a framework for variants of Transformers, ATTCAT [35], which formulates Attentive Class Activation Tokens to estimate their importance, and IIA [7] and DIX [8], which propose path integration using attention. However, these approaches overlook the relative effects of transformations and fail to faithfully aggregate token contributions across all modules, hindering reliable post-hoc interpretations for Vision Transformers.

3. Analysis

Vision Transformer [17] sequentially stacks n_L layers, each containing an MHSA or an FFN. We first reinterpret these layers generically and then analyze the problem in Vision Transformer explanations.

3.1. Revisiting Transformer Layers

From a general viewpoint, MHSA and FFN both process weighted linear combinations of original and transformed tokens. Starting with a reinterpretation of MHSA, we illustrate this shared principle, then we show how FFN emerges as a particular case within the unified formulation.

In MHSA, every token of the input sequence attends to all others by projecting the embeddings to a query, key, and value. Formally, let $\mathbf{E} \in \mathbb{R}^{n\times d}$ be the embeddings matrix (a sequence of tokens), where n is the number of tokens, and d is the dimensionality of embedding space. The projections can be expressed as:

$$\mathbf{Q} = \mathbf{E}\mathbf{W}^{\mathbf{Q}}, \quad \mathbf{K} = \mathbf{E}\mathbf{W}^{\mathbf{K}}, \quad \mathbf{V} = \mathbf{E}\mathbf{W}^{\mathbf{V}}, \tag{1}$$

where $\mathbf{W}^{\mathbf{Q}} \in \mathbb{R}^{d\times d_Q}$, $\mathbf{W}^{\mathbf{K}} \in \mathbb{R}^{d\times d_K}$, and $\mathbf{W}^{\mathbf{V}} \in \mathbb{R}^{d\times d_V}$ are parameter matrices. Subsequently, the attention map $\mathbf{A}$ is computed by:

$$\mathbf{A} = \text{Softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_Q}}\right) \in \mathbb{R}^{n\times n}. \tag{2}$$

Then, contextualization is performed on value $\mathbf{V}$ using attention map $\mathbf{A}$, and another linear transformation further projects the resulting embeddings back to the space $\mathbb{R}^d$:

$$(\mathbf{A}\mathbf{V})\mathbf{W}^{\mathbf{H}} \in \mathbb{R}^{n\times d}, \tag{3}$$

where $\mathbf{W}^{\mathbf{H}} \in \mathbb{R}^{d_V\times d}$. To obtain the final output, MHSA integrates the results from multiple heads and incorporates them into original embeddings $\mathbf{E}$ from the previous layer by skip-connection [23]. The output of MHSA is given by:

$$\text{MHSA}(\mathbf{E}) = \mathbf{E} + \sum_{h=1}^{n_H} (\mathbf{A}\mathbf{V})\mathbf{W}^{\mathbf{H}}, \tag{4}$$

where n_H is the number of heads. We omit the subscript h for simplicity. Given the associative property of matrix multiplication, we can regard the combination of value projection and multi-head integration as a simple yet equivalent token transformation featured by $\widetilde{\mathbf{W}} = \mathbf{W}^{\mathbf{V}}\mathbf{W}^{\mathbf{H}}$:

$$(\mathbf{A}\mathbf{V})\mathbf{W}^{\mathbf{H}} = \mathbf{A}\mathbf{E}(\mathbf{W}^{\mathbf{V}}\mathbf{W}^{\mathbf{H}}) = \mathbf{A}(\mathbf{E}\widetilde{\mathbf{W}}). \tag{5}$$

We then reformulate the MHSA using the definition of transformed embeddings $\widetilde{\mathbf{E}} = \mathbf{E}\widetilde{\mathbf{W}}$:

$$\text{MHSA}(\mathbf{E}) = \mathbf{E} + \sum_{h=1}^{n_H} \mathbf{A}\widetilde{\mathbf{E}}. \tag{6}$$

From a vector-level view, each token of the MHSA's output is a weighted sum of original and transformed tokens. With this insight, the FFN embodies a basic form of 'contextualization', where the number of heads n_H=1 and the attention map $\mathbf{A}$=$\mathbf{I}$ (identity matrix), as contrasted with Eq. (6). Utilizing the concept of transformed embeddings $\widetilde{\mathbf{E}}$, the FFN can be expressed as:

$$\text{FFN}(\mathbf{E}) = \mathbf{E} + \mathbf{I}\widetilde{\mathbf{E}}, \tag{7}$$

where $\widetilde{\mathbf{E}}$ consists of tokens that have been transformed within the FFN.

3.2. Problem in Explaining Vision Transformers

As depicted in Eq. (6) and Eq. (7), Vision Transformer layers are generically expressed as weighted sums of original and transformed tokens, where each token is multiplied by a scaling weight. Existing attention-based explanations typically consider these scaling weights as the corresponding tokens' contributions, which overlooks the influences imparted by the tokens themselves. As illustrated in Figure 1, relying solely on attention weights can misrepresent the contributions from various image patches. This issue necessitates a comprehensive method to faithfully interpret the inner mechanism of Transformer layers and capture the true rationales behind the predictions.

