ASCENT-ViT: Attention-based Scale-aware Concept Learning Framework for Enhanced Alignment in Vision Transformers

Sanchit Sinha , **Guangzhi Xiong** and **Aidong Zhang**

University of Virginia

{sanchit, hhu4zu, aidong}@virginia.edu,

Abstract

As Vision Transformers (ViTs) are increasingly adopted in sensitive vision applications, there is a growing demand for improved interpretability. This has led to efforts to forward-align these models with carefully annotated abstract, human-understandable semantic entities - concepts. Concepts provide global rationales to the model predictions and can be quickly understood/intervened on by domain experts. Most current research focuses on designing model-agnostic, plug-and-play generic concept-based explainability modules that do not incorporate the inner workings of foundation models (e.g., inductive biases, scale invariance, etc.) during training. To alleviate this issue for ViTs, in this paper, we propose ASCENT-ViT, an attention-based, concept learning framework that effectively composes scale and position-aware representations from multiscale feature pyramids and ViT patch representations, respectively. Further, these representations are aligned with concept annotations through attention matrices - which incorporate spatial and global (semantic) concepts. ASCENT-ViT can be utilized as a classification head on top of standard ViT backbones for improved predictive performance and accurate and robust concept explanations as demonstrated on five datasets, including three widely used benchmarks (CUB, Pascal APY, Concept-MNIST) and two real-world datasets (AWA2, KITS). An appendix of the paper with more comprehensive results is available at https://arxiv.org/abs/2501.09221.

1 Introduction

With the surge of advanced deep learning (DL) methods, deep neural networks (DNNs), especially Transformer-based networks [Vaswani *et al.*, 2017], have been widely deployed in human communities for the benefit of our life [Dargan *et al.*, 2020; Shorten *et al.*, 2021]. Research in developing high-performing, generalizable, and efficient architectures has led to experts classifying a family of fundamental architectures as 'Foundation Models' [Bommasani *et al.*, 2021]. In computer vision, the most widely utilized foundation model, Vi-sion Transformers (ViT), shows great scalability and impressive performance in various downstream tasks, which promotes its use in real-world applications like autonomous driving [Ando *et al.*, 2023]. However, with the development and application of such advanced DNNs, there is a growing concern about their lack of interpretability leading to hesitation in applying the technology in high-stakes areas such as medical diagnosis, facial recognition, finance, etc. Given the current situation, several studies on the interpretability of DNNs have emerged [Li *et al.*, 2022].

Current research on DNN interpretability can be broadly categorized into two classes - dubbed **backward** and **forward** alignment [Gabriel, 2020]. *Backward* alignment is post-hoc interpretation, which explains already trained models. Specifically, some post-hoc methods assign relative importance scores to different features considered [Ribeiro *et al.*, 2016; Sundararajan *et al.*, 2017], while some other approaches will rank the input training samples according to their importance to prediction [Koh and Liang, 2017; Ghorbani and Zou, 2019]. On the other hand, *forward* alignment methods try to design and train intrinsically interpretable models. One such approach attempts to incorporate human-understandable concepts into model architecture and training processes. Concepts can be considered as shared abstract entities across multiple instances, which provide a general understanding of the task processed by the model. Many studies have explored how to integrate the model's decision-making process with task-relevant concepts to achieve transparent and interpretable model predictions [Koh *et al.*, 2020; Pedapati *et al.*, 2020; Jeyakumar *et al.*, 2020; Heskes *et al.*, 2020; Chen *et al.*, 2019]. Formally, given a set of concept representations C associated with an input sample x, the task prediction is $y = g(C)$, where g is any interpretable function used to aggregate concepts. Multiple works such as [Alvarez Melis and Jaakkola, 2018] and [Agarwal *et al.*, 2021] have postulated using a weighted linear sum of concepts as g. Other works such as [Koh *et al.*, 2020] have postulated using a shallow feed-forward network as mapping from concepts to prediction.

With the success of attention mechanism [Vaswani *et al.*, 2017] in designing large-scale networks for language [Devlin *et al.*, 2018] and vision applications [Dosovitskiy *et al.*, 2020], multiple diverse use-cases for attention have been proposed. A particular use case of attention in concept-based ex-

plainability is utilizing the attention mechanism in modeling the concept aggregation function g as first proposed in [Rigotti *et al.*, 2021]. Note that concepts are independent entities shared among different training samples, hence attention can be reliably used to assign "relevance" scores to the concepts relative to a prediction.

ViTs [Dosovitskiy *et al.*, 2020] demonstrate significant improvement over traditional models like CNNs for large-scale vision applications. ViTs borrow their architecture from transformer-based language approaches [Vaswani *et al.*, 2017; Devlin *et al.*, 2018]. A ViT first segments an image into equal patches, which are projected into an embedding space with positional information. The representation is then passed through stacked transformer blocks which is used in downstream tasks. ViTs have demonstrated better performance than CNNs, especially when pre-trained with large-scale data. However, due to the very architecture of ViTs being borrowed from language paradigm, they lack vision-specific *inductive biases* like CNNs. CNNs have been specifically designed for image modalities, and capture explicit inductive biases such as - scale invariance and transformation equivariance. As an example, ViTs are demonstrated not to be scale-insensitive [Xu *et al.*, 2021], a known strength of CNNs. Researchers have attempted to improve ViTs by incorporating CNN-like inductive biases in their architectures with great success [Liu *et al.*, 2021; Graham *et al.*, 2021].

However, concept-based explainability approaches are still designed to be model-agnostic [Koh *et al.*, 2020] and plug-and-play [Rigotti *et al.*, 2021] to fit all possible use cases. As incorporating various inductive biases in the model architecture provides significant performance improvement, a detailed design decision should be made for concept-based explainability modules as well. For instance, explainability modules for ViTs can benefit from scale awareness in addition to patch awareness during concept learning in ViTs. As a consequence, we design a novel framework for ViTs that incorporates desirable properties (inductive biases) of ViTs - patch awareness and CNNs - scale invariance. Specifically, in this paper, we propose **Attention-based Scale-aware Concept Learning Framework for Enhanced Alignment in Vision Transformers (ASCENT-ViT)**, a concept-based explainability module for ViTs to effectively *align* human-annotated concepts with model representations. Unlike model-agnostic plug-and-play explainability, ASCENT-ViT effectively composes inductive biases of CNNs and ViTs - scale invariance and image-patch relationship awareness respectively into scale and patch-aware representations. More precisely, ASCENT-ViT consists of 3 distinct modules - (i) **Multi-scale Encoding (MSE) Module**, which models concepts at various scales, (ii) **Deformable Multi-Scale Fusion (DMSF) Module**, which composes multi-scale concepts with patch embeddings, and (iii) **Concept-Representation Alignment Module (CRAM)**, which aligns concepts with the learned model representations. To summarize, our contributions are:

- We propose ASCENT-ViT, a concept-based explainability method that effectively composes inductive biases from CNNs and ViTs to perform better concept-representation alignment. We are the first work to effec-

tively integrate these inductive biases in concept-based explainability modules.

