Attributable Visual Similarity Learning

Borui Zhang, Wenzhao Zheng, Jie Zhou, Jiwen Lu*
Department of Automation, Tsinghua University, China
Beijing National Research Center for Information Science and Technology, China
{zhang-br21, zhengwz18}@mails.tsinghua.edu.cn; {jzhou, lujiwen}@tsinghua.edu.cn

Abstract

This paper proposes an attributable visual similarity learning (AVSL) framework for a more accurate and explainable similarity measure between images. Most existing similarity learning methods exacerbate the unexplainability by mapping each sample to a single point in the embedding space with a distance metric (e.g., Mahalanobis distance, Euclidean distance). Motivated by the human semantic similarity cognition, we propose a generalized similarity learning paradigm to represent the similarity between two images with a graph and then infer the overall similarity accordingly. Furthermore, we establish a bottom-up similarity construction and top-down similarity inference framework to infer the similarity based on semantic hierarchy consistency. We first identify unreliable higher-level similarity nodes and then correct them using the most coherent adjacent lower-level similarity nodes, which simultaneously preserve traces for similarity attribution. Extensive experiments on the CUB-200-2011, Cars196, and Stanford Online Products datasets demonstrate significant improvements over existing deep similarity learning methods and verify the interpretability of our framework. [1]

1. Introduction

Similarity learning is a fundamental task in the field of computer vision, where most prevalent works (i.e., metric learning methods) employ a distance metric to measure the similarities between samples. They transform features into an embedding space and define the dissimilarity as the Euclidean distance in this space, where the objective is to cluster similar samples together and separate dissimilar ones apart from each other. While conventional methods use hand-crafted features like SIFT [21] and LBP [1], deep metric learning methods employ convolutional neural networks (CNNs) [16] to extract more representative features and demonstrate superior performance. In recent years, similarity learning has been widely applied to various vision tasks such as face recognition [11, 30, 35], person re-identification [4, 10, 18, 47], and image classification [2, 22].

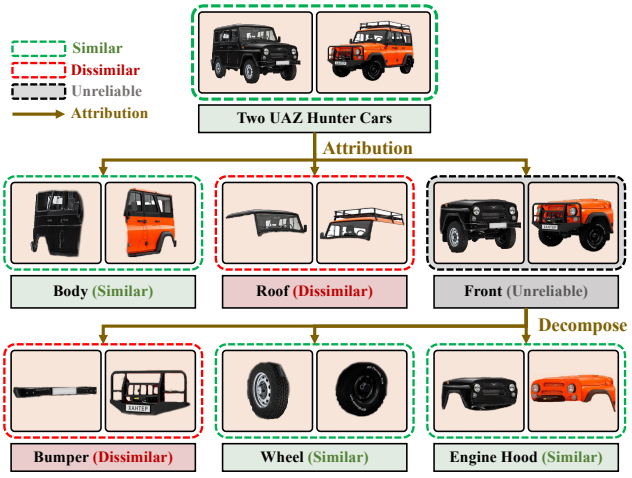


Figure 1. The motivation of the proposed AVSL framework. Humans recognize each image as a complex set of concepts and compare two images hierarchically [17]. For example, when inferring the similarity between two cars, humans usually first compare high-level features such as shapes or colors and then turn to finer features such as wheel structures when a coarse observation does not distinctly distinguish them. Motivated by this, we propose to employ a graph structure to decompose sample pairs into discriminative concept nodes, which is more consistent with how humans perceive the cognitive distance and is beneficial to the attribution of the similarity measurement.

The essential goal of visual similarity learning is to obtain a similarity measure that generalizes well to unseen data. It has been shown that the good generalization of the human visual system comes from the ability to parse objects into parts and relations and learn the underlying concepts [17]. Humans also infer the similarity between two images hierarchically by first comparing high-level features and then delving into lower-level features, as illustrated in Figure 1. However, most existing similarity learning methods simply project each sample to one single vector and employ the Mahalanobis distance or Euclidean distance as the similarity function. They only use the top-level feature to

*Corresponding author.
[1] Code: https://github.com/zbr17/AVSL.

represent an image and directly compute the similarity without inference. Also, using a single vector for similarity measure exacerbates the unexplainability caused by the black-box CNNs and leads to untraceable similarity measurement, i.e., we can hardly attribute the overall similarity to specific features. To alleviate this issue, some methods [3,33,59] attempt to extend neural network visualization techniques to deep metric learning and generate a saliency map for each image. Still, they treat the similarity computing model as a black box and can only explain it subjectively in a post hoc way, where the similarity computing process remains untraceable and unexplainable.

