ProtoPFormer: Concentrating on Prototypical Parts in Vision Transformers for Interpretable Image Recognition

Mengqi Xue[1] , **Qihan Huang**[2] , **Haofei Zhang**[2] , **Jingwen Hu**[2] , **Jie Song**[2,3] , **Mingli Song**[1,2,3] and **Canghong Jin**[1]
[1]Hangzhou City University
[2]Zhejiang University
[3]ZJU-Bangsun Joint Research Center
{mqxue, jinch}@hzcu.edu.cn, {qh.huang, haofeizhang, jw_hu, sjie, brooksong}@zju.edu.cn

Abstract

Prototypical part network (ProtoPNet) and its variants have drawn wide attention and been applied to various tasks due to their inherent self-explanatory property. Previous ProtoPNets are primarily built upon convolutional neural networks (CNNs). Therefore, it is natural to investigate whether these explainable methods can be advantageous for the recently emerged Vision Transformers (ViTs). However, directly utilizing ViT-backed models as backbones can lead to prototypes paying excessive attention to background positions rather than foreground objects (*i.e.*, the "distraction" problem). To address the problem, this paper proposes prototypical part Transformer (ProtoPFormer) for interpretable image recognition. Based the architectural characteristics of ViTs, we modify the original ProtoPNet by creating separate global and local branches, each accompanied by corresponding prototypes that can capture and highlight representative holistic and partial features. Specifically, the global prototypes can guide local prototypes to concentrate on the foreground and effectively suppress the background influence. Subsequently, local prototypes are explicitly supervised to concentrate on different discriminative visual parts. Finally, the two branches mutually correct each other and jointly make the final decisions. Moreover, extensive experiments demonstrate that ProtoPFormer can consistently achieve superior performance on accuracy, visualization results, and quantitative interpretability evaluation over the state-of-the-art (SOTA) baselines. Our code has been released at *https://github.com/zju-vipa/ProtoPFormer*.

1 Introduction

The emergence of deep neural networks (DNNs) has created unprecedented achievements in machine learning, thanks to their powerful capabilities of learning representations [Talaei Khoei *et al.*, 2023]. However, the lack of transparency hinders DNNs' wider applications in areas requiring traceable and understandable decisions. To further explore the interpretability of DNNs, many researchers propose various approaches to promote the advancement in explainable artificial intelligence (XAI) [Gao and Guan, 2023]. Among these methods, ProtoPNet [Chen *et al.*, 2019], inspired by the human vision system, has attracted increasing research interest and many follow-up studies with its self-explanatory property for XAI. Considering the example image in Fig. 1, we can identify this sample as a Painted Bunting by comparing the features of its beak, wings, and feathers with existing bird species, even without expertise. Similarly, ProtoPNet aims to precisely perceive and recognize discriminative parts of objects with category-specific prototypes. As shown in Fig. 1 (a), the highest activated prototypes capture features of the bird's head and wings. By making predictions through the linear combination of prototypes' similarity with image patches, ProtoPNet is inherently interpretable and can be analyzed by visualization when post-processing.

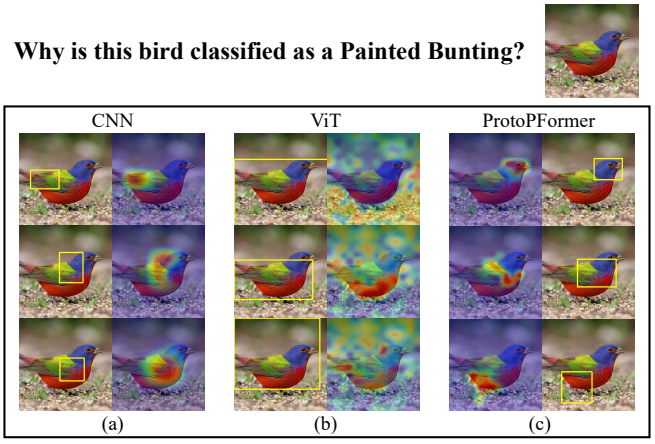

Figure 1: Visual comparison of prototypes on an example image between a CNN-based ProtoPNet (ResNet34) and a ViT-based ProtoPNet (DeiT-Ti), and our ProtoPFormer (DeiT-Ti).

ProtoPNet and its variants are mainly developed on convolutional neural networks (CNNs). While recent years vision Transformer (ViTs) [Dosovitskiy *et al.*, 2021] have been introduced into computer vision, challenging the domination of CNNs with their superior performance due to the

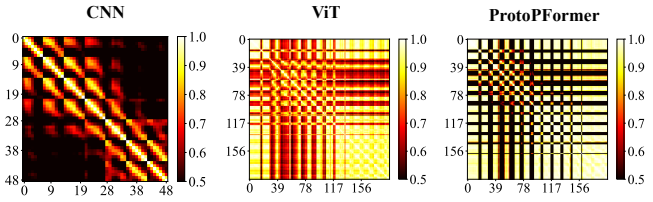

Figure 2: Heatmaps of unit similarities in three backbones.

ability to model long-range dependencies [Liu *et al.*, 2021; Chen *et al.*, 2022a]. Naturally, we want to investigate whether prototype-based methods can be utilized for ViT-backed models with their case-based reasoning process. Disappointingly, directly applying ProtoPNet to a ViT (removing the class token) leads to a "distraction" problem: the learned prototypes are prone to obtain high activation scores in the background and show scattered and fragmented activation scores in the foreground, as shown in Fig. 1 (b). The "distraction" problem stems from the homogeneity of different units in ViTs. Fig. 2 shows that the ViT has significantly higher unit similarities than the CNN. The unit similarities are the average similarities between all pairs of units or tokens in the feature map or visual sequence, which indicates that the same prototype may have high similarity with many units in the same token sequence and generate scattered visualization results in the ViT (more quantitative analysis experiments of unit similarities are provided in the the supplementary material). The unsatisfactory visualization results violate the idea that makes prototypes point out critical visual evidence for each case in ProtoPNet. The lack of intrinsic inductive bias makes prototypes of ViTs focus less on prototypical parts and more on long-range dependency.