4. The Proposed Method

In this section, we propose TokenTM. We first introduce the background of attention-based explanations for Vision Transformers. Then, we develop a token transformation measurement, which evaluates the effects of transformed tokens. Finally, we design an aggregation framework to accumulate both attention and transformation information across all layers.

4.1. Attention-based Explanations

Attention-based explanation methods [1, 12, 13] measure the contribution of each token using the attention information, as expressed by:

$$\mathbf{C} = \mathbf{O} + \mathbb{E}_h\left[(\nabla_{\mathbf{A}} p(c))^+ \odot \mathbf{T}\right], \mathbf{O} = \mathbf{I}, \mathbf{T} = \mathbf{A}. \tag{8}$$

Here, $\odot$ is the Hadamard product. $\mathbf{C} \in \mathbb{R}^{n\times n}$ denotes the contribution map, where C_{ij} represents the influence of the j-th input token on the i-th output token. Matrices $\mathbf{O} \in \mathbb{R}^{n\times n}$ and $\mathbf{T} \in \mathbb{R}^{n\times n}$ reflect the contributions

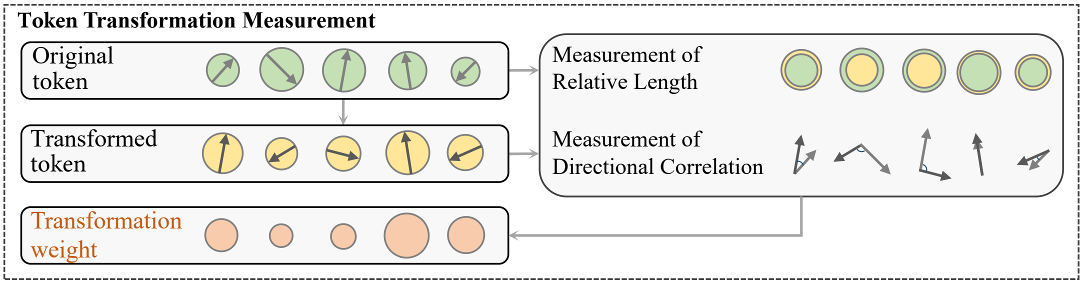
Token Transformation Measurement
Original token
Transformed token
Transformation weight
Measurement of Relative Length
Measurement of Directional Correlation

Figure 2. Illustration of our token transformation measurement. We depict original and transformed tokens with circles and arrows. Circle sizes reflect lengths, and arrows denote directions. The effects of token transformation are reflected by the changes in length and direction. Our method considers both properties to evaluate these effects, resulting in the corresponding transformation weights.

from the original and transformed tokens, respectively. Previous explanation methods quantify $\mathbf{O}$ and $\mathbf{T}$ simply using scaling weights. Specifically, $\mathbf{I}$ is an identity matrix representing the self-contributions of original tokens, and $\mathbf{A}$ is the attention weights for scaling transformed tokens. Moreover, $\nabla_{\mathbf{A}} p(c) = \frac{\partial p(c)}{\partial \mathbf{A}}$ is the partial derivative of attention map *w.r.t.* the predicted probability for class c. $\mathbb{E}_h$ is the mean over multiple heads. Note that it averages across heads using the gradient $\nabla_{\mathbf{A}} p(c)$ to become class-specific, and it removes the negative contributions before averaging to avoid distortion [6, 12, 13, 48]. Previous methods' limitation lies in their assumption of equal influences from the skip connection and the attention weights, neglecting the difference between original and transformed tokens.

4.2. Token Transformation Measurement

To account for the contributions from transformed tokens, we introduce a measurement that gauges the influence of token transformations and subsequently derives transformation weights. Using these weights, we recalibrate the matrices $\mathbf{O}$ and $\mathbf{T}$ to encapsulate both attention and transformation insights. Since MHSA and FFN are depicted in a unified form (see Section 3), we can first derive our measurement based on MHSA, and then adapt it for FFN from a generic perspective.

Drawing from foundational principles of token representations, we emphasize two core attributes: length and direction [27, 44]. The intuition is that tokens with greater lengths and consistent spatial orientations predominantly determine the results of linear combinations. Thus, our measurement will involve two components, corresponding to these two attributes. The first component is a length function $\mathrm{L}(\mathbf{x})\!:\!\mathbb{R}^d \!\to\! \mathbb{R}^+$, which measures the length of a token, whether the original or transformed. Mathematically, we instantiate L using L^2 norm of the embedding space $\mathbb{R}^d$, *i.e.,* $\mathrm{L}(\mathbf{x}) = \|\mathbf{x}\|_2$. Considering both the length measurement L and the attention weights, we reintroduce $\mathbf{O}$ and $\mathbf{T}$ as:

$$\mathbf{O} = \mathbf{I} \cdot \mathrm{diag}(\mathrm{L}(\mathbf{E}_1), \mathrm{L}(\mathbf{E}_2), \ldots, \mathrm{L}(\mathbf{E}_n)), \tag{9}$$

$$\mathbf{T} = \mathbf{A} \cdot \mathrm{diag}(\mathrm{L}(\widetilde{\mathbf{E}}_1), \mathrm{L}(\widetilde{\mathbf{E}}_2), \ldots, \mathrm{L}(\widetilde{\mathbf{E}}_n)). \tag{10}$$