- We demonstrate through quantitative experiments that ASCENT-ViT improves both prediction accuracy and concept learning as compared to model-agnostic attention-based concept explainability modules across 7 different vision-transformer architectures.

- We demonstrate through diverse real-world visual examples that ASCENT-ViT captures concept annotations missed by model-agnostic approaches and is robust to transformations and perturbations.

2 Related Work

Vision Transformers (ViTs) and Inductive biases. Multiple approaches have improved the original ViT [Dosovitskiy *et al.*, 2020] performance [Atito *et al.*, 2021], efficiency [Chen *et al.*, 2022] and explainability [Xu *et al.*, 2023]. Even though ViTs offer better downstream task performance as compared to CNN-centric approaches [Tuli *et al.*, 2021; Raghu *et al.*, 2021], the consensus among researchers remains that ViTs have much less stringent inductive biases and some approaches improve them [Xu *et al.*, 2021]. To alleviate this problem, namely - lack of scale invariance and transform equivariance, DETR [Zhu *et al.*, 2020] proposes an attention mechanism to incorporate scale features while SWIN [Liu *et al.*, 2021] uses a shifted window approach in ViTs with great success. A few approaches have improved ViT architectures with scale-awareness [Lin *et al.*, 2023; Guan *et al.*, 2024].

Concept-based Explainability and Attention. Concept-based explanations explain models using human-understandable concepts [Koh *et al.*, 2020; Sinha *et al.*, 2023; Sinha *et al.*, 2024a]. Concepts are high-level attributes shared across different instances [Chen *et al.*, 2019]. Explorations uncover the decision-making of neural networks via concepts, including learning latent concept scores [Alvarez Melis and Jaakkola, 2018], finding concept prototypes [Li *et al.*, 2018; Chen *et al.*, 2019], and activations [Sinha *et al.*, 2024b].

Comparison with related approaches. Recent approaches incorporate an external Attention matrix to align model representations with human-understandable concepts. Two related works to our method ASCENT-ViT are - Concept Bottleneck Models (CBMs) [Koh *et al.*, 2020], CT [Rigotti *et al.*, 2021], and BotCL [Wang *et al.*, 2023]. Both CBMs and BotCL utilize an intermediate layer to perform alignment between concepts and representations but neither utilize spatial concept annotation maps - CBMs discretize the concepts to binary labels while BotCL is completely unsupervised and utilizes slot attention to learn representative concepts. For completeness, we use CBM as one of the baselines to compare our method's performance. The closest work to ASCENT-ViT is CT [Rigotti *et al.*, 2021], which utilizes a model-agnostic plug-and-play module to align concepts with representations. However, there are significant differences between our approach and CT. Firstly, their module is model-agnostic and does not account for the inductive biases. Secondly, their method does not incorporate the effect of scale - making it fragile to minor transformations. Finally, their method does not provide any

analysis on its effectiveness over ViT scale and architecture - a very important determinant of the extent of misalignment.

3 Methodology

In this section, we introduce our proposed *Attention-based Scale-aware Concept Learning Framework for Enhanced Alignment in Vision Transformers (ASCENT-ViT)*. ASCENT-ViT is composed of three independent modules, namely - the Multi-scale encoding (MSE) module which extracts multi-scale features for enhanced scale awareness, the Deformable Multi-Scale Fusion (DMSF) Module which incorporates image scales using Deformable Attention, and the Concept-Representation Alignment Module (CRAM) which aligns scale and patch aware features with concept annotations. The ASCENT-ViT can be utilized as a classification head on top of standard ViT backbones for improved concept-based explanations. The overall schematic representation of the ASCENT-ViT is shown in Figure 1.

Problem Setting: A concept-based model consists of networks f and g. For a given training sample $\{(x_i, y_i, C_i)\}$, where $x_i \in \mathbb{R}^D$ denotes the i-th training sample, $y_i \in \mathbb{R}$ is the target classification label for sample x_i, and $C_i \in \mathbb{R}^T$ is a vector of T human annotated concepts, the network f learns a mapping from image space to the concept space, i.e., $f : \mathbb{R}^D \to \mathbb{R}^T$, while the network g maps concepts to the prediction space, i.e., $g : \mathbb{R}^T \to \mathbb{R}$. The training procedure entails supervising the concept space with human-annotated concepts and promoting the alignment of concepts and model representations. In our method, the concepts C_i are the concept maps with pixel-level annotations.
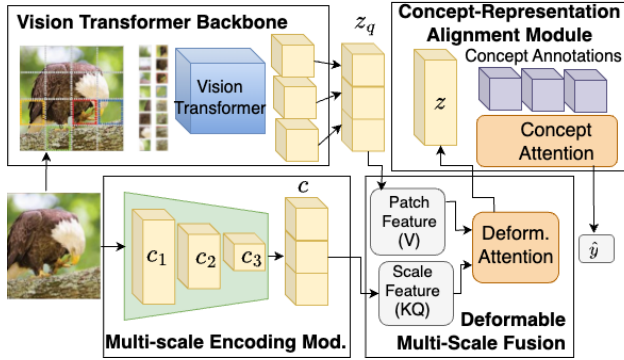

Figure 1: Schematic overview of the proposed ASCENT-ViT module. Given an input image, the Multi-scale encoding (MSE) module encodes representations at various scales in **c**. Patch-aware representations z_q from the ViT are composed using the Deformable Multi-Scale Fusion (DMSF) Module. The Concept-Representation Alignment Module (CRAM) aligns the learned representations **z** with concept annotations C. $\hat{y}$ is the model estimation of the true task label y.