To solve the aforementioned limitations, we propose *prototypical part transformer* (ProtoPFormer) to appropriately and effectively apply the prototype-based method with ViTs for interpretable image recognition in two steps, as shown in Fig. 1 (c). ProtoPFormer proposes global and local prototypes to concentrate the "distracted" prototypes on the discriminative parts to build a self-interpretable ViT-backend model. As reported by [Raghu *et al.*, 2021], the class token (*i.e.*, the global branch) of ViTs progressively aggregates information from all the image tokens and produces the high-level abstraction of targets; on the contrary, image tokens (*i.e.*, the local branch) remain strong similarities to their corresponding spatial locations in inputs. Hence the global and local prototypes are designed to compute similarity scores with output embeddings of the class token and image tokens for capturing and highlighting holistic and partial features of targets and fully capitalizing on the built-in architectural characteristics of ViTs. Next, we gradually perform a two-step concentration process to solve the "distraction" problem with the proposed two types of prototypes. In the first step, the global prototypes perceive the holistic features of targets and devise a foreground preserving (FP) mask to guide the local branch to selectively keep foreground-related image tokens and eliminate the influence of the background. In the second step, A prototypical part concentration (PPC) loss is designed to promote inter-prototype divergence and centralize scattered similarity scores, encouraging local prototypes further focus on diverse prototypical parts as visual explanations. Fig. 2 shows that ProtoPFormer significantly decreases the unit similarities of ProtoPNet on the ViT backbone, which mitigates the homogeneity of different units and makes the prototypes more concentrated. Finally, the predictions from the local and global branches are combined to make final decisions jointly.

Our experiments have proved that combining the two types of high-level abstracted features can mutually correct each other's decisions from their exclusive views. Moreover, extensive experiments have demonstrated that our ProtoPFormer not only enjoys superior performance on accuracy and quantitative interpretability results [Kim *et al.*, 2022b], and faithfully reasons the decision-making processes from both global and local perspectives, appropriately and efficiently resolving the limitations of previous prototype-based methods in ViTs. Besides, more interpretability analysis, ablation studies and visualization results are presented in the supplement. In conclusion, our contributions are summarized as follows:

- Based on the architectural characteristics of ViTs, we propose ProtoPFormer with global and local prototypes to capture and highlight the holistic and partial features of target objects complementarily.
- A two-step process is performed to progressively solve the "prototype distraction" problem and point out visual evidence associatively from global and local perspectives.
- Extensive experiments have demonstrated that ProtoPFormer can achieve superior performance and transparently reason the decision-making processes, benefiting from the strategy of mutual correction and joint decision of global and local prototypes.

2 Related Work

2.1 Interpretability with CNNs

The interpretability of the inherent decision-making process of DNNs has become a grand challenge in computer vision. In general, previous works of model interpretability can be divided into two groups: *self-interpretable models* and *post-hoc analysis* [Gao and Guan, 2023]. Self-interpretable models are elaborately designed neural networks that have transparent reasoning processes with regularization techniques [Subramanian *et al.*, 2018; Böhle *et al.*, 2022] or accountable components [Zhang *et al.*, 2019; Zarlenga *et al.*, 2023]. The post-hoc analysis methods focus on undermining interpretable information of a well-trained DNN with various techniques like visualization [Bychkov *et al.*, 2018], saliency analysis [Chefer *et al.*, 2021a; Hu *et al.*, 2023] and gradients [Selvaraju *et al.*, 2020; Singla *et al.*, 2019]. ProtoPNet [Chen *et al.*, 2019] combines the characteristics of both schools with faithfully reasoning the decision-making process by the linear combination of the prototype's similarity scores and visualizing the importance of discriminative parts as post-hoc analysis. Many follow-up

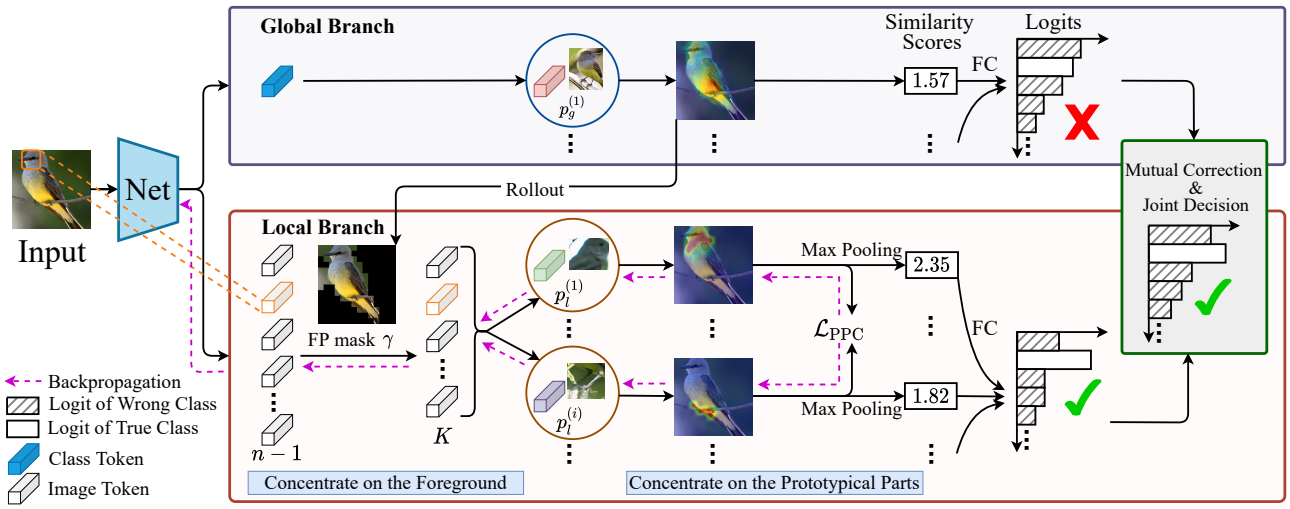

Figure 3: Illustration of ProtoPFormer for image recognition interpretation. The global branch provides guidance for the local branch with the FP mask. The strategy of mutual correction and joint decision makes them contribute complementarily to final predictions, capitalizing on the built-in architectures in ViTs. The loss propagation of $\mathcal{L}_{\text{CE}}$ is omitted for simplicity.

studies extend ProtoPNet to many areas like medical image processing and explanatory debugging [Keswani *et al.*, 2022; Nauta *et al.*, 2021; Donnelly *et al.*, 2022; Bontempelli *et al.*, 2022; Gautam *et al.*, 2023; Huang *et al.*, 2023]. While these CNN-based methods tend to obtain unsatisfactory results both on classification accuracies and visualization of prototypes when directly applying to ViTs [Dosovitskiy *et al.*, 2021]. Besides, recently [Böhle *et al.*, 2022] propose self-interpretable classifiers using patch-wise similarity, which requires enormous labor to define critical patches and interact with machines while only conducting experiments with CNNs. Therefore, we introduce ProtoPFormer to point out visual evidence for ViTs automatically.