Note that C_{ij} indicates the contribution of the j-th input token. Thus we apply the function L column-wise. This approach accounts for both the attention weights and the tokens themselves. To evaluate the changes in contributions after transformations, we now regard the original tokens as reference units and analyze the relative effects of transformations. This is done by normalizing $\mathbf{O}$ and $\mathbf{T}$ column-wise *w.r.t.* the lengths of the original tokens. As a result, we obtain:

$$\mathbf{O} = \mathbf{I}, \quad \mathbf{T} = \mathbf{A} \cdot \mathrm{diag}\left(\frac{\mathrm{L}(\widetilde{\mathbf{E}}_1)}{\mathrm{L}(\mathbf{E}_1)}, \frac{\mathrm{L}(\widetilde{\mathbf{E}}_2)}{\mathrm{L}(\mathbf{E}_2)}, \ldots, \frac{\mathrm{L}(\widetilde{\mathbf{E}}_n)}{\mathrm{L}(\mathbf{E}_n)}\right). \tag{11}$$

In this formulation, $\mathbf{O}$ uses the identity matrix to represent the contributions from original tokens as basic reference units. Meanwhile, $\mathbf{T}$ discerns the relative influences of transformed tokens compared to the original ones using the ratio of their lengths. As detailed in Section 4.3, this approach allows us to initialize a contribution map based on the lengths of the model's input tokens, and then iteratively update the map across layers. The update employs the ratios of token effects between consecutive layers, tracing the evolution of transformations within the model. This framework ensures that our analysis remains grounded to the initial input to the model, yet dynamically adapts to every token transformation and contextualization encountered during the inference process.

Our second component focuses on directions. Beyond adjusting lengths, token transformations also influence directions [18]. A transformed token that stays directionally closer in representation space is expected to have a stronger correlation with its original counterpart. To quantify this directional correlation, we introduce a function $\mathrm{C}(\mathbf{x}, \widetilde{\mathbf{x}})\!:\!\mathbb{R}^d \!\times\! \mathbb{R}^d \!\to\! \mathbb{R}$. Mathematically, we employ Cosine similarity, which measures the angle between a pair of tokens, *i.e.,* $\mathrm{C}(\mathbf{x}, \widetilde{\mathbf{x}}) = \cos\langle \mathbf{x}, \widetilde{\mathbf{x}} \rangle$. Next, we will use the function C to complement the length factors in matrix $\mathbf{T}$. Simply multiplying C as a coefficient may introduce negative

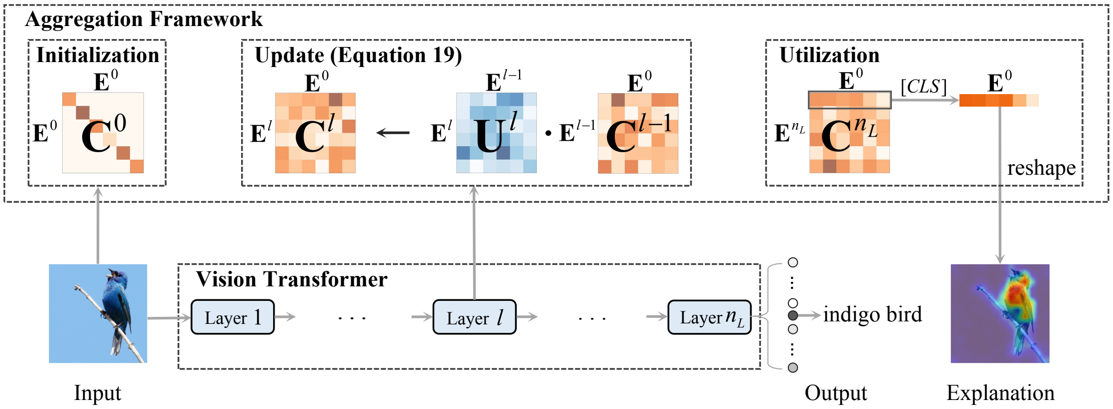

Figure 3. Illustration of our aggregation framework and the explanation pipeline. The overall contribution map is initialized by input token lengths and is updated using our $\mathbf{U}^l$ to trace token evolution across layers. In $\mathbf{C}^l$, each i-th row represents the influences of input tokens $\mathbf{E}^0$ on the output of the l-th layer $\mathbf{E}^l$. For $\mathbf{C}^{n_L}$, the row *w.r.t.* $[CLS]$ token is extracted and reshaped to produce the final explanation map.

values to the contribution map, which will distort the signs of contributions through aggregation [6, 12, 13, 37, 42]. Inspired by [29], we propose the Normalized Exponential Cosine Correlation (NECC). This measurement is normalized to emphasize the relative magnitudes of each correlation instead of the polarity, thereby serving as an effective positive weighting factor. For a sequence of original tokens $\mathbf{S}_{\mathbf{E}} = (\mathbf{E}_1, \mathbf{E}_2, \ldots, \mathbf{E}_n)$ and their transformed counterparts $\mathbf{S}_{\widetilde{\mathbf{E}}} = (\widetilde{\mathbf{E}}_1, \widetilde{\mathbf{E}}_2, \ldots, \widetilde{\mathbf{E}}_n)$, the weighting factor for the i-th pair is given by:

$$\mathrm{NECC}(i) = \frac{\exp(\mathrm{C}(\mathbf{E}_i, \widetilde{\mathbf{E}}_i))}{\sum_{k=1}^{n} \exp(\mathrm{C}(\mathbf{E}_k, \widetilde{\mathbf{E}}_k))}. \tag{12}$$