In this paper, we propose an attributable visual similarity learning (AVSL) framework to actively explain the learned similarity measurement. We generalize the prevalent metric learning paradigm to represent the similarity between images by a graph and then analyze it to infer the overall similarity. We use CNNs to extract hierarchical visual features in a bottom-up manner, where higher-level features encode more abstract concepts [45,50] and can be regarded as a combination of low-level features [51,52]. We further construct an undirected graph to represent the similarity between images. We then propose a top-down similarity inference method based on hierarchy consistency. We start from high-level similarity nodes and rectify identified unreliable nodes using adjacent low-level similarity nodes until reaching the lowest level, which is similar to how humans compare two objects from coarse to fine. The overall similarity can be easily attributed to the effect of each similarity node corresponding to certain visual concepts. Our framework can be readily applied to existing deep metric learning methods with various loss functions and sampling strategies. Extensive experiments on the widely used CUB-200-2011 [37], Cars196 [15], and Stanford Online Products [24] datasets demonstrate that our AVSL framework can significantly boost the performance of various deep metric learning methods and achieve state-of-the-art results. We also conduct visualization experiments to demonstrate the attributability and interpretability of our method.

2. Related Work

Similarity learning: Similarity learning aims to learn a similarity function to accurately measure the semantic similarities between images. Conventional methods adopt the Mahalanobis distance to learn linear metric functions and further use kernel tricks to model nonlinear relations. Recent deep metric learning methods employ convolutional neural networks to learn an embedding space and use the Euclidean distance for similarity measurement, where the majority of works focus on designing different loss functions [7, 13, 23, 27, 30, 32, 34, 40] and sampling strategies [5, 6, 8, 25, 30, 41, 43, 49, 55] for more effective training of the metric. For example, the contrastive loss [7] pulls positive pairs together while pushing negative ones farther than a fixed margin. Song et al. [24] further proposed a lifted structured loss considering the global connections among a mini-batch. Movshovitz et al. [23] simplified the pair sampling to linear complexity by including proxies in the loss formulations. Still, an appropriate sampling strategy has been proven to be effective to boost performance. For example, Schroff et al. [30] presented a semi-hard sampling strategy to select informative samples while discarding outliers. Harwood et al. [8] proposed a smart sampling strategy adaptive to different training stages.

Other works explore different designs of the similarity function to improve the performance. For example, Yuan et al. [48] and Huang et al. [12] proposed an SNR distance and a PDDM module, respectively, to better guide the training process but still uses the conventional Euclidean distance during testing. Verma et al. [36] learned hierarchical distance metrics based on the class taxonomy. Ye et al. [44] employed a set of metrics to describe similarities from different perspectives. However, all the aforementioned methods represent dissimilarity by projecting samples into single points in the Euclidean distance which implies the triangle inequation, while the proposed AVSL framework represents samples in a graph manner to model relations between concepts. Zheng et al. [57] also exploited relations by projecting samples with multiply embedders to learn a sub-space structure. Differently, we propose to decompose the overall similarity hierarchically with hierarchy consistency as the inductive bias and employ a top-down similarity structure compatible with bottom-up similarity construction.

Explainable artificial intelligence: Explainable artificial intelligence (XAI) has attracted considerable attention in recent years, resulting from the demand for stabler and safer AI applications. One category of works aims to interpret the outputs of black-box models by visualization or imitation [3, 28, 31, 33, 46, 50, 52, 58, 59] (i.e., passive methods). For example, Zeiler et al. [50] and Selvaraju et al. [31] projected hidden feature maps into the input space using deconvolution and gradients, respectively, which can assist humans to understand the semantics of the hidden layers. Ribeiro et al. [28] and Zhang et al. [52] employed linear regressions and graph models to imitate complex rules in the black-box inference process. Another category of works attempts to modify the model architecture to improve its explainability [38, 42, 53] (i.e., active methods). For instance, Zhang et al. [53] restricted each kernel of hidden layers to encoding a single concept. Wu et al. [42] proposed a tree regularization loss to favor models that can be more easily approximated by a simple decision tree.