2.2 Interpretability with ViTs

Transformer[Vaswani *et al.*, 2017] is introduced into computer vision filed and has achieved impressive success [Touvron *et al.*, 2021a; Liu *et al.*, 2021; Zhang *et al.*, 2023]. With the wide applications of ViTs, some approaches are proposed to explore their interpretability. The most intuitive way is to analyze the attention weights [Vaswani *et al.*, 2017; Abnar and Zuidema, 2020]. Nonetheless, this simple assumption may not be a fail-safe indicator [Serrano and Smith, 2019]. For ViTs, some reasons the decision-making process via gradients [Yao *et al.*, 2022; Chen *et al.*, 2022b], attributions [Chefer *et al.*, 2021b; Yuan *et al.*, 2021] and redundancy reduction [Pan *et al.*, 2021]. While these methods can only attend to the global features, leaving out discriminative parts. In particular, ViT-Net [Kim *et al.*, 2022a] and ConceptTransformer [Rigotti *et al.*, 2021] include visual prototypes/concepts into ViTs as visual explanations. While ViT-Net merely adopts ViTs as backbones to extract features. The interpretability mainly comes from the neural tree that also introduces many parameters and ignores the architectural character of ViTs. ConceptTransformer adds user-defined concepts (*e.g.*, attribute annotations) to enforce an additive relation between token embeddings and concepts. The user-defined concepts require expensive and time-consuming human labeling. Comparatively, ProtoPFormer is designed precisely for ViTs and capitalize on the built-in class token and image tokens in Transformer with category-specific prototypes that can be automatically learned when training.

3 Preliminaries

3.1 ProtoPNets

A typical ProtoPNet is composed of three sequential modules: (1) a backbone network maps an input image to a sequence $X \in \mathbb{R}^{n\times d}$, where n is the length of the visual sequence and d is the embedding dimension; (2) a prototype layer $\text{Proto}(X)$ transforms X into a similarity score vector $s \in \mathbb{R}^m$, where m denotes the number of learnable prototypes; (3) a fully connected (FC) layer $\text{FC}(s)$ makes the prediction with s. In the prototype layer, particularly, the i-th similarity score s_i is the max pooled value from the similarity map $S_i = \text{Sim}(X, p^{(i)})$, where $p^{(i)} \in \mathcal{P}$ is the i-th prototype and $\text{Sim}(\cdot,\cdot)$ computes the similarity between the given prototype p_i and all visual tokens, defined in [Chen *et al.*, 2019]. Generally, ProtoPNets assign k prototypes equally for each class and therefore $m = kC$ (C is the number of classes).

Vision transformers. This paper focuses on ViTs adopting the attention mechanism as the original Transformer [Vaswani *et al.*, 2017]. ViTs firstly embed disjoint image patches as a sequence of image tokens. Then they are appended with a class token and fed into multiple encoder layers composed of a multi-head self-attention (MHSA) module and a multilayer perception (MLP). Given a visual sequence $X \in \mathbb{R}^{n\times d}$, according to [Vaswani *et al.*, 2017] the MHSA layer can be rewritten as

$$\text{MHSA}(X) = \sum_{h=1}^{H} A_h X W_h, \tag{1}$$

where $A_h = \mathrm{softmax}(\Psi_h)$ is the normalized self-attention matrix by row-wise softmax of head h ($\Psi_h \in \mathbb{R}^{n\times n}$), H is the head number, and $W_h \in \mathbb{R}^{d\times d}$ is the linear projection matrix. In particular, the global information is gradually aggregated to the class token solely for the final classification by each MHSA layer.

3.2 ViT-backed ProtoPNets

To directly implement a ViT as the backbone, we remove the class token from the sequence and feed remaining tokens to the following prototype layer. However, unlike CNNs, which are stacked with local perception units, *e.g.*, convolutional and pooling layers, the MHSA mechanism aggregates global information to every visual token, which is consequently incompatible with prototypes representing local visual parts. Furthermore, as experiments demonstrates in Section 4.1, the ViT-backed ProtoPNet and its variants show notably performance degeneration compared with CNN-backed ProtoPNets. This phenomena can be explained by visualizing the learned prototypes shown in Fig. 5, revealing a "prototype distraction" problem that the tokens with high similarity scores are distributed irregularly and spread the whole similarity map, including background positions. Hence, not only the classification accuracy but also the interpretability coming with the semantic meaning of prototypes suffers from the incompatibility between the long-range perception of ViTs and the local visual dependency of ProtoPNets.

4 ProtoPFormer

To tackle the abovementioned problem, we propose explicitly constraining the activated similarity map in the local branch so that every prototype is enforced to represent an individual visual part as intended. Moreover, employing the global information aggregated to the class token, the global branch is proposed to model inter-class and intra-class differences contributing complementarily to final predictions along with the local prototype branch.

The overall architecture is illustrated in Fig. 3, in which a class token $t_c \in \mathbb{R}^{1\times d}$ and a feature sequence $X_f \in \mathbb{R}^{(n-1)\times d}$ are split from the visual sequence $X = [t_c, X_f]$ and fed into a global prototype branch and a local prototype branch respectively. Similar to the CNN-backed ProtoPNets settings, the global branch has m_g learnable prototypes $\mathcal{P}_g = \{p_g^{(1)}, \ldots, p_g^{(m_g)}\}$ (m_g^c for each class), and the local branch has m_l learnable prototypes $\mathcal{P}_l = \{p_l^{(1)}, \ldots, p_l^{(m_l)}\}$ (m_l^c for each class). The predicted logits of the two branches are weighted summed for generating the complementary result, formulated as

$$z_c = \lambda_g z_g + \lambda_l z_l, \tag{2}$$

where z_g and z_l are outputs from the FC classification layer of global and local branches in respective, and $\lambda_{\{g,l\}}$ are weighted coefficients for each branch.