Treating the original tokens as reference units, NECC is proportional to the extent to which each transformed token correlates with its corresponding original token. Finally, employing two components of our transformation measurement, we define the transformation weights $\mathbf{W}$:

$$\mathbf{W} = \mathrm{diag}\left(\frac{\mathrm{L}(\widetilde{\mathbf{E}}_1)}{\mathrm{L}(\mathbf{E}_1)}\mathrm{NECC}(1), \ldots, \frac{\mathrm{L}(\widetilde{\mathbf{E}}_n)}{\mathrm{L}(\mathbf{E}_n)}\mathrm{NECC}(n)\right), \tag{13}$$

and the update map $\mathbf{U}$ for MHSA, incorporating both attention and transformation information:

$$\mathbf{U} = \mathbf{O} + \mathbb{E}_h \left[(\nabla_{\mathbf{A}} p(c))^+ \odot \mathbf{T} \right], \tag{14}$$

with

$$\mathbf{O} = \mathbf{I}, \quad \mathbf{T} = \mathbf{A} \cdot \mathbf{W}. \tag{15}$$

Here, U_{ij} denotes the influence of the j-th input token on the i-th output token of the MHSA layer. This formulation balances both length and direction, reflecting how much of the original information is retained or altered in the transformed tokens, to faithfully evaluate their contributions.

We now adapt our established approach to the FFN layer. Recall that FFN also involves a weighted sum of original and transformed tokens (see Eq. (7)). For 'contextualization' in FFN, there is only a single-head and the weighted combination is performed locally. This simpler form can be easily reflected by discarding the gradient-weighted multi-head integration from Eq. (14) and changing the attention map $\mathbf{A}$ in Eq. (15) to an identity matrix. We formulate the update map for the FFN layer:

$$\mathbf{U} = \mathbf{O} + \mathbf{T}, \tag{16}$$

with

$$\mathbf{O} = \mathbf{I}, \quad \mathbf{T} = \mathbf{I} \cdot \mathbf{W}. \tag{17}$$

Here, U_{ij} denotes the influence of the j-th input token on the i-th output token of the FFN layer. As illustrated by Figure 3, update maps capture the relative effects and will serve as refinements to the overall contribution map as it aggregates across multiple layers.

4.3. Aggregation Framework

In Vision Transformers, the influences of token transformations and contextualizations do not merely reside within isolated layers but accumulate across them. For a comprehensive contribution map that reflects the roles of initial tokens corresponding to input image regions, it is necessary to assess the relationships between input tokens and their evolved counterparts in deeper layers. To this end, we introduce an aggregation framework that cohesively measures the combined influences of contextualization and transformation across the entire model.

4.3.1 Initialization

We denote the overall contribution map by $\mathbf{C} \in \mathbb{R}^{n \times n}$, where n is the number of tokens. The C_{ij} will faithfully accumulate the influence of the j-th input token on the i-th output token. At the initial state, before entering Transformer layers, each input token simply contains itself without contextualization or transformation. We represent this

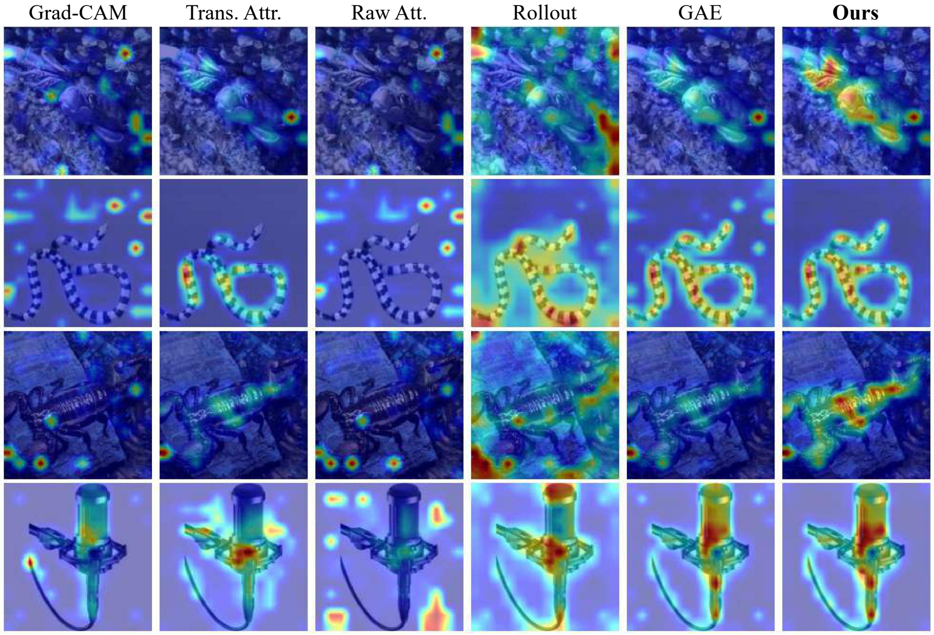

Figure 4. Visualizations of post-hoc explanation heatmaps. Our method captures more object-centric regions as the rationales.

state using initial tokens' lengths, as depicted in Figure 3. Specifically, we initialize the map as a diagonal matrix $\mathbf{C}^0$:

$$\mathbf{C}^0 = \text{diag}(\text{L}(\mathbf{E}_1^0), \text{L}(\mathbf{E}_2^0), \ldots, \text{L}(\mathbf{E}_n^0)), \tag{18}$$

where $\mathbf{E}^0$ denotes the initial embeddings that are fed to the first Transformer layer.