A few works [3,33,54,59] seek to extend neural network visualization techniques for deep metric learning. Nevertheless, they can only obtain global saliency maps and can hardly conduct quantitive attribution analysis of overall

similarities, which cannot provide detailed interpretations of similarity models. To the best of our knowledge, we are the first to explore an attributable and explainable similarity learning framework. Imitating humans to compare objects from coarse to fine, the proposed AVSL can attribute overall similarities to hierarchical hidden concepts.

3. Proposed Approach

In this section, we first present a generalized similarity learning paradigm and then elaborate on the proposed bottom-up similarity construction and top-down similarity inference. Finally, we present the AVSL framework and demonstrate how to quantitatively attribute the similarity to different levels of features under our framework.

3.1. Generalized Similarity Learning Paradigm

Let $\mathbf{X} = \{\mathbf{x}^{(1)}, \mathbf{x}^{(2)}, \cdots, \mathbf{x}^{(\mathbf{N})}\}$ denotes the image set, where sample $\mathbf{x}^{(n)} \in \mathbf{X}$ has a label $l^{(n)} \in \{l_1, l_2, \cdots, l_C\}$ with C being the number of classes. Given an L-layer CNN f and a sample $\mathbf{x}$, we call the l-layer outputs as feature maps, denoted as $\mathbf{z}^l = f^l(\mathbf{x}) \in \mathbb{R}^{c_l \times h_l \times w_l}$, where c_l, h_l, and w_l denote channel, height, and weight respectively. Then a pooling operation $g^l(\cdot)$ reduces feature maps to vectors $\mathbf{v}^l = g^l(\mathbf{z}^l) \in \mathbb{R}^{c_l}$. Existing deep metric learning methods usually add a linear projector $h^l(\cdot)$ to map $\mathbf{v}^l$ into an r-dimension embedding space: $\mathbf{e}^l = h^l(\mathbf{v}^l) \in \mathbb{R}^r$, where the dissimilarity between two images $\mathbf{x}, \mathbf{x}'$ is $\hat{d}(\mathbf{x}, \mathbf{x}') = d(\mathbf{e}^l, \mathbf{e}'^l) = \|\mathbf{e}^l - \mathbf{e}'^l\|_2$. For simplicity, We use similarity and dissimilarity interchangeably unless stated otherwise.

However, existing deep similarity learning methods only utilize the features from the top layer while discarding those from the hidden layers, which might contain complementary information. To address this, we propose a generalized similarity learning (GSL) paradigm, which constructs an undirected graph $\mathcal{H}$ involving embeddings from each layer to compute overall similarities. We denote each element of the embedding $\mathbf{e}^l$ as e_i^l, and define the **similarity node** of $\mathcal{H}$ as $\delta_i^l = |e_i^l - e_i'^l|$. In addition, the edge ω_{ij} of $\mathcal{H}$ will be elaborated in Section 3.2. To sum up, the GSL paradigm is composed of two modules:

- **Similarity construction**: compute similarity nodes δ_i^l and edges ω_{ij} to construct an undirected graph $\mathcal{H}$.
- **Similarity inference**: infer the overall similarity d according to the graph $\mathcal{H}$.

The conventional metric learning methods can be regarded as a special case of GSL paradigm as shown in Figure 2 when only constructing $\mathcal{H}$ with the top layer similarity nodes δ_i^L and defining the overall similarity as $d = \sum_{i=1}^{r} (\delta_i^L)^2$.

Taking full advantage of the hierarchy consistency in deep CNNs, we further propose an attributable visual similarity learning (AVSL) framework composed of bottom-up similarity construction and top-down similarity inference.