4.1 Concentration on the Foreground

The first step is to concentrate local prototypes on the foreground and eliminate the influence of the background via an adaptive binary mask, named foreground preserving (FP) mask, to selectively preserve foreground-related image tokens and screen background-related image tokens. The global branch provide guidance to the local branch via the FP mask, capitalizing on the built-in class token and image tokens in ViTs. As shown in Figure Fig. 3, we use the rollout method [Abnar and Zuidema, 2020] to generate the FP mask from the rollout matrix of the class token. For a ViT model, the attention rollout matrix at the l-th layer ($l \geq 1$) is defined recursively as $\tilde{\mathcal{A}}_r^{(l)} = \mathcal{A}^{(l)}\tilde{\mathcal{A}}_r^{(l-1)}$, where $\mathcal{A}^{(l)}$ is computed based on the attention matrices $A_h^{(l)}$ at this layer:

$$\mathcal{A}^{(l)} = I_n + \frac{1}{H}\sum_{h=1}^{H} A_h^{(l)}. \tag{3}$$

The initial attention rollout matrix $\mathcal{A}^{(0)}$ is predefined as the identity matrix I_n. As mentioned in [Abnar and Zuidema, 2020], $\tilde{\mathcal{A}}_{1,i}$ is the influence score of the i-th token to the class token, which help us distinguish how likely it could be as a foreground token as visualization in Figure Fig. 3.

Given a backbone ViT with L encoder layer, we extract the rollout matrix at the $(L-1)$-th layer for filtering out background tokens. Let $\hat{a}_c \in \mathbb{R}^{n-1}$ be the rollout attention values to the class token, we only preserve top-K foreground tokens for computing the L-th encoder layer. More specifically, let $\gamma \in \{0,1\}^{n-1}$ denote a binary foreground preserving (FP) mask ($\gamma_i = 1$ represents that token i is a preserved foreground token). To remove selected background tokens, we modify the softmax normalization applied to Ψ in Eq. (1) as

$$A_{i,j} = \frac{\gamma_j \exp(A_{i,j})}{\sum_{k=1}^{n-1}\gamma_k \exp(A_{i,k})}. \tag{4}$$

Eq. (4) cuts off the connection between the background and the foreground, thus avoiding changing information at this step. In the following prototype layer, only the similarity scores generated from the foreground-related tokens are preserved for further class prediction. Generally, this step tentatively concentrates local prototypes in the foreground with the proposed FP mask.

4.2 Concentration on Prototypical Parts

With selected foreground tokens, the local prototypes can be forced to concentrate on heterogeneous visual parts with explicit supervision. For achieving such a purpose, we model the similarity map (reshaped to a two-dimension array like image patches) with regard to each local prototype as a bivariate Gaussian function:

$$\mathrm{Gaussian}(x|\mu,\Sigma) = \frac{1}{2\pi|\Sigma|^{\frac{1}{2}}} e^{-\frac{1}{2}(x-\mu)^\top \Sigma^{-1}(x-\mu)}, \tag{5}$$

where $x \in \mathbb{R}^2$ represents the position on the similarity map; $\mu \in \mathbb{R}^2$ and $\Sigma \in \mathbb{R}^{2\times 2}$ are the parameters controlling the center position and dispersion in respective. On the one hand, by minimizing the eigenvalues of Σ, we are able to dissolve the distraction problem. On the other hand, to promote the divergency of prototypes, we can supervise the prototypes by pushing the centers away from each other.

Gaussian fitting. When achieving N data points $D = \{(x_i, s_i)\}_{i=1}^N$ (x_i represent the position of the similarity

Method	DeiT-Ti			DeiT-S			CaiT-XXS-24		
	CUB	Dogs	Cars	CUB	Dogs	Cars	CUB	Dogs	Cars
Base	80.57 *5.56M*	81.05 *5.55M*	86.21 *5.56M*	84.28 *21.74M*	89.00 *21.71M*	90.06 *21.74M*	83.95 *11.80M*	85.62 *11.79M*	90.19 *11.80M*
ProtoPNet	81.36 *5.95M*	81.47 *5.79M*	86.84 *5.94M*	84.04 *22.12M*	86.85 *21.97M*	88.21 *22.12M*	84.02 *12.18M*	84.62 *12.03M*	88.87 *12.18M*
ProtoTree	68.50 *5.70M*	68.46 *5.70M*	70.02 *5.70M*	70.57 *21.89M*	72.73 *21.89M*	74.95 *21.89M*	72.33 *11.94M*	73.69 *11.94M*	73.15 *11.94M*
TesNet	77.72 *6.38M*	76.54 *5.97M*	84.69 *6.36M*	81.36 *22.56M*	75.08 *22.15M*	87.31 *22.54M*	81.52 *12.62M*	77.01 *12.21M*	88.12 *12.60M*
Def. ProtoPNet	75.79 *7.86M*	79.26 *6.99M*	82.01 *7.82M*	79.53 *25.93M*	79.59 *24.45M*	87.42 *25.86M*	81.09 *14.10M*	80.67 *13.23M*	87.51 *14.05M*
CT	74.71 *5.84M*	N/A	N/A	79.74 *22.75M*	N/A	N/A	78.81 *12.08M*	N/A	N/A
ViT-Net	81.98 ***10.42M***	80.96 ***10.08M***	88.41 ***10.41M***	84.26 ***26.66M***	88.21 ***26.30M***	**91.34** ***26.64M***	84.51 ***16.32M***	84.67 ***16.32M***	**91.54** ***16.65M***
ProtoPFormer	**82.26** *6.33M*	**82.20** *5.91M*	**88.48** *6.13M*	**84.85** *22.50M*	**89.97** *22.09M*	90.86 *22.30M*	**84.79** *12.57M*	**86.26** *12.15M*	91.04 *12.36M*

Table 1: The acc@1 performance comparison (%) between seven SOTA baselines and our ProtoPFormer on three datasets averaged over six runs. Here, "Params" refers to parameter numbers of corresponding ViTs averaged on adopted datasets, please refer to Appendix for detailed model sizes. Bold fonts and blue fonts are used to indicate the best and the second best accuracy, respectively.

value s_i on the heatmap S) along with the FP mask γ, we are able to estimate the parameters of the Gaussian model $\mathrm{Gaussian}(\cdot|\mu, \Sigma)$. Precisely, by removing background tokens, μ and Σ can be estimated by

$$\begin{cases} \hat{\mu} = \dfrac{\sum_{i=1}^{N} \gamma_i s_i x_i}{\sum_{i=1}^{N} \gamma_i s_i}, \\ \hat{\Sigma} = \dfrac{\sum_{i=1}^{N} \gamma_i s_i (x_i - \mu)(x_i - \mu)^\top}{\sum_{i=1}^{N} \gamma_i s_i - 1}. \end{cases} \tag{6}$$