3.1 Multi-scale Encoding Module

The Multi-scale encoding Module consists of a Convolutional Neural Network (CNN) **F** which can be thought of as S stacked convolution blocks (with implicit activation and pooling operations) f_i, such that $\mathbf{F} = f_S \circ f_{S-1} \circ f_{S-2} \dots, \circ f_1$. CNNs embody inductive biases in the form of convolution
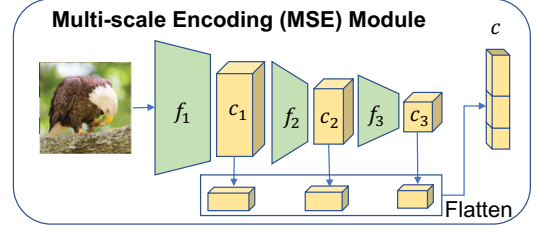
(a) The Multi-scale encoding (MSE) Module.
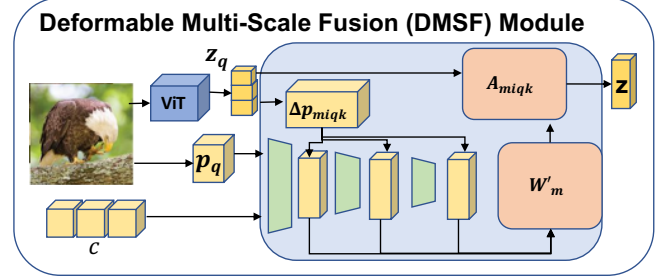
(b) The Deformable Multi-Scale Fusion (DMSF) Module.

Figure 2: (a) Detailed view of the MSE module which extracts multi-scale features $\{c_i\}_1^S$ and concatenates them together into a vector **c**. (b) Detailed view of the proposed DMSF module utilizing Deformable Attention operation to combine multi-scale features **c** with patch embeddings from ViT (z_q) to output scale and patch-aware representations **z**.

operations which capture the *local semantics* of images at various scales, i.e., each block computes feature maps representing various scales. As a consequence, for a given input image $x \in R^D$, we extract S multiscale features from various blocks of **F**. Mathematically, each feature for a particular scale can be computed as

$$c_i = f_i \circ f_{i-1} .. \circ f_1(x), \ i \in \{1, \cdots, S\} \qquad (1)$$

Subsequently, the combined multiscale features are concatenated by the MSE module to output a combined and spatially aware multi-scale feature **c**. Mathematically, the scale-aware feature **c** is given as,

$$\mathbf{c} = \{\text{flatten}(c_1), \text{flatten}(c_2), .. \text{flatten}(c_S)\} \qquad (2)$$

Note that the feature **c** is sometimes called the feature pyramid [Lin *et al.*, 2017; Chen *et al.*, 2022] because it intuitively represents multiple scales of the same image, each scale progressively smaller than the previous scale. The architecture of the MSE module is presented in Fig. 2 (a).

3.2 Deformable Multi-Scale Fusion Module

The DMSF module consists of two distinct steps. The first step computes an attention operation using deformed (scaled) outputs **c** from the MSE module with the Multi-Scale Deformable Attention (MSDA) operation [Zhu *et al.*, 2020] as discussed below. The second step composes a list of patch outputs z_q from a ViT backbone with multi-scale features computed from the MSDA operation using a combination of a learnable parameter (**I**) and a tunable hyperparameter (ψ).

Multi-scale Deformable Attention (MSDA) Operation. As discussed before, the fundamental problem in the standard

attention mechanism used in ViTs is the lack of scale and spatial awareness. To incorporate multi-scale feature maps, [Zhu et al., 2020] proposed MSDA. The MSDA operation is characterized by a set of reference points (depicted by coordinates $\mathbf{p_q}$ and their offsets from a reference point $\Delta \mathbf{p}$) shared among features corresponding to multiple scales of the same image's feature maps which localizes semantics across scales. Mathematically, the MSDA operation for a set of feature-maps of various scales $\{c_i\}_1^S$ over a query-sample $\mathbf{z_q}$, the $MSDA(\mathbf{z_q}, \mathbf{p_q}, \mathbf{c})$ operation can be represented as,

$$\sum_{m=1}^{M} \mathbf{W}_m [\sum_{i=1}^{S} \sum_{k=1}^{K} \mathbf{A_{miqk}} \cdot \mathbf{W'}_m c_i(\phi_i(\mathbf{p_q}) + \Delta \mathbf{p_{miqk}})] \tag{3}$$

where M is the number of attention heads (indexed by m), S is the number of scales (indexed by i), K is the number of sampled reference points (indexed by k). $\mathbf{W}$ projects Value vector to multiple heads while $\mathbf{W'}$ is the inverse projection. $\mathbf{A}$ denotes the attention matrix, and $\Delta \mathbf{p_{miqk}}$ represents offset corresponding to sampled reference points $\mathbf{p_q}$, for each head and scale. Correspondingly, each element in $\mathbf{A}$ is referenced as head (m) and scale (i), while q and k represent the query and key indices, implying the number of reference points sampled is $S * K$. The function ϕ scales up/down the reference points to corresponding scales as c_i. Intuitively, the module can be understood as utilizing multi-scale features as key/value vectors that transform a patch-aware vector ($\mathbf{z_q}$) into a patch and scale-aware vector ($\mathbf{z}$).

Adaptive Patch Composition. Note that the output from the ViT backbone is represented as $\mathbf{z_q}$ (Figure 1), which is treated as the query vector for the MSDA operation. Even though the MSDA operation introduces scale awareness in the outputs of the ViT backbone, a tradeoff between patch-specific semantics and scale awareness is desirable due to the wide gulf in the inductive biases in them. To alleviate this, we introduce a learnable vector $\mathbf{I}$ which controls the contribution of scale features, and a scalar hyperparameter ψ which weighs the effect of the MSDA operation on the final patch vector. Mathematically, the final output of the DMSF module is as follows:

$$\mathbf{z} = norm(\mathbf{z_q} + \psi \cdot \mathbf{I} \cdot MSDA(\mathbf{z_q}, \mathbf{p_q}, \mathbf{c})) \tag{4}$$

where $norm$ represents the layer norm, $\mathbf{z_q}$ represents the input from the ViT backbone, $\mathbf{p_q}$ represents the reference points while $\mathbf{c} = \{c_i\}_1^S$ represents the multi-scale features computed by the MSE module. Note that adaptive composition is sensitive to the initialization of the $\mathbf{I}$ vector and the tunable hyperparameter ψ. An overview of the DMSF module is shown in Fig. 2 (b).