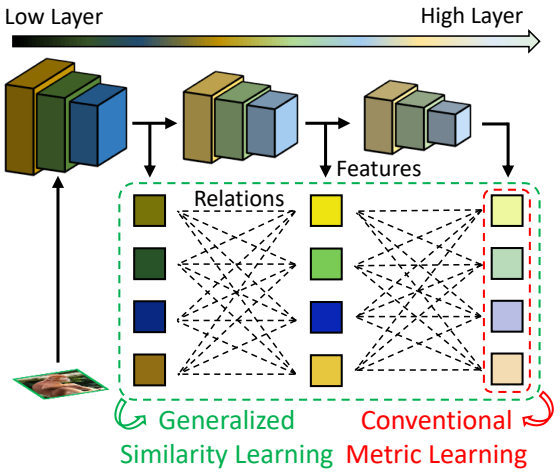


Figure 2. An illustration to show the difference between the proposed GSL and conventional DML. After extracting hierarchical features by applying CNNs, the conventional metric learning methods mainly focus on top-level features while the proposed GSL takes full advantage of features from all layers and the interactions between them to construct the similarity graph.

3.2. Bottom-Up Similarity Construction

Conventional similarity learning methods only use embeddings from the top layer to compute the overall similarity, making it difficult to trace back to different concepts, which are encoded by embeddings from all layers as evidenced by Zhang et al. [52]. Different levels of features encode different levels of concepts containing complementary information, where the large receptive fields of high-level features enable them to represent high-level semantic information and omit some high-frequency details, while low-level features can capture detailed information such as textures but fail to perceive global semantics due to the restricted receptive fields. Still, high-level features can be regarded as the combination of low-level features [51, 52], and their connections can be exploited for the subsequent similarity inference. Therefore, we propose a bottom-up similarity construction method to compute different levels of similarity nodes and the connections between them.

The first step is to compute similarity nodes δ_i^l. We extract the feature map $\mathbf{z}^l$ of the l-th layer by convolutional blocks from bottom to top, and then obtain the feature vector $\mathbf{v}^l$ using global pooling. Subsequently, we employ a fully connected layer to map the feature vector to the corresponding embedding $\mathbf{e}^l$. Finally we obtain the similarity nodes by computing the square of the difference between normalized embeddings $\tilde{\mathbf{e}}^l, \tilde{\mathbf{e}}'^l$:

$$\delta_i^l = |\tilde{e}_i^l - \tilde{e}_i'^l|^2. \tag{1}$$

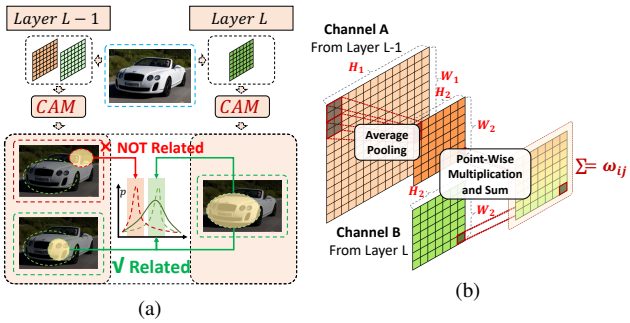


Figure 3. An illustration to show how to construct edges. (a) The basic idea of how to compute correlations. We propose to recover nodes to spatial distributions employing CAMs and regard the overlap degree of corresponding distributions as the correlation between nodes. (b) The detailed operations. We first rescale each CAMs into the same size and then compute the convolution of two normalized CAMs as the correlation value of the edge.

The second step is to compute edge w_{ij}^l between nodes δ_i^l and δ_j^{l-1}. Since pooling operation erases the spatial information, which encodes relations between different nodes, we propose to utilize CAMs [58] of each node to recover relations as illustrated in Figure 3. We first compute CAMs of nodes as follows:

$$\mathbf{u}_i^l = \sum_{j=1}^{c^l} a_{ij}\mathbf{z}_j^l \in \mathbb{R}^{h_l \times w_l}, \tag{2}$$