Then we propose a prototypical part concentration (PPC) loss to make local prototypes concentrate on different and centralized representative parts for each class. By constraining $\mathrm{tr}(\hat{\Sigma})$, the trace of $\hat{\Sigma}$, the PPC loss minimizes the sum of eigenvalues of $\hat{\Sigma}$. In the meantime, this loss also encourages prototypes belonging to the same class to have diverse $\hat{\mu}$. The PPC loss can be computed as $\mathcal{L}_{\mathrm{PPC}} = \lambda_\mu \mathcal{L}^{\mu}_{\mathrm{PPC}} + \lambda_\sigma \mathcal{L}^{\sigma}_{\mathrm{PPC}}$, where $\mathcal{L}^{\mu}_{\mathrm{PPC}}$ is defined as

$$\mathcal{L}^{\mu}_{\mathrm{PPC}} = \frac{1}{m^l_c m^l_c} \sum_{i \neq j} \max(t_\mu - \|\hat{\mu}^c_i - \hat{\mu}^c_j\|^2, 0), \tag{7}$$

and $\mathcal{L}^{\sigma}_{\mathrm{PPC}}$ can be written as

$$\mathcal{L}^{\sigma}_{\mathrm{PPC}} = \mathrm{tr}\left(\max(0, \hat{\Sigma} - t_\sigma)\right). \tag{8}$$

Here t_μ and t_σ are two predefined thresholds to guarantee that the PPC loss only penalizes too close center coordinates and encourages small covariance values for concentrating c-class private local prototypes to learn and make decisions on different distinctive visual concepts. λ_μ and λ_σ represent their factors. The final optimization objection of ProtoPFormer is to minimize $\mathcal{L} = \mathcal{L}_{\mathrm{CE}} + \mathcal{L}_{\mathrm{PPC}}$, $\mathcal{L}_{\mathrm{CE}}$ is the conventional cross-entropy loss.

5 Experiments

5.1 Experimental Settings

Datasets. We conduct experiments on three widely-used datasets including CUB [Welinder *et al.*, 2010], Dogs [Khosla *et al.*, 2011] and Cars [Krause *et al.*, 2013]. All images are resized to 224×224 pixels without cropping.

Backbones. Three popular vision Transformers: DeiT-Ti [Touvron *et al.*, 2021a], DeiT-S [Touvron *et al.*, 2021a], and CaiT-XXS-24 [Touvron *et al.*, 2021b], are adopted as the ViT backbones, initialized with the official pre-trained weights on ImageNet-1k [Russakovsky *et al.*, 2015].

Parameters. All models are trained for 200 epochs with AdamW optimizer [Loshchilov and Hutter, 2019] and cosine LR scheduler. The weighted coefficients $\lambda_\mu, \lambda_\sigma$ are set to 0.5, 0.1, and the two thresholds t_μ, t_σ are 2 and 1. We use 10 local prototypes for all datasets, and 10, 5, and 5 global prototypes for CUB, Dogs and Cars datasets, respectively. K is 81 for CUB and Dogs, and 121 for Cars. FC layers are non-trainable in ProtoPFormer.

Baselines. We compare the proposed ProtoPFormer with the classic and state-of-the-art (SOTA) prototype-based approaches. (1) **Base** represents the vanilla ViT model, serving as the non-interpretable counterpart of our method. (2) **ProtoPNet** [Chen *et al.*, 2019] is the first work that interprets DNNs' decisions through a linear combination of similarity scores of prototypes. (3) **ProtoTree** [Nauta *et al.*, 2021] combines prototypes with decision trees for hierarchical reasoning. (4) **TesNet** [Wang *et al.*, 2021] introduces a transparent embedding space with class-aware basis concepts. (5) **Def. ProtoPNet** [Donnelly *et al.*, 2022] designs spatially flexible prototypes for handling images with pose variations. (6) **CT** [Rigotti *et al.*, 2021] stands for ConceptTransformer which utilizes attributes as visual concepts. This method cannot be applied to Dogs and Cars which lack visual attribute labels. (7) **ViT-Net** [Kim *et al.*, 2022a] integrates ViTs and trainable neural trees based on ProtoTree, which only employs ViTs as feature extractors without fully exploiting their architectural characteristics. For fair comparison, we rerun all baselines with three ViT backbones and use grid search to turn their hyperparameters.

5.2 Performance Comparison

In Section 4.1, we report the top-1 accuracy and parameter numbers of ProtoPFormer and our competitors on involved datasets with three ViT backbones. Generally, it can

be observed that our proposed method consistently achieves superior performance compared to baseline methods and economizing parameters. Specifically, Base, as the non-interpretable counterpart, steadily shows upper-middle performance than other self-interpretable baselines, which implies that previous SOTA prototype-based methods are confronted with the trade-off between accuracy and transparency for ViT-backed models. ProtoPNet, TesNet, and Def. ProtoPNet suffer from varying degrees of performance degradation with Transformer backbones on three datasets, indicating that these approaches, designed based on CNN models, are unsuited for learning prototypical parts from token embeddings in Transformers. CT obtains worse results than our method with visual concepts defined by user-defined attributes on CUB, which require time-consuming labeling and rely on human judgment. In contrast, our prototypes are automatically learned along with the training process, faithfully reflecting the discriminative information for the decisions of ViTs. ViT-Net has comparable accuracy with our method but introduces significant extra parameters with the neural trees. Comparatively, ProtoPFormer solely adds small additional parameters, the global prototypes, to achieve the necessary improvement for ViTs. In summary, extensive experiments have verified that our ProtoPFormer outperforms seven baselines on adopted datasets with three ViTs, meanwhile avoiding increasing too many overheads.