3.3 Concept-Representation Alignment Module

The Concept-Representation Alignment Module (CRAM) is inspired by the Attention mechanism proposed in [Vaswani et al., 2017] wherein, an attention matrix $\mathbf{A_{att}}$ computation is conditioned on three vectors of the same dimensions - Query (Q), Key (K), and Value (V). Mathematically, the attention operation is as follows:

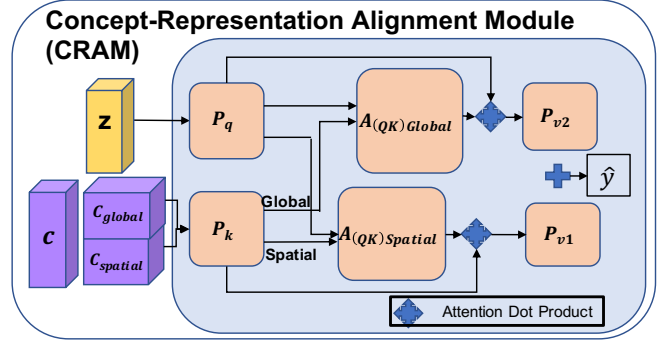$$\mathbf{Att}(Q, K, V) = softmax(\frac{QK^T}{\sqrt{dim}})V = \mathbf{A_{att}}V \tag{5}$$

Figure 3: Detailed view of the Concept-Representation Alignment Module (CRAM). CRAM aligns scale and patch-aware representations ($\mathbf{z}$) and human-annotated concepts (C). The matrices P_q, P_k and P_{v1},P_{v2} are projection matrices while $\mathbf{A_{Spatial}}$ and $\mathbf{A_{Global}}$ are attention matrices between concept projections and patch embedding projections. The final prediction $\hat{y}$ is the average of outputs between spatial and global attention operations.

The attention matrix attends to each value in the Query and Key vectors with a composite score. CRAM treats the Query vector as the patch representation $\mathbf{z}$ from the DMSF module and the Key Vector as Human-annotated Concepts C. As patches and concepts are continuous and one-hot respectively, projection matrices convert them to continuous vectors. Mathematically, the attention matrix for patch embeddings ($\mathbf{z}$) and concepts (C) associated to an input sample x is calculated as follows:

$$\mathbf{A_{QK}} = softmax(\frac{(\mathbf{z}P_q)(CP_k)^T}{\sqrt{dim}}) \tag{6}$$

where the matrices P_q and P_k are learnable projection matrices transforming patches and concepts to Query and Key vectors respectively, while dim represents the dimension of the patch and concept embeddings. Intuitively, each entry in the matrix $\mathbf{A_{QK}}$ encapsulates the contribution of a concept on a patch and is a measure of alignment. The final prediction is then calculated by multiplying the attention matrix with concept projections ($V = CP_k$) followed by a final learnable projection matrix (P_v) to map to logits. Mathematically each logit $\hat{y}$ in the prediction $\hat{y}$ can be calculated as an average of concept contributions on each patch.

$$\hat{y} = \frac{1}{|\mathbf{z}|} \sum^{|\mathbf{z}|} \mathbf{A_{QK}}VP_v \tag{7}$$

where $|\mathbf{z}|$ stands for the dimension of $\mathbf{z}$. In practice, two different attention matrices ($\mathbf{A_{Spatial}}$ and $\mathbf{A_{Global}}$ - the QK subscript is omitted for clarity) are calculated for different subsets of concepts - spatial and global. The spatial concepts can be localized, while the global concepts are more semantic and cannot be localized. The final output is then the sum of logits computed from both spatial and global attention matrices. Mathematically the sum can be written as:

$$\hat{y} = \frac{1}{|\mathbf{z}|} \sum^{|\mathbf{z}|} \mathbf{A_{Spatial}}VP_{v1} + \frac{1}{|\mathbf{z}|} \sum^{|\mathbf{z}|} \mathbf{A_{Global}}VP_{v2} \tag{8}$$

where P_{v1} and P_{v2} are different versions of P_v associated with $\mathbf{A_{Spatial}}$ and $\mathbf{A_{Global}}$, respectively. The detailed description of CRAM is shown in Figure 3.

3.4 End-to-end Training of ViT with ASCENT-ViT

The overall schematic pipeline of our approach is depicted in Figure 1. As MSE, DMSF and CRAM modules are standalone modules that can be appended on top of any standard ViT backbone. To sum up, the inference step for an input image x utilizing ASCENT-ViT computes the patch embeddings $\mathbf{z_q}$ through the following procedure:

$$\hat{y} = CRAM(\mathbf{z}, C), \quad \text{where } \mathbf{z} = DMSF(\mathbf{z_q}, \mathbf{c}), \mathbf{c} = MSE(x) \tag{9}$$

Training Objective. The final training objective includes the task prediction objective utilizing a classification loss. For datasets with explicit localization of concepts in the image (e.g. CUB), the attention matrix $\mathbf{A_{Spatial}}$ can be regularized by human-annotated localizations. Mathematically, the training objective can be given as:

$$\mathcal{L}_{ASCENT-ViT} = \mathcal{L}(\hat{y}, y) + \lambda ||\mathbf{A_{Spatial}} - H||_F \tag{10}$$

where $\mathcal{L}$ represents any classification loss, H denotes human-annotated localization points for the concepts, λ is a tunable hyperparameter and the subscript F is the Frobenius norm.