where $\mathbf{z}_j^l$ denotes the j-th slice of the feature map $\mathbf{z}^l$, and a_{ij} indicates weights of the linear layer $h^l(\cdot)$. We consider two nodes correlated if the two distributions of the corresponding CAMs are statistically similar. After rescaling and vectorizing CAMs to the same scale vectors $\hat{\mathbf{u}}_i^l, \hat{\mathbf{u}}_j^{l-1} \in \mathbb{R}^p$, where $p = \min\{h_l, h_{l-1}\} \times \min\{w_l, w_{l-1}\}$, we establish the correlation $\hat{\omega}_{ij}^l$ by computing the inner product of $\hat{\mathbf{u}}_i^l$ and $\hat{\mathbf{u}}_j^{l-1}$ as follows:

$$\hat{\omega}_{ij}^l = \langle \hat{\mathbf{u}}_i^l, \hat{\mathbf{u}}_j^{l-1} \rangle. \tag{3}$$

To obtain the final edges ω_{ij}^l, we adopt the momentum updating strategy to gradually incorporate all training samples:

$$\omega_{ij}^l \leftarrow \gamma\omega_{ij}^l + (1-\gamma)\hat{\omega}_{ij}^l, \tag{4}$$

where γ is a momentum factor.

3.3. Top-Down Similarity Inference

Having constructed the graph $\mathcal{H}$ composed of similarity nodes δ_i^l and correlation edges ω_{ij}^l, we want to incorporate them to compute an overall similarity. Taking full advantage of the hierarchy consistency of CNNs, we propose a top-down similarity inference method based on the graph $\mathcal{H}$. On the one hand, we argue that different levels of features encode complementary information, enabling the corresponding similarity nodes to produce relatively independent similarity judgment. On the other hand, the most correlated similarity nodes of adjacent levels should be consistent, which can be used as a non-trivial constraint to restrict the overall similarity. Motivated by this, we propose to identify unreliable higher-level similarity nodes and rectify them using adjacent lower-level similarity nodes with the largest correlations.

Analogous to the process of humans comparing images from coarse to fine, we infer the overall similarity from top to bottom. We first estimate the reliability of the similarity nodes at the l-th layer to identify the unreliable ones. Intuitively, we deem a similarity node unreliable if its corresponding CAM is unable to focus clearly on specific regions. Formally, we employ the standard deviation of normalized CAMs to compute the reliability as:

$$\eta_i^l = std(\hat{\mathbf{u}}_i^l) \cdot std(\hat{\mathbf{u}}_i'^l), \tag{5}$$

where $std(\cdot)$ denotes the standard deviation and $\hat{\mathbf{u}}_i^l$, $\hat{\mathbf{u}}_i'^l$ indicate normalized CAMs of samples x, x'. Then we apply a sigmoid function to map η_i^l to the range of $(0,1)$ as:

$$p_i^l = \sigma(\alpha_i^l\eta_i^l + \beta_i^l) = \frac{e^{\alpha_i^l\eta_i^l+\beta_i^l}}{e^{\alpha_i^l\eta_i^l+\beta_i^l}+1} \in (0,1), \tag{6}$$

where α_i^l, β_i^l are node-wise learnable parameters.

We then rectify unreliable nodes at the higher-level layer with the correlated ones at the adjacent lower-level layer. For an unreliable δ_i^l at the l-th layer, we denote the index set of k most correlated nodes at the $(l-1)$-th layer as:

$$\mathbf{I}(\delta_i^l) = \{j | \omega_{ij}^l \in \max_k\{\omega_{im:m=1,2,\cdots,r}^l\}\}, \tag{7}$$

where $\max_k(\cdot)$ denotes the set of k largest values. Subsequently, we compute the rectified similarity node $\hat{\delta}_i^l$ by the sum of the original one δ_i^l and adjacent related low-level ones δ_j^{l-1} weighted by the unreliability p_i^l as follows:

$$\hat{\delta}_i^l = \begin{cases} p_i^l\delta_i^l + (1-p_i^l)\sum_{j=1}^{r}\tilde{\omega}_{ij}^l\hat{\delta}_j^{l-1}, & 2 \le l \le L \\ \delta_i^l, & l = 1 \end{cases} \tag{8}$$