5.3 Visualization Analysis

Reasoning Process. Fig. 4 shows a typical reasoning process of ProtoPFormer. The global and local branches make complementary predictions contributing to the final decision. Prototypes of each branch compute corresponding similarity scores with token embeddings and produce the final points with linearly combined scores. In this case, when classifying an Indigo Bunting, global prototypes mistakenly have a slightly larger response with the holistic features of the Boat Tailed Grackle class than Indigo Bunting. Meanwhile, local prototypes successfully discover that the test bird reveals many close parallels with the Indigo-Bunting's exclusive prototypes: the indigo features covered the bird's head, belly, and black feet. Pointing out prototypical visual evidence helps local prototypes make sound judgments and correct the global branch. With the mutual correction and joint decision strategy, global and local prototypes capture holistic and partial features of target objects in a complementary way, benefiting the final decisions of ProtoPFormer.

Visual Comparison. In this subsection, we analyze the interpretability of ProtoPFormer through visualizing local prototypes after training and compare the results with five baselines, illustrated in Fig. 5. The bounding box covers the top 5% similarity scores in the same map, and heat maps are generated by up-sampling and mapping the activation maps to the pixel space, following the same visualization process in ProtoPNet. It can be observed that previous prototype-based competitors obtain unsatisfied visualization results: similarity scores are distributed irregularly and spread throughout the whole similarity map while paying excessive attention on the background. For example, two prototypes of ProtoPNet show high similarity with the background when classifying

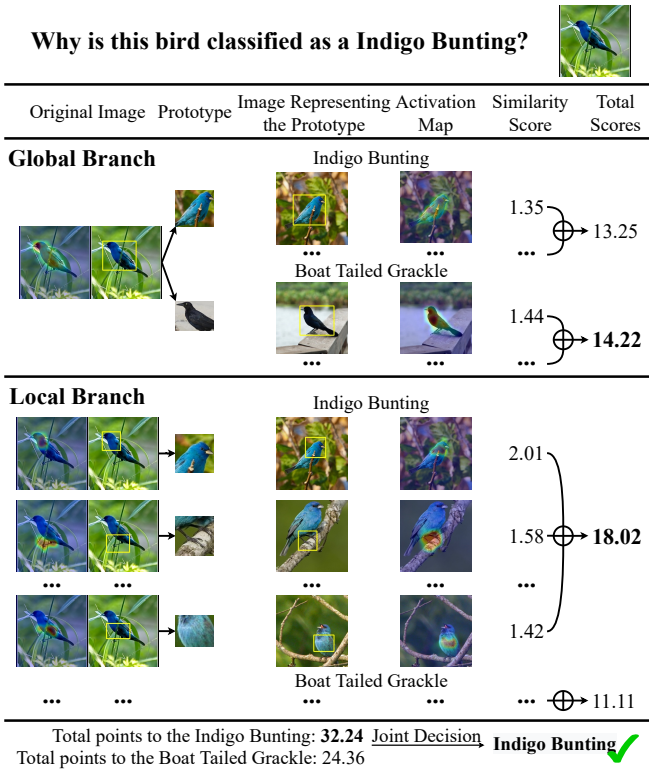

Figure 4: The reasoning process of our ProtoPFormer in classifying the species of a bird with DeiT-Ti, where $\oplus$ denotes summation of similarity scores.

the Red Winged Blackbird, which dramatically impairs its self-explanatory. Nevertheless, ViT-Net and Def.ProtoPNet train prototypes to capture features of the greensward for identifying two dog breeds, as dogs often appear accompanied by the grass. We ascribe this phenomenon to the vulnerability of previous CNN-based baselines to ViT backbones, which mistakenly learn the misleading information related but not congruent with objects and ignore targeting cues. By contrast, ProtoPFormer preciously captures diverse discriminative prototypical parts with the two-step concentration. Notably, the high activated prototypes of ProtoPFormer show significant activations with a central tendency, *e.g.*, the birds' head and under-surface and the dogs' head and belly.

5.4 Analysis of Interpretability for ProtoPFormer

Interpretability Evaluation of Prototypes with HIVE. We follow HIVE [Kim *et al.*, 2022b], a interpretability evaluation method, to quantitatively evaluate the visual interpretability of the learned local prototypes through two evaluation tasks, *i.e.*, a agreement and distinction task, on four SOTA methods and ProtoPFormer. Specifically, the agreement task is used to examine users' confidence in model predictions based on their explanations provided by prototypes, and the distinction task is used to identify the correct predictions based on explanations provided by five methods. We separately compare three sets of 30 examples and 10 examples for the agreement and distinction task with 186 participants (remaining anonymous) on CUB with DeiT-Ti. As shown in Table 2, our pro-