4 Experiments and Results

4.1 Dataset Description

(1) CUB200 [Wah *et al.*, 2011]: The Caltech-UCSD Birds-200-2011 dataset consists of 11,788 photos of 200 different classes of birds annotated with concepts representing physical traits of birds like wing color, beak size, etc. **(2) AWA2** [Xian *et al.*, 2018]: Animals with Attributes-2 consists of 37,322 images of a combined 50 animal classes with 85 binary concepts like number of legs, presence of tail, etc. We utilize Segment Anything Model (v2)[1] to annotate 10 selected concepts. **(3) KITS** [Heller *et al.*, 2023]: The Kidney Tumor Segmentation (KITS) challenge dataset consists of images of kidneys where the task is segmentation of renal cysts and tumors. We augment each image with the concepts from the metadata, namely - kidney, tumor, and cysts. We sample 10000 images with an 80:20 train/test split. **(4) PascalAPY** [Farhadi *et al.*, 2009]: We utilize Pascal Object Recognition Dataset as processed and described in [Rigotti *et al.*, 2021]. After processing, it consists of 14350 images with 20 attributes each. **(5) C-MNIST** [Sinha *et al.*, 2023]: Concept-MNIST entails augmenting MNIST dataset with two "spatial" concepts in the form of curved and straight lines. We consider two tasks - even/odd and digit classification.

4.2 Baselines and Evaluation Metrics

Comparision Baselines. As discussed in Related Work, our method is the first to utilize inductive biases in attention-based concept explainability. The closest comparison to our method is CBM [Koh *et al.*, 2020] (CBMs do not utilize concept attention maps, only global concepts) and [Rigotti *et al.*, 2021] which only utilizes an inferior attention-only architecture similar to only a CRAM module. **We denote [Rigotti *et al.*, 2021] as "Only CRAM"**. We compare our approach on a variety of ViT baselines - with ViT-Base, ViT-Large [Dosovitskiy *et al.*, 2020], SWIN [Liu *et al.*, 2021] for all datasets.

[1]https://github.com/facebookresearch/sam2

ViT-Base and ViT-Large are chosen to demonstrate the effect of scale awareness, while SWIN is chosen to demonstrate the effect of inductive bias in the backbone. In addition, we also compare the CUB dataset on DeIT [Touvron *et al.*, 2021], ViT-Base (Dino) [Caron *et al.*, 2021]. SWIN incorporates inductive biases from CNNs. Dino is a robustly trained ViT while DeIT is a distilled version, further capturing the diversity in training procedures.

Evaluation Metrics. The evaluation metric for task prediction is the accuracy in the five datasets. For datasets with spatial concept annotations, namely CUB, AWA2, and KITS, we utilize the Pixel True Positive Rate (Px. TPR) which measures the percentage of correctly identified concept annotations. Due to the extremely precise and sensitive setting of KITS dataset, we measure the total pixel accuracy which encompasses both accurately identified concept and nonconcept pixels. For datasets with no spatial concept annotations (only global) - namely C-MNIST and Pascal APY, we utilize the 0-1 error (misclassification) for concepts, as only binary concepts are considered. Additionally, for CUB200, as localization information for each concept is present, the concept errors are directly evaluated with the Frobenius norm of the difference between ground-truth annotations and predicted concept attention matrices.

4.3 Implementation Details

ViT Backbone. We set the patch size to correspond to 16x16 pixels. Each image is resized to (224,224), and hence a patch sequence is (224/16, 224/16) = (14,14). Following [Rigotti *et al.*, 2021], we append the <CLS>token to the patch sequence after positional embeddings encapsulating global semantics. This results in an input sequence of length of 196+1 = 197 tokens. The internal embedding size (dim) of ViT is 1024.

MSE Module. We utilize three scales in the Multiscale Encoding Module, i.e., $S = 3$. We utilize scales 1/8, 1/16, and 1/32 corresponding to scale-aware vector $\mathbf{c}$ composed of sizes: $c_1 = 1024$, $c_2 = 196$, and $c_3 = 16$, making the size of $\mathbf{c} = 1029$. The first convolution block consists of three convolution layers with batch norm followed by Max pooling. The following convolution blocks consist of a single convolution layer of kernel size=3 and stride=2 with batch norm.

DMSF Module. We utilize Multiscale Deformable Attention [Zhu *et al.*, 2020]. We used 16 attention heads and four reference points, along with layer norms for each key, query, and value vector. We strip <CLS>token before passing through the DMSF module and append it to the output. The initialization value of $\mathbf{I}$ is set as 0.01. The value of ψ is tunable and set as 1 for CUB, 2 for Concept-MNIST and Pascal aPY, and 0.5 for AWA2 and KITS.

CRAM Module. We utilize the same embedding dimensions, i.e., $dim = 1024$ for Key, Query and Value vectors. The number of attention heads is set as 2 for both global and spatial concept attention matrices.

Training Details. We train each dataset and backbone for 50 epochs with early stopping. The maximum learning rate is set at 5e-5 with a linear warmup for the first 10 epochs followed by Cosine decay. The batch size is 16 for each dataset with mixed precision (16-bit default) optimization.

Head	Backbone	CUB200		AWA2		KITS	Concept-MNIST		Pascal aPY
		Accuracy	Px. TPR	Accuracy	Px. TPR	Px. Acc.	Class.	Odd/Even	Accuracy
CBM	ViT-Large	81.03	-	53.14	-	-	93.83	97.12	81.1
CRAM (CT)	ViT-Base	84.41 ± 0.3	82.25 ± 0.3	62.3 ± 0.2	64.02 ± 0.5	91.89 ± 2.4	95.71 ± 0.1	97.58 ± 0.1	81.0 ± 0.8
	ViT-Large	86.31 ± 0.2	83.55 ± 1.1	63.6 ± 0.1	68.04 ± 1.4	94.33 ± 2.0	95.74 ± 0.1	97.94 ± 0.1	81.3 ± 0.8
	SWIN	85.71 ± 0.1	82.28 ± 0.8	62.3 ± 0.1	67.84 ± 0.7	92.81 ± 1.8	94.83 ± 0.1	97.68 ± 0.1	81.3 ± 0.8
ASCENT-ViT	ViT-Base	85.71 ± 0.3	83.93 ± 0.2	62.3 ± 0.1	64.11 ± 1.2	92.33 ± 0.8	$\mathbf{95.83 \pm 0.1}$	97.91 ± 0.1	81.4 ± 0.8
	ViT-Large	$\mathbf{87.26 \pm 0.3}$	$\mathbf{87.21 \pm 0.9}$	$\mathbf{63.7 \pm 0.1}$	$\mathbf{73.12 \pm 0.3}$	$\mathbf{95.84 \pm 1.9}$	$\mathbf{95.83 \pm 0.1}$	$\mathbf{97.91 \pm 0.1}$	$\mathbf{81.9 \pm 0.8}$
	SWIN	85.74 ± 0.1	83.41 ± 1.3	62.5 ± 0.4	69.66 ± 0.4	93.02 ± 1.3	94.83 ± 0.1	97.91 ± 0.1	81.4 ± 0.8

Table 1: We report Accuracy and True Positive Rate (TPR) for CUB200 and AWA2 datasets, over backbones ViT-Base, ViT-Large, and SWIN augmented with CRAM (Rows 2-4) and ASCENT-ViT augmented backbones (Rows 5-7). For the KITS dataset, we report the total pixel accuracy. For Concept-MNIST, we report accuracy on two tasks - Digit classification and Odd/Even detection; for Pascal aPY, we report both accuracy on object detection tasks. All results are averaged over 3 seeds, and std dev is reported.