where $\tilde{\omega}_{ij}^l = \frac{\mathbb{I}_{j\in I(\delta_i^l)}\omega_{ij}^l}{\sum_{k=1}^{r}\mathbb{I}_{k\in I(\delta_i^l)}\omega_{ik}^l} \in [0,1]$ denotes the normalized edges, $\mathbb{I}_{(\cdot)}$ is the indicative function, and $\delta_i^l, \omega_{ij}^l$ are nodes and edges of the graph $\mathcal{H}$. Since $\delta_i^l \ge 0, p_i^l \in (0,1), \tilde{\omega}_{ij}^l \ge 0$, we know that $\hat{\delta}_i^l \ge 0$. For convenience, we reorganize (8) in a matrix format as follows:

$$\hat{\boldsymbol{\delta}}^l = \mathbf{P}^l\boldsymbol{\delta}^l + (\mathbf{I}-\mathbf{P}^l)\tilde{\mathbf{W}}^l\hat{\boldsymbol{\delta}}^{l-1} \tag{9}$$

$$\text{where } \hat{\boldsymbol{\delta}}^l = [\hat{\delta}_1^l\ \hat{\delta}_1^l\ \cdots\ \hat{\delta}_r^l] \in \mathbb{R}^r$$

$$\boldsymbol{\delta}^l = [\delta_1^l\ \delta_2^l\ \cdots\ \delta_r^l] \in \mathbb{R}^r$$

$$\mathbf{P}^l = \begin{cases} \text{diag}(p_1^l, \cdots, p_r^l), & l \ge 2 \cdot \\ \mathbf{I}, & l = 1 \end{cases}$$