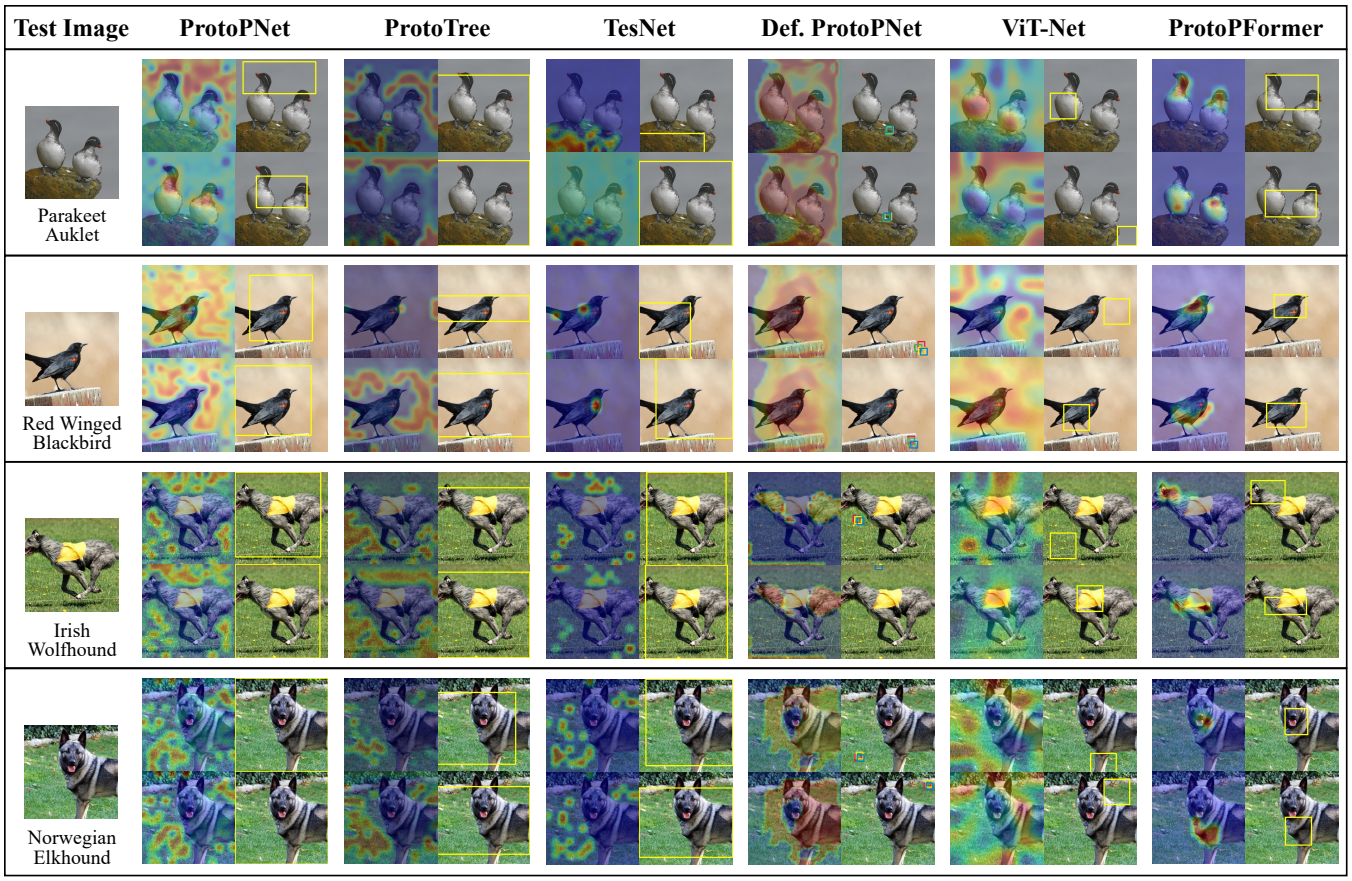

Figure 5: Visual demonstration of the two most activated local prototypes in heat maps and bounding boxes on example images (randomly chosen from the CUB and Dog datasets) of five prototype-based baselines and ProtoPFormer with DeiT-S.

CUB	ProtoPNet	ProtoTree	TesNet	ViT-Net	Ours
Correct	33.54	51.67	50.83	23.13	66.88
Incorrect	63.78	47.07	46.37	75.49	34.98

(a) Agreement task results.

CUB	ProtoPNet	ProtoTree	TesNet	ViT-Net	Ours
	8.89	1.67	2.67	10.00	56.67

(b) Distinction task results.

Table 2: Interpretability evaluation (%) of local prototypes.

posed ProtoPFormer obtains the highest correct scores on the agreement task compared to other methods, indicating that the local prototypes in our method can make local prototypes pay attention to different local parts and capture the representative object parts with Transformer-backed models. Moreover, in the distinction task, we present the activation maps of the five prototype-based methods for the participants and require them to choose the correct explanations for the test images. Table 2 also shows that the activation maps provided by ProtoPFormer are considered as the most discriminative explanation among the competitors by participants.

6 Conclusion

In this paper, we propose ProtoPFormer for appropriately and effectively applying the prototype-based method with ViTs for interpretable image recognition. Experiments have demonstrated that our ProtoPFormer can achieve the superior performance and capture the representative visual evidence through learned prototypes, benefiting Transformer-backed models with the self-explanatory characteristic. In future work, we plan to investigate the potential applications of ProtoPFormer in other critical areas like model debugging and medical image diagnosis.

Acknowledgments

This research was supported by "Pioneer" and "Leading Goose" R&D Program of Zhejiang (2024C01167, 2024C01114), Zhejiang Provincial Natural Science Foundation of China (LQ24F020020, LD24F020011), Zhejiang Province High-Level Talents Special Support Program "Leading Talent of Technological Innovation of Ten-Thousands Talents Program" (No. 2022R52046) and National Science and Technology Major Project (2022ZD0119103).

Contribution Statement

Mengqi Xue and Qihan Huang equally contribute to this work. Canghong Jin is the corresponding author.

References

[Abnar and Zuidema, 2020] Samira Abnar and Willem Zuidema. Quantifying attention flow in transformers. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4190–4197, Online, July 2020. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

[Böhle *et al.*, 2022] Moritz Böhle, Mario Fritz, and Bernt Schiele. B-cos networks: Alignment is all we need for interpretability. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10329–10338, 2022.

[Bontempelli *et al.*, 2022] Andrea Bontempelli, Stefano Teso, Fausto Giunchiglia, and Andrea Passerini. Concept-level debugging of part-prototype networks. *arXiv preprint arXiv:2205.15769*, 2022.

[Bychkov *et al.*, 2018] Dmitrii Bychkov, Nina Linder, Riku Turkki, Stig Nordling, Panu E Kovanen, Clare Verrill, Margarita Walliander, Mikael Lundin, Caj Haglund, and Johan Lundin. Deep learning based tissue analysis predicts outcome in colorectal cancer. *Scientific reports*, 8(1):1–11, 2018.

[Chefer *et al.*, 2021a] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 782–791, 2021.

[Chefer *et al.*, 2021b] Hila Chefer, Shir Gur, and Lior Wolf. Transformer interpretability beyond attention visualization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 782–791, 2021.

[Chen *et al.*, 2019] Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. This looks like that: deep learning for interpretable image recognition. *Advances in Neural Information Processing Systems*, 32, 2019.

[Chen *et al.*, 2022a] Xiangning Chen, Cho-Jui Hsieh, and Boqing Gong. When vision transformers outperform resnets without pre-training or strong data augmentations. In *International Conference on Learning Representations*, 2022.

[Chen *et al.*, 2022b] Zhiwei Chen, Changan Wang, Yabiao Wang, Guannan Jiang, Yunhang Shen, Ying Tai, Chengjie Wang, Wei Zhang, and Liujuan Cao. Lctr: On awakening the local continuity of transformer for weakly supervised object localization. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 36, pages 410–418, 2022.

[Donnelly *et al.*, 2022] Jon Donnelly, Alina Jade Barnett, and Chaofan Chen. Deformable protopnet: An interpretable image classifier using deformable prototypes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10265–10275, 2022.

[Dosovitskiy *et al.*, 2021] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[Gao and Guan, 2023] L. Gao and L. Guan. Interpretability of machine learning: Recent advances and future prospects. *IEEE MultiMedia*, 30(04):105–118, oct 2023.