4.4 Quantitative Results

Task Prediction Performance

Table 1 reports the task prediction performance across all five datasets with ViT backbones using CBM (Row-1), augmented with only CRAM (Rows 2-4) and ASCENT-ViT (Rows 5-7). Rows 2-4 list the prediction performance of backbones augmented with only the CRAM module on - ViT-Base, ViT-Large, and SWIN. Rows 5-7 list the performance on the same architectures using ASCENT-ViT. ASCENT-ViT outperforms CBMs and only CRAM on all datasets. Interestingly, on SWIN backbones the performance of ASCENT-ViT and 'only CRAM' is on par - implying that SWIN captures scale inductive biases better than ViTs but performs worse than ViT-Large (a larger model). These results demonstrate that both size and inductive biases of ViTs are important factors in improving prediction and concept performances. Our module is effective in introducing scale awareness in standard ViTs and improving their performance.

Concept Annotation Performance

As concept performance metrics are different for datasets with spatial concept annotations and global annotations, we discuss the results individually as reported in Table 1.

True Positive Rate (TPR). For CUB200 and AWA2, we observe that pixel-wise TPR improves by ∼4% on CUB200 and ∼5% implying that ASCENT-ViT captures pixels of interest much better than only CRAM. In Fig 6, we demonstrate the effect on random test images where ASCENT-ViT captures concept annotations more effectively than only CRAM.

Pixel Accuracy. For KITS, we report total pixel accuracy which includes both concept and background annotations. Note that due to the sensitive nature of KITS, it is important to correctly identify both relevant (cysts) and irrelevant concepts (kidney surface). We observe ASCENT-ViT outperforms only CRAM by ∼1.5% over all architectures on KITS.

Misclassification Errors. For datasets with only global concepts and no spatial annotations, we report the misclassification error. We observe ASCENT-ViT outperforms only CRAM on both C-MNIST and Pascal aPY.

4.5 Ablation Studies

Hyperparameter Ablations

We demonstrate the effect of the most important hyperparameters in the DMSF module in Figure 4. For all experiments, a ViT-Base model is utilized and trained on the CUB dataset.
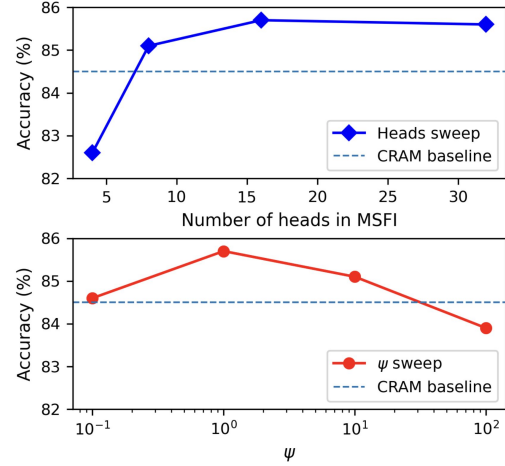

Figure 4: Effect of most relevant hyperparameters of DMSF module.

Figure 4 (TOP) demonstrates the effect of increasing the number of attention heads on the prediction performance. We observe that the increased number of attention heads improves performance by attending to various scales effectively. However, the maximum performance is observed at 16 heads implying that too many heads might result in the model focusing on non-relevant parts. Next, Figure 4 (BOTTOM) demonstrates the predictive performance as a function of increasing values of ψ (i.e., increasing weight of scale-aware features on **z**). Here, we observe that a tradeoff exists between ψ and predictive performance. The reason for this behavior is with higher ψ, scale-aware inductive biases begin dominating the patch embeddings and performance gravitates towards models with high CNN-like inductive biases (Table 1).

Modules	Concept-MNIST		CUB200	AWA2
	Odd/Even	Class.	Class.	Class.
ViT-base	97.12	93.82	82.91	61.7
w/ CRAM	97.78	95.71	84.41	62.1
w/ CRAM+MSE	97.87	95.65	85.17	62.7
ASCENT-ViT	**97.88**	**95.84**	**85.73**	**63.4**

Table 2: Task prediction performance ablating on proposed modules of ASCENT-ViT over a single run on tasks. NOTE: ASCENT-ViT implies Backbone w/ MSE+DMSF+CRAM.

Module Ablations

Next, we progressively demonstrate each submodule's effect in Table 2. Line-1 lists the performance of ViT-base with no concept information. Next, Line-2 lists the performance only using the CRAM module. Line-3 lists the performance with
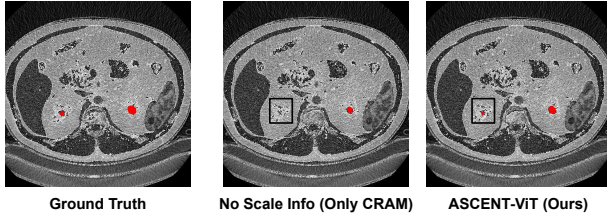
Ground Truth   **No Scale Info (Only CRAM)**   **ASCENT-ViT (Ours)**

Figure 5: Effect of Scale: Visualized concept annotations for a correctly classified sample from KITS. As seen, ASCENT-ViT identifies <u>small</u> 'cyst' annotation much more accurately (Acc=91.2%) than only CRAM (Acc=88.4%) where no scale information is utilized.

both the MSE and CRAM modules. As we observe, adding scale-aware features indeed improves performance. The last line (Line-5) lists performance when the DMSF module performs feature fusion - performing the best.