$$\tilde{\mathbf{W}}^l = (\tilde{\omega}_{ij}^l) \in \mathbb{R}^{r\times r}$$

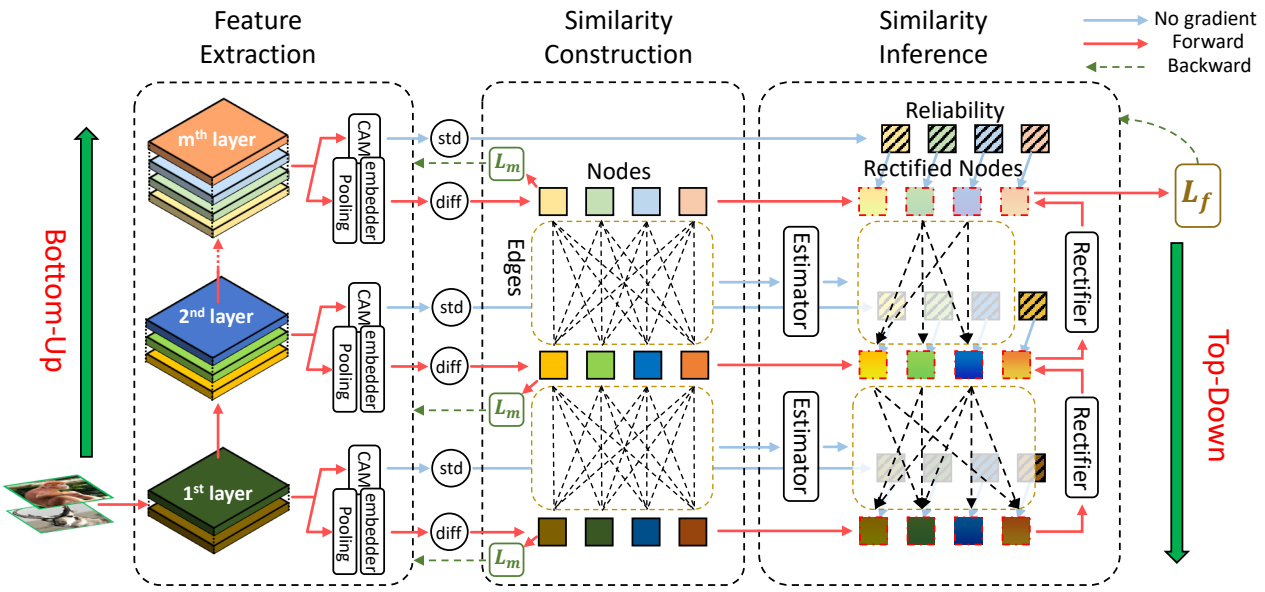


Figure 4. An illustration of the architecture of the proposed AVSL framework. We first extract a set of feature maps from multiple layers of a CNN network and perform global pooling followed by linear projection to obtain the set of embedding. We then compute the square of the absolute differences between the corresponding embeddings as similarity nodes. For similarity inference, we first estimate the reliability of the similarity nodes and rectify them using the most correlated ones in the adjacent lower level. We compute the overall similarity as the sum of the rectified similarity nodes of the top layer which can be conveniently attributed to specific similarity nodes in different levels.

Finally, we define the overall similarity between two images as the sum of rectified top-level similarity nodes as $\hat{d} = \sum_{i=1}^{r} \hat{\delta}_i^L$, while the smaller value indicates more similarity. Following (9), we can infer it recursively in a top-down manner efficiently.

3.4. Attributable Visual Similarity Learning

The proposed AVSL framework employs a bottom-up similarity construction and top-down similarity inference method based on hierarchy consistency to extend the conventional similarity learning, as shown in Figure 4. We divide our framework into three phases: **training**, **attribution**, and **evaluation**.

Training: In order to learn network parameters, We define l-th level similarity as $d^l = \sum_{i=1}^{r} \delta_i^l$. The proposed AVSL is compatible with existing deep metric learning methods with various loss functions and sampling strategies to further improve their performance. For a particular loss function $L(\cdot)$, the overall objective of the proposed AVSL framework is formulated as follows:

$$\min_{\theta_1,\theta_2} J = \min_{\theta_1} \sum_{l=1}^{L} L^m(d^l) + \min_{\theta_2} L^f(\hat{d}) \tag{10}$$

where θ_1 corresponds to the CNN network parameters and θ_2 represents the parameters of the similarity inference module including α_i^l, β_i^l of the reliability estimatation modules in (6). We only use the loss on overall similarities to train the similarity inference module and the loss on level similarities to train the similarity construction module. L^m targets at learning discriminative embeddings for each layer, while L^f only aims at learning the similarity inference process to obtain an accurate and robust overall similarity.

Attribution: It is essential to analyze how the model infers the similarity. We reorganize the overall similarity $\hat{d}$ in a linear combination format following (9) as:

$$\begin{aligned} \hat{d} &= \sum_{i=1}^{r} \hat{\delta}_i^L = \mathbf{1}\hat{\boldsymbol{\delta}}^L \\ &= \mathbf{1}\mathbf{P}^L\boldsymbol{\delta}^L + \mathbf{1}(\mathbf{I}-\mathbf{P}^L)\tilde{\mathbf{W}}^L\hat{\boldsymbol{\delta}}^{L-1} \\ &= \sum_{l=1}^{L} \mathbf{1}\boldsymbol{\Lambda}^l\boldsymbol{\delta}^l = \sum_{l=1}^{L}\sum_{i=1}^{r} \lambda_i^l \delta_i^l, \end{aligned} \tag{11}$$

where $\boldsymbol{\Lambda}^l = (\mathbf{I}-\mathbf{P}^L)\tilde{W}^L \cdots (\mathbf{I}-\mathbf{P}^{l+1})\tilde{W}^{l+1}\mathbf{P}^l$, and $\boldsymbol{\lambda}^l = [\lambda_1^l\ \lambda_1^l\ \cdots \lambda_r^l] = \mathbf{1}^T\boldsymbol{\Lambda}^l$. The weight λ_i^l represents the sensitivity of the overall similarity to each similarity node δ_i^l, which means the change of the similarity node δ_i^l will contribute more to the change of the overall similarity $\hat{d}$ if its corresponding weight λ_i^l is larger. Finally, saliency maps are generated to demonstrate the attribution process. The proposed AVSL framework is convenient for visualization since we compute similarity nodes and corresponding CAMs simultaneously in a single forward propagation.

Evaluation: During the evaluation, we freeze all parameters and only compute the overall similarities $\hat{d}$ using (11) to represent the similarities between the given query samples and gallery samples.