[Gautam *et al.*, 2023] Srishti Gautam, Marina M.-C. Höhne, Stine Hansen, Robert Jenssen, and Michael Kampffmeyer. This looks more like that: Enhancing self-explaining models by prototypical relevance propagation. *Pattern Recognition*, 136:109172, 2023.

[Hu *et al.*, 2023] Brian Hu, Paul Tunison, Brandon Richard-Webster, and Anthony Hoogs. Xaitk-saliency: An open source explainable ai toolkit for saliency. *Proceedings of the AAAI Conference on Artificial Intelligence*, 37(13):15760–15766, Sep. 2023.

[Huang *et al.*, 2023] Qihan Huang, Mengqi Xue, Wenqi Huang, Haofei Zhang, Jie Song, Yongcheng Jing, and Mingli Song. Evaluation and improvement of interpretability for self-explainable part-prototype networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2011–2020, 2023.

[Keswani *et al.*, 2022] Monish Keswani, Sriranjani Ramakrishnan, Nishant Reddy, and Vineeth N Balasubramanian. Proto2proto: Can you recognize the car, the way i do? In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10233–10243, 2022.

[Khosla *et al.*, 2011] Aditya Khosla, Nityananda Jayadevaprakash, Bangpeng Yao, and Li Fei-Fei. Novel dataset for fine-grained image categorization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshop*, Colorado Springs, CO, June 2011.

[Kim *et al.*, 2022a] Sangwon Kim, Jaeyeal Nam, and Byoung Chul Ko. Vit-net: Interpretable vision transformers with neural tree decoder. In *International Conference on Machine Learning*, pages 11162–11172. PMLR, 2022.

[Kim *et al.*, 2022b] Sunnie SY Kim, Nicole Meister, Vikram V Ramaswamy, Ruth Fong, and Olga Russakovsky. Hive: evaluating the human interpretability of visual explanations. In *European Conference on Computer Vision*, pages 280–298. Springer, 2022.

[Krause *et al.*, 2013] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*, pages 554–561, 2013.

[Liu *et al.*, 2021] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10012–10022, 2021.

[Loshchilov and Hutter, 2019] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019.

[Nauta *et al.*, 2021] Meike Nauta, Ron van Bree, and Christin Seifert. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14933–14943, 2021.

[Pan *et al.*, 2021] Bowen Pan, Rameswar Panda, Yifan Jiang, Zhangyang Wang, Rogerio Feris, and Aude Oliva. Ia-red^2: Interpretability-aware redundancy reduction for vision transformers. *Advances in Neural Information Processing Systems*, 34:24898–24911, 2021.

[Raghu *et al.*, 2021] Maithra Raghu, Thomas Unterthiner, Simon Kornblith, Chiyuan Zhang, and Alexey Dosovitskiy. Do vision transformers see like convolutional neural networks? *Advances in Neural Information Processing Systems*, 34:12116–12128, 2021.

[Rigotti *et al.*, 2021] Mattia Rigotti, Christoph Miksovic, Ioana Giurgiu, Thomas Gschwind, and Paolo Scotton. Attention-based interpretability with concept transformers. In *International Conference on Learning Representations*, 2021.

[Russakovsky *et al.*, 2015] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, 2015.

[Selvaraju *et al.*, 2020] 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. *International Journal of Computer Vision*, 128(2):336–359, 2020.

[Serrano and Smith, 2019] Sofia Serrano and Noah A. Smith. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy, July 2019. Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics.

[Singla *et al.*, 2019] Sahil Singla, Eric Wallace, Shi Feng, and Soheil Feizi. Understanding impacts of high-order loss approximations and features in deep learning interpretation. In *International Conference on Machine Learning*, pages 5848–5856. PMLR, 2019.

[Subramanian *et al.*, 2018] Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. Spine: Sparse interpretable neural embeddings. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[Talaei Khoei *et al.*, 2023] Tala Talaei Khoei, Hadjar Ould Slimane, and Naima Kaabouch. Deep learning: Systematic review, models, challenges, and research directions. *Neural Computing and Applications*, 35(31):23103–23124, 2023.

[Touvron *et al.*, 2021a] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *International Conference on Machine Learning*, pages 10347–10357. PMLR, 2021.

[Touvron *et al.*, 2021b] Hugo Touvron, Matthieu Cord, Alexandre Sablayrolles, Gabriel Synnaeve, and Hervé Jégou. Going deeper with image transformers. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 32–42, 2021.

[Vaswani *et al.*, 2017] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in Neural Information Processing Systems*, 30, 2017.

[Wang *et al.*, 2021] Jiaqi Wang, Huafeng Liu, Xinyue Wang, and Liping Jing. Interpretable image recognition by constructing transparent embedding space. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 895–904, 2021.

[Welinder *et al.*, 2010] Peter Welinder, Steve Branson, Takeshi Mita, Catherine Wah, Florian Schroff, Serge Belongie, and Pietro Perona. Caltech-ucsd birds 200. Technical Report CNS-TR-201, Caltech, 2010.

[Yao *et al.*, 2022] Yuan Yao, Fang Wan, Wei Gao, Xingjia Pan, Zhiliang Peng, Qi Tian, and Qixiang Ye. Ts-cam: Token semantic coupled attention map for weakly supervised object localization. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–13, 2022.

[Yuan *et al.*, 2021] Tingyi Yuan, Xuhong Li, Haoyi Xiong, Hui Cao, and Dejing Dou. Explaining information flow inside vision transformers using markov chain. In *eXplainable AI approaches for debugging and diagnosis.*, 2021.

[Zarlenga *et al.*, 2023] Mateo Espinosa Zarlenga, Zohreh Shams, Michael Edward Nelson, Been Kim, and Mateja Jamnik. Tabcbm: Concept-based interpretable neural networks for tabular data. *Transactions on Machine Learning Research*, 2023.

[Zhang *et al.*, 2019] Quanshi Zhang, Yu Yang, Haotian Ma, and Ying Nian Wu. Interpreting cnns via decision trees. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 6261–6270, 2019.

[Zhang *et al.*, 2023] Haofei Zhang, Mengqi Xue, Xiaokang Liu, Kaixuan Chen, Jie Song, and Mingli Song. Schema inference for interpretable image classification. In *The Eleventh International Conference on Learning Representations*, 2023.