4.3.2 Update Rules

Based on the DAG representation of information flow [1], layer-wise aggregation is mathematically equivalent to matrix multiplication of intermediate maps. This approach allows for tracing token contributions over the entire model. After initialization, we iteratively update the contribution map $\mathbf{C}^{l-1}$ using $\mathbf{U}^l$, which incorporates the effects of transformed tokens:

$$\mathbf{C}^l \leftarrow \mathbf{U}^l \cdot \mathbf{C}^{l-1} \quad \text{for} \quad l = 1, 2, \ldots, n_L. \tag{19}$$

In this formula, $\mathbf{C}^{l-1}$ represents the map that has traced the token contributions up to the $(l-1)$-th layer but has yet to include the effects from the l-th layer, and $\mathbf{U}^l$ indicates the update map from the l-th layer. Using this recursive formula, the map aggregates information about both token transformation and contextualization over the entire model. After applying the updates for all the layers, the final contribution map can be expressed as:

$$\mathbf{C}^{n_L} = \mathbf{U}^{n_L} \cdot \mathbf{U}^{n_L-1} \cdot \ \cdots \ \cdot \mathbf{U}^1 \cdot \mathbf{C}^0, \tag{20}$$

This framework provides a cumulative understanding of how initial input tokens contribute to the final output, thereby faithfully localizing the most important pixels for Vision Transformers' predictions.

4.4. Utilizing Contribution Map for Interpretation

Vision Transformers often use designated tokens to perform specific tasks. For example, ViT [17] performs image classification based on a $[CLS]$ token. This special token becomes significant in the final prediction as it integrates information from all the inputs. To interpret the model's prediction, one can analyze the contribution of each initial input token to the final $[CLS]$ token [1, 13, 26, 35]. In our proposed TokenTM, this is embodied in the overall contribution map $\mathbf{C}^{n_L}$. Specifically, the row in $\mathbf{C}^{n_L}$ that corresponds to the $[CLS]$ token represents the influences of original tokens. As illustrated in Figure 3, we extract this row and reshape it to the image's spatial dimensions, forming a contribution heatmap. This heatmap highlights regions that are highly influential in determining the prediction, serving as a post-hoc explanation of the model.

5. Experiments

5.1. Baseline Methods

We consider baseline methods that are widely used and applicable from three types. **(i)** Gradient-based: Grad-CAM [37], **(ii)** Attribution-based: LRP [9], Conservative LRP [3], and Transformer Attribution [13]), and **(iii)** Attention-based: Raw Attention [26], Rollout [1], ATTCAT [35], and GAE [12]).

5.2. Evaluated Properties

Localization Ability. This property measures how well an explanation can localize the foreground object recognized by the model. Intuitively, a reliable explanation should be

	Pix. Acc. ↑	mAP ↑	mIoU ↑
Grad-CAM [37]	67.37	79.20	46.13
LRP [9]	50.96	55.87	32.82
Cons. LRP [3]	64.06	68.01	36.23
Trans. Attr. [13]	79.17	85.81	61.02
Raw Att. [26]	59.07	76.51	36.36
Rollout [1]	59.01	73.76	39.43
ATTCAT [35]	43.88	48.38	27.90
GAE [12]	79.05	85.71	61.56
Ours	**81.79**	**86.45**	**64.22**

Table 1. Results of Localization Ability on ImageNet-Seg. [21].

object-centric, *i.e.*, accurately highlighting the object that the model uses to make the decision. Following previous works [6, 12, 13, 37], we perform segmentation on the ImageNet-Segmentation dataset [21], using DeiT [46], a prevalent Vision Transformer model. We regard the explanation heatmaps as preliminary semantic signals. Then we use the average value of each heatmap as a threshold to produce binary segmentation maps. Based on the ground-truth maps, we evaluate the performance on three metrics: Pixel-wise Accuracy, mean Intersection over Union (mIoU), and mean Average Precision (mAP).

Impact on Accuracy. This aspect focuses on how well an explanation captures the correlations between pixels and the model's accuracy. We assess this property by perturbation tests on CIFAR-10, CIFAR-100 [28], and ImageNet [36]. The pre-trained DeiT model is used to perform both positive and negative perturbation tests *w.r.t.* the predicted and the ground-truth class. Specifically, these tests include two stages. First, we produce explanation maps for the class we want to visualize. Second, we gradually remove pixels and evaluate the resulting mean top-1 accuracy. In positive tests, pixels are removed from the most important to the least, while in negative tests, the removal order is reversed. A faithful explanation should assign higher scores to pixels that contribute more. Thus, a severe drop in the model's accuracy is expected in positive tests, while we expect the accuracy to be maintained in negative tests. For quantitative evaluation, we compute the Area Under the Curve (AUC) [4] *w.r.t.* the accuracy curve of different perturbation levels.