Model Name	Inductive Bias	Only CRAM	ASCENT-ViT
ViT-Large	Positional (Pos.)	86.31 ± 0.2	**87.26 ± 0.3**
DeIT-Base	Positional (Pos.)	76.81 ± 0.8	**76.96 ± 0.3**
SWIN (1x1)	CNN-like	86.44 ± 0.8	**86.72 ± 0.6**
ViT-Base (Dino)	Pos. + Invariance	76.5 ± 0.2	**77.2 ± 0.1**

Table 3: Performance on various ViTs on CUB200 using only CRAM and ASCENT-ViT. We observe that architectures with CNN-like inductive biases show smaller improvements and have comparable performance for both ASCENT-ViT and CRAM.

ViT Backbone Ablations

Table 3 lists the prediction performance of DeIT-Base, SWIN 1x1, and ViT-Base (Dino), variants on CUB200 averaged over three seeds. We observe that ASCENT-ViT outperforms CRAM on models with either a missing Positional or CNN-like inductive bias (DeIT, SWIN, ViT-Base (Dino)). This observation is expected as ASCENT-ViT provides explainability which combines both Positional and CNN-like scale-invariant inductive biases. Note that on SWIN, which contains pre-encoded CNN-like inductive biases, ASCENT-ViT is on par with CRAM. This observation reaffirms the motivation of our approach - incorporating multiple inductive biases in the explainability module provides better explanations.

Overhead of Explanations. We also provide the overhead of the additional modules as a percentage of model parameters. In most cases, the models augmented with CRAM consist of $\simeq$2% additional parameters than just the backbone implying a negligible increase in the training times.

4.6 Discussion

Figure 6 shows two examples of correctly classified images from CUB200 and one from KITS. The first row (Ground Truth) represents the ground-truth human annotations of the concepts. The second and third rows represent the explanations produced using models trained using only CRAM and ASCENT-ViT on the ViT-Base backbone, respectively. We observe that ASCENT-ViT can better capture regions 'missed' by only CRAM - the brown edge of wings in the first sample and beak shape in the second as compared to ground truth. All the pixels of the cysts in the kidney image are better captured by ASCENT-ViT as compared to only CRAM which misses relevant pixels in the middle of the cyst.
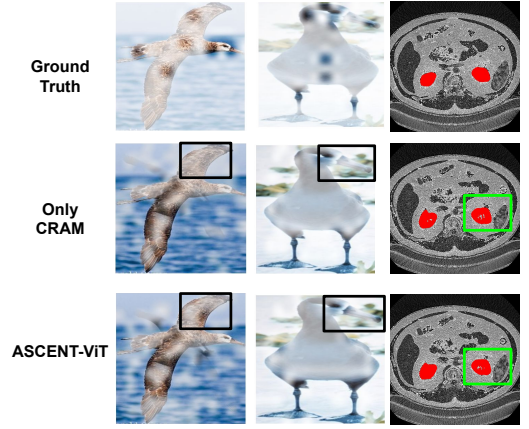

Figure 6: Visualizing concept attention scores over two samples from test set of CUB200 and one sample from KITS. We highlight the area of interest with a bounding box. (LEFT) ASCENT-ViT captures the brown front and back edge of wings better than utilizing only CRAM (i.e., assigns higher attention weights). (CENTER) ASCENT-ViT captures the annotations over the beak (tip) and eyes better than CRAM. (RIGHT) ASCENT-ViT captures the cyst annotation much more accurately as CRAM ignores some inner patches.
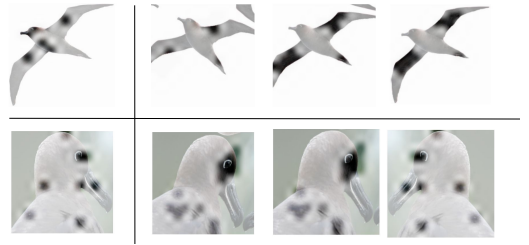

Figure 7: Robustness of concept explanations generated by ASCENT-ViT with transformations on test images as compared to ground truth (LEFT). Row-1 shows random rotations and crops across the center. Row-2 shows random flips and rotations.

Robustness to transformations. We also report the qualitative performance of ASCENT-ViT in small transformations in Figure 7 - rotation, random cropping, and zoom.

5 Conclusion

In this paper, we propose ASCENT-ViT, a concept-based explainability framework that aligns concepts with representations. ASCENT-ViT captures concepts across multiple scales using a Multi-scale Encoding Module and composes effective multiscale concept-patch relationships through a Multi-scale Feature Interaction Module. Finally, composite representations are aligned with concepts using a Concept-Representation Alignment Module. Quantitative and qualitative results demonstrate superior performance over existing methods on multiple ViT backbones. We hope our work helps in designing robust concept-based explainability modules for large-scale DNNs.

Acknowledgements

References

[Agarwal *et al.*, 2021] Rishabh Agarwal, Levi Melnick, Nicholas Frosst, Xuezhou Zhang, Ben Lengerich, Rich Caruana, and Geoffrey E Hinton. Neural additive models: Interpretable machine learning with neural nets. *Advances in neural information processing systems*, 34:4699–4711, 2021.

[Alvarez Melis and Jaakkola, 2018] David Alvarez Melis and Tommi Jaakkola. Towards robust interpretability with self-explaining neural networks. *Advances in neural information processing systems*, 31, 2018.

[Ando *et al.*, 2023] Angelika Ando, Spyros Gidaris, Andrei Bursuc, Gilles Puy, Alexandre Boulch, and Renaud Marlet. Rangevit: Towards vision transformers for 3d semantic segmentation in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5240–5250, 2023.

[Atito *et al.*, 2021] Sara Atito, Muhammad Awais, and Josef Kittler. Sit: Self-supervised vision transformer. *arXiv preprint arXiv:2104.03602*, 2021.

[Bommasani *et al.*, 2021] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[Caron *et al.*, 2021] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 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.*, 2022] Zhe Chen, Yuchen Duan, Wenhai Wang, Junjun He, Tong Lu, Jifeng Dai, and Yu Qiao. Vision transformer adapter for dense predictions. *arXiv preprint arXiv:2205.08534*, 2022.

[Dargan *et al.*, 2020] Shaveta Dargan, Munish Kumar, Maruthi Rohit Ayyagari, and Gulshan Kumar. A survey of deep learning and its applications: a new paradigm to machine learning. *Archives of Computational Methods in Engineering*, 27:1071–1092, 2020.