Impact on Probability. This further measures how well the explanations capture important pixels for the model's predicted probabilities. Similarly, it is evaluated by perturbation tests. We report results on Area Over the Perturbation Curve (AOPC) and Log-odds score (LOdds) [14, 34, 39]. These metrics quantify the average change of output probabilities *w.r.t.* the predicted label. For comprehensive evaluations, we report results on ViT-B and ViT-L [17].

	Predicted label		GT label	
	Neg. ↑	Pos. ↓	Neg. ↑	Pos. ↓
Grad-CAM [37]	61.24	47.32	61.26	47.34
LRP [9]	72.62	72.14	72.54	72.13
Cons. LRP [3]	44.92	49.15	45.76	47.87
Trans. Attr. [13]	67.65	43.64	65.82	43.17
Raw Att. [26]	58.46	47.17	58.46	47.17
Rollout [1]	64.88	42.89	64.88	42.89
ATTCAT [35]	41.78	42.66	41.91	42.52
GAE [12]	71.67	37.77	71.81	37.70
Ours	**74.90**	**37.14**	**75.05**	**36.48**

Table 2. Results of Impact on Accuracy on CIFAR-10 [28].

	Predicted label		GT label	
	Neg. ↑	Pos. ↓	Neg. ↑	Pos. ↓
Grad-CAM [37]	47.18	41.91	47.28	41.78
LRP [9]	46.76	46.31	46.80	46.29
Cons. LRP [3]	27.11	27.99	26.84	27.42
Trans. Attr. [13]	44.65	24.99	44.19	23.90
Raw Att. [26]	37.23	28.62	37.23	28.62
Rollout [1]	41.34	24.96	41.34	24.96
ATTCAT [35]	23.52	22.15	23.90	21.73
GAE [12]	52.14	17.52	52.65	17.32
Ours	**54.78**	**16.07**	**55.30**	**15.89**

Table 3. Results of Impact on Accuracy on CIFAR-100 [28].

	Predicted label		GT label	
	Neg. ↑	Pos. ↓	Neg. ↑	Pos. ↓
Grad-CAM [37]	43.84	26.11	44.06	25.87
LRP [9]	42.55	41.29	42.56	41.29
Cons. LRP [3]	33.24	34.35	34.33	34.03
Trans. Attr. [13]	57.48	19.00	58.26	18.38
Raw Att. [26]	48.70	25.44	48.70	25.44
Rollout [1]	43.23	32.34	43.23	32.34
ATTCAT [35]	32.34	31.71	33.59	30.68
GAE [12]	57.53	15.88	58.85	15.20
Ours	**58.12**	**15.75**	**59.36**	**15.09**

Table 4. Results of Impact on Accuracy on ImageNet [36].

5.3. Experimental Results

Qualitative Evaluation. As shown in Figure 4, interpretation heatmaps of TokenTM are more precise and exhaustive. Rather than indicating plain background areas as rationale, the integration of cross-layer token transformation information in TokenTM effectively eliminates noisy regions, allowing a more object-focused analysis. More visualizations are provided in the supplementary.

Quantitative Evaluation. (i) Localization Ability. Table 1 reports the segmentation results on the ImageNet-Segmentation dataset. Our proposed TokenTM significantly outperforms all the baselines on pixel accuracy, mIoU, and mAP, demonstrating its stronger localization ability. **(ii) Impact on Accuracy.** Tables 2, 3, and 4 show the results of the perturbation tests for both classes. For positive tests, a lower AUC indicates superior performance, while for negative tests, a higher AUC is desirable. The results underscore the superiority of our method. **(iii) Impact on Probability.**

	AOPC ↑		LOdds ↓	
	ViT-B [17]	ViT-L [17]	ViT-B [17]	ViT-L [17]
Grad-CAM [37]	0.557	0.457	-4.020	-3.023
LRP [9]	0.474	0.508	-3.259	-3.634
Cons. LRP [3]	0.613	0.614	-4.272	-4.443
Trans. Attr. [13]	0.721	0.716	-5.622	-5.541
Raw Att. [26]	0.652	0.655	-4.911	-4.961
Rollout [1]	0.688	0.689	-5.259	-5.183
ATTCAT [35]	0.641	0.645	-4.461	-4.622
GAE [12]	0.714	0.699	-5.437	-5.601
Ours	**0.764**	**0.726**	**-5.781**	**-5.755**

Table 5. Results of Impact on Probability on more Vision Transformer models.

	Segmentation			Perturbation Test	
	Pix. Acc. ↑	mAP ↑	mIoU ↑	Neg. ↑	Pos. ↓
Baseline	52.94	67.78	33.42	44.92	31.95
+ AF	76.30	85.28	58.34	55.25	16.98
+ AF + L	77.50	85.39	59.59	55.03	16.28
+ AF + L + NECC	**78.22**	**85.69**	**60.29**	**55.42**	**16.18**

Table 6. Results of ablation study on the proposed components, *i.e.*, transformation measurement (L and NECC) and aggregation framework (AF).

Table 5 reports the results of perturbation tests assessing the model's output probability. Our TokenTM achieves the best performance on Vision Transformers variants.

5.4. Ablation Study

Ablation Study on Proposed Components. We conduct ablation studies on the effect of our proposed transformation measurement (L and NECC) and aggregation framework (AF). We perform experiments on ImageNet [36], based on ViT-B [17]. Four variants are considered in our ablative experiment: **(i)** Baseline, which merely applies Eq. (8) to the input tokens without applying our proposed components, **(ii)** Baseline + AF, which initializes the contribution map as an identity matrix and aggregates across multiple layers using $\mathbf{U}^l$, where our proposed relative lengths and NECC measurements are discarded, **(iii)** Baseline + AF + L, which additionally use relative length measurement in $\mathbf{U}^l$ while still excluding the NECC terms, and **(iv)** Baseline + AF + L + NECC, which is our TokenTM. As shown in Table 6, each proposed component improves the performance on both segmentation and perturbation tests, validating their effectiveness in the Vision Transformer explanation.

Ablation Study on Aggregation Depth. To better understand the cumulative nature of Vision Transformers using TokenTM, we study the impact of varying the aggregation depth within our method. Specifically, we increase the number of aggregated layers from the initial few to the entire network depth. Table 7 reports the ablative results on ImageNet [36], using ViT-B [17], where 'n layers' denotes a variant of our TokenTM that (only) aggregates the first n layers of the model. The results reveal a consistent enhancement in performance with increased layer aggregation depth, which suggests that a deeper aggregation of token transformation and contextualization effects is pivotal for capturing the true rationale of the model's inference. Furthermore, Figure 5 illustrates the reasoning procedure of the Vision Transformer. As the aggregation encompasses more layers, the heatmaps progressively refine the focused areas and become more concentrated on the object, localizing the rationale behind the model's prediction.

	Segmentation			Perturbation Test	
	Pix. Acc. ↑	mAP ↑	mIoU ↑	Neg. ↑	Pos. ↓
3 layers	63.63	76.40	42.99	43.32	22.57
6 layers	68.38	80.35	48.66	48.28	20.17
9 layers	70.46	81.74	51.11	50.42	19.24
12 layers	**78.22**	**85.69**	**60.29**	**55.42**	**16.18**

Table 7. Results of ablation study on the aggregation depth.

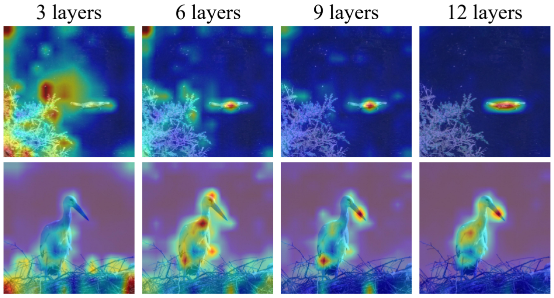

Figure 5. Visualizations of the localization process using our TokenTM. As more layers are incorporated into the aggregation, the heatmaps become increasingly focused, particularly in the regions surrounding the object recognized by the model.

6. Conclusions

In this paper, we explored the challenge of explaining Vision Transformers. Upon reinterpretation, we expressed Transformer layers using a generic form in terms of weighted linear combinations of original and transformed tokens. This new perspective revealed a principal issue in existing post-hoc methods: they solely rely on attention weights and ignore significant information from token transformations, which leads to misunderstandings of rationales behind the models' decisions. To tackle this issue, we proposed TokenTM, an explanation method comprising a token transformation measurement and an aggregation framework. Quantifying the contributions of input tokens throughout the entire model, our method generates more faithful post-hoc interpretations for Vision Transformers.

Acknowledgments: This research is supported by NSF IIS-2309073 and ECCS-212352101. This article solely reflects the opinions and conclusions of its authors and not the funding agents.

References

[1] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *ACL*, 2020. 1, 2, 3, 6, 7, 8

[2] Chirag Agarwal, Satyapriya Krishna, Eshika Saxena, Martin Pawelczyk, Nari Johnson, Isha Puri, Marinka Zitnik, and Himabindu Lakkaraju. Openxai: Towards a transparent evaluation of model explanations. In *NeurIPS*, 2022. 1

[3] Ameen Ali, Thomas Schnake, Oliver Eberle, Grégoire Montavon, Klaus-Robert Müller, and Lior Wolf. Xai for transformers: Better explanations through conservative propagation. In *ICML*, 2022. 2, 6, 7, 8

[4] Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. A diagnostic study of explainability techniques for text classification. In *EMNLP*, 2020. 7

[5] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. In *PloS one*, 10(7):e0130140, 2015. 2

[6] Oren Barkan, Edan Hauon, Avi Caciularu, Ori Katz, Itzik Malkiel, Omri Armstrong, and Noam Koenigstein. Gradsam: Explaining transformers via gradient self-attention maps. In *CIKM*, 2021. 4, 5, 7

[7] Oren Barkan, Yuval Asher, Amit Eshel, Noam Koenigstein, et al. Visual explanations via iterated integrated attributions. In *ICCV*, 2023. 2

[8] Oren Barkan, Yehonatan Elisha, Jonathan Weill, Yuval Asher, Amit Eshel, and Noam Koenigstein. Deep integrated explanations. In *CIKM*, 2023. 2

[9] Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for neural networks with local renormalization layers. In *ICANN*, 2016. 6, 7, 8