[Devlin *et al.*, 2018] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[Dosovitskiy *et al.*, 2020] 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 *International Conference on Learning Representations*, 2020.

[Farhadi *et al.*, 2009] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *2009 IEEE conference on computer vision and pattern recognition*, pages 1778–1785. IEEE, 2009.

[Gabriel, 2020] Iason Gabriel. Artificial intelligence, values, and alignment. *Minds and machines*, 30(3):411–437, 2020.

[Ghorbani and Zou, 2019] Amirata Ghorbani and James Zou. Data shapley: Equitable valuation of data for machine learning. In *International Conference on Machine Learning*, pages 2242–2251. PMLR, 2019.

[Graham *et al.*, 2021] Benjamin Graham, Alaaeldin El-Nouby, Hugo Touvron, Pierre Stock, Armand Joulin, Hervé Jégou, and Matthijs Douze. Levit: a vision transformer in convnet's clothing for faster inference. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 12259–12269, 2021.

[Guan *et al.*, 2024] Bin Guan, Jinkun Yao, and Guoshan Zhang. An enhanced vision transformer with scale-aware and spatial-aware attention for thighbone fracture detection. *Neural Computing and Applications*, pages 1–14, 2024.

[Heller *et al.*, 2023] Nicholas Heller, Fabian Isensee, Dasha Trofimova, Resha Tejpaul, Zhongchen Zhao, Huai Chen, Lisheng Wang, Alex Golts, Daniel Khapun, and Daniel Shats. The kits21 challenge: Automatic segmentation of kidneys, renal tumors, and renal cysts in corticomedullary-phase ct. *arXiv preprint arXiv:2307.01984*, 2023.

[Heskes *et al.*, 2020] Tom Heskes, Evi Sijben, Ioan Gabriel Bucur, and Tom Claassen. Causal shapley values: Exploiting causal knowledge to explain individual predictions of complex models. *Advances in neural information processing systems*, 33:4778–4789, 2020.

[Jeyakumar *et al.*, 2020] Jeya Vikranth Jeyakumar, Joseph Noor, Yu-Hsi Cheng, Luis Garcia, and Mani Srivastava. How can i explain this to you? an empirical study of deep neural network explanation methods. *Advances in Neural Information Processing Systems*, 2020.

[Koh and Liang, 2017] Pang Wei Koh and Percy Liang. Understanding black-box predictions via influence functions. In *International Conference on Machine Learning*. PMLR, 2017.

[Koh *et al.*, 2020] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020.

[Li *et al.*, 2018] Oscar Li, Hao Liu, Chaofan Chen, and Cynthia Rudin. Deep learning for case-based reasoning through prototypes: A neural network that explains its predictions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

[Li *et al.*, 2022] Xuhong Li, Haoyi Xiong, Xingjian Li, Xuanyu Wu, Xiao Zhang, Ji Liu, Jiang Bian, and Dejing Dou. Interpretable deep learning: Interpretation, interpretability, trustworthiness, and beyond. *Knowledge and Information Systems*, 64(12):3197–3234, 2022.

[Lin *et al.*, 2017] Tsung-Yi Lin, Piotr Dollár, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2117–2125, 2017.

[Lin *et al.*, 2023] Weifeng Lin, Ziheng Wu, Jiayu Chen, Jun Huang, and Lianwen Jin. Scale-aware modulation meet transformer. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6015–6026, 2023.

[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.

[Pedapati *et al.*, 2020] Tejaswini Pedapati, Avinash Balakrishnan, Karthikeyan Shanmugam, and Amit Dhurandhar. Learning global transparent models consistent with local contrastive explanations. *Advances in Neural Information Processing Systems*, 33, 2020.

[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.

[Ribeiro *et al.*, 2016] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. Why should i trust you?: Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, 2016.

[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.

[Shorten *et al.*, 2021] Connor Shorten, Taghi M Khoshgoftaar, and Borko Furht. Deep learning applications for covid-19. *Journal of big Data*, 8(1):1–54, 2021.

[Sinha *et al.*, 2023] Sanchit Sinha, Mengdi Huai, Jianhui Sun, and Aidong Zhang. Understanding and enhancing robustness of concept-based models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pages 15127–15135, 2023.

[Sinha *et al.*, 2024a] Sanchit Sinha, Guangzhi Xiong, and Aidong Zhang. Colidr: Co ncept l earn i ng using aggregated d isentangled r epresentations. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 2699–2710, 2024.

[Sinha *et al.*, 2024b] Sanchit Sinha, Guangzhi Xiong, and Aidong Zhang. A self-explaining neural architecture for generalizable concept learning. *arXiv preprint arXiv:2405.00349*, 2024.

[Sundararajan *et al.*, 2017] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International conference on machine learning*, pages 3319–3328. PMLR, 2017.

[Touvron *et al.*, 2021] 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.

[Tuli *et al.*, 2021] Shikhar Tuli, Ishita Dasgupta, Erin Grant, and Thomas L Griffiths. Are convolutional neural networks or transformers more like human vision? *arXiv preprint arXiv:2105.07197*, 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.

[Wah *et al.*, 2011] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The caltech-ucsd birds-200-2011 dataset. 2011.

[Wang *et al.*, 2023] Bowen Wang, Liangzhi Li, Yuta Nakashima, and Hajime Nagahara. Learning bottleneck concepts in image classification. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition*, pages 10962–10971, 2023.

[Xian *et al.*, 2018] Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning—a comprehensive evaluation of the good, the bad and the ugly. *IEEE transactions on pattern analysis and machine intelligence*, 41(9):2251–2265, 2018.

[Xu *et al.*, 2021] Yufei Xu, Qiming Zhang, Jing Zhang, and Dacheng Tao. Vitae: Vision transformer advanced by exploring intrinsic inductive bias. *Advances in neural information processing systems*, 34:28522–28535, 2021.

[Xu *et al.*, 2023] Li Xu, Xin Yan, Weiyue Ding, and Zechao Liu. Attribution rollout: a new way to interpret visual transformer. *Journal of Ambient Intelligence and Humanized Computing*, 14(1):163–173, 2023.

[Zhu *et al.*, 2020] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable detr: Deformable transformers for end-to-end object detection. *arXiv preprint arXiv:2010.04159*, 2020.