[10] Gino Brunner, Yang Liu, Damian Pascual, Oliver Richter, Massimiliano Ciaramita, and Roger Wattenhofer. On identifiability in transformers. In *ICLR*, 2020. 2

[11] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 1

[12] Hila Chefer, Shir Gur, and Lior Wolf. Generic attention-model explainability for interpreting bi-modal and encoder-decoder transformers. In *ICCV*, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[13] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *CVPR*, 2021. 1, 2, 3, 4, 5, 6, 7, 8

[14] Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. Generating hierarchical explanations on text classification via feature interaction detection. In *ACL*, 2020. 7

[15] Piotr Dabkowski and Yarin Gal. Real time image saliency for black box classifiers. In *NeurIPS*, 2017. 2

[16] Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. Eraser: A benchmark to evaluate rationalized nlp models. In *ACL*, 2020. 1

[17] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 1, 3, 6, 7, 8

[18] Kawin Ethayarajh. How contextual are contextualized word representations? comparing the geometry of bert, elmo, and gpt-2 embeddings. In *EMNLP-IJCNLP*, 2019. 4

[19] Ruth C Fong and Andrea Vedaldi. Interpretable explanations of black boxes by meaningful perturbation. In *ICCV*, 2017. 2

[20] Jindong Gu, Yinchong Yang, and Volker Tresp. Understanding individual decisions of cnns via contrastive backpropagation. In *ACCV*, 2018. 2

[21] Matthieu Guillaumin, Daniel Küttel, and Vittorio Ferrari. Imagenet auto-annotation with segmentation propagation. *IJCV*, 110:328–348, 2014. 7

[22] Shir Gur, Ameen Ali, and Lior Wolf. Visualization of supervised and self-supervised neural networks via attribution guided factorization. In *AAAI*, 2021. 2

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016. 3

[24] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 1

[25] Brian Kenji Iwana, Ryohei Kuroki, and Seiichi Uchida. Explaining convolutional neural networks using softmax gradient layer-wise relevance propagation. In *ICCV Workshop*, 2019. 2

[26] Sarthak Jain and Byron C Wallace. Attention is not explanation. In *NAACL*, 2019. 1, 2, 6, 7, 8

[27] Goro Kobayashi, Tatsuki Kuribayashi, Sho Yokoi, and Kentaro Inui. Attention is not only a weight: Analyzing transformers with vector norms. In *EMNLP*, 2020. 4

[28] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 7

[29] Yang Liu, Qingchao Chen, and Samuel Albanie. Adaptive cross-modal prototypes for cross-domain visual-language retrieval. In *CVPR*, 2021. 5

[30] Ze Liu, Han Hu, Yutong Lin, Zhuliang Yao, Zhenda Xie, Yixuan Wei, Jia Ning, Yue Cao, Zheng Zhang, Li Dong, et al. Swin transformer v2: Scaling up capacity and resolution. In *CVPR*, 2022. 1

[31] Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In *NeurIPS*, 2017. 2

[32] Aravindh Mahendran and Andrea Vedaldi. Visualizing deep convolutional neural networks using natural pre-images. *IJCV*, 120:233–255, 2016. 1, 2

[33] Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *PR*, 65:211–222, 2017. 2

[34] Dong Nguyen. Comparing automatic and human evaluation of local explanations for text classification. In *NAACL*, 2018. 7

[35] Yao Qiang, Deng Pan, Chengyin Li, Xin Li, Rhongho Jang, and Dongxiao Zhu. Attcat: Explaining transformers via at-

tentive class activation tokens. In *NeurIPS*, 2022. 1, 2, 6, 7, 8

[36] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115:211–252, 2015. 7, 8

[37] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, 2017. 1, 2, 5, 6, 7, 8

[38] Sofia Serrano and Noah A Smith. Is attention interpretable? In *ACL*, 2019. 1

[39] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *ICML*, 2017. 2, 7

[40] Avanti Shrikumar, Peyton Greenside, Anna Shcherbina, and Anshul Kundaje. Not just a black box: Learning important features through propagating activation differences. In *ICML*, 2017. 2

[41] Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. In *ICML Workshop*, 2017. 2

[42] Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for simplicity: The all convolutional net. In *ICLRW*, 2015. 5

[43] Suraj Srinivas and François Fleuret. Full-gradient representation for neural network visualization. In *NeurIPS*, 2019. 2

[44] Gilbert Strang. *Linear algebra and its applications*. Belmont, CA: Thomson, Brooks/Cole, 2006. 4

[45] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *ICML*, 2017. 1, 2

[46] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021. 1, 7

[47] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 1, 2

[48] Elena Voita, David Talbot, Fedor Moiseev, Rico Sennrich, and Ivan Titov. Analyzing multi-head self-attention: Specialized heads do the heavy lifting, the rest can be pruned. In *ACL*, 2019. 4

[49] Sarah Wiegreffe and Yuval Pinter. Attention is not not explanation. In *EMNLP*, 2019. 1, 2

[50] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, 2014. 2

[51] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, 2016. 2

[52] Bolei Zhou, David Bau, Aude Oliva, and Antonio Torralba. Interpreting deep visual representations via network dissection. *IEEE TPAMI*, 41(9):2131–2145, 2018. 1