4. Experiment

In this section, we conducted experiments on three widely used datasets including CUB-200-2011 [37], Cars196 [15], and Stanford Online Products [24] to evaluate the accuracy and interpretability of our AVSL framework. We used the Recall@Ks as the performance metrics, which compute the percentage of well-separated samples acknowledged if we can find at least one corrected retrieved sample in the K nearest neighbors.

4.1. Dataset

For quantitative evaluation, we conducted experiments under a zero-shot setting following the non-intersecting dataset partition protocol [24]. The split scheme of datasets are as follows:

- **CUB-200-2011** [37] consists of 200 bird species and 11,788 images. We split the first 100 species (5,864 images) for training and the rest 100 species (5,924 images) for testing.
- **Cars196** [15] contains 196 car types and 16,185 images. The first 98 types (8,054 images) were used for training while the other 98 types (8,131 images) were kept for testing.
- **Stanford Online Products** [24] includes 22,634 classes of online products totaling 120,053 images. We divide the first 11,318 classes (59,551 images) into training set and the rest 11,316 classes (60,502 images) into testing set.

For qualitative demonstration, we further visualized the similarity attribution results of some randomly selected samples in CUB-200-2011 and Cars196. All datasets are publicly available for non-commercial research and educational purposes.

4.2. Implementation Details

We conducted all the experiments using the PyTorch package [26] on an NVIDIA RTX 3090 GPU and employed the ResNet50 [9] as the CNN feature extractor (i.e., f^m) for fair comparisons. Limited by the GPU device memory, we only selected feature maps for every three layers for similarity construction (i.e., layers 3, 4, and 5). We employed a global pooling operation (i.e., g^l) and a linear layer (i.e., h^l) after each selected layer. We fixed the embedding size to 512 for all selected layers. For data argumentation, we first resized images to 256 by 256 to apply random reshaping and horizontal flip and then randomly cropped them to 224 by 224. Before training, we initialized the CNN with weights pre-trained on ImageNet ILSVRC dataset [29]. We adopted AdamW [20] to train our model with an initial learning rate 1×10^{-4} and a weight decay of 0.0001. We fixed the batch size to 180 and set the momentum factor γ to 0.5. For the margin loss [41], we set the margin factors α and β to 1.2 and 0.2, respectively. For the ProxyAnchor loss [13], we set the temperature $\alpha = 16$, positive margin $\gamma_{pos} = 1.8$, and negative margin $\gamma_{neg} = 2.2$. We tuned all hyperparameters by grid search on a reserved validation set.

4.3. Quantitative Results and Analysis

Comparisons with existing methods: We applied the proposed AVSL framework to the margin loss [41] and the ProxyAnchor loss [13] for demonstration and compared our framework with several baseline methods. Table 1 shows the image retrieval performance on the CUB-200-2011 [37], Cars196 [15], and Stanford Online Products [24] respectively. We mark the best results with bold red and highlight our superior results over the associated methods without AVSL in bold black.

We observe that our AVSL framework can greatly improve the original deep metric learning methods by a large margin and achieve state-of-the-art performance on three datasets. We ascribe the improvement to exploiting the graph structure by employing the hierarchy consistency between different similarity nodes as the inductive bias which is consistent with how humans perceive the semantic visual similarity. By rectifying unreliable higher-level similarity nodes with the most correlated ones in the lower-level layer, we achieve a more accurate and robust similarity measure with the proposed top-down similarity inference.

Ablation study: We first performed an ablation study to evaluate the contribution of each component of the proposed AVSL framework. We report the experimental results on the Cars196 [15] dataset with the ProxyAnchor loss [13], as shown in Table 2, but we observe similar outcomes with the other loss functions. We highlight the best results using bold numbers.

ProxyAnchor denotes the baseline method of using the ProxyAnchor loss. **+ M** is short for 'multi-layer'. In this trial, we exerted extra loss constraints on embeddings of all layers (i.e., layers 3, 4, and 5). **+ R** stands for 'reliability', which means that we utilize the reliabilities defined in (6) and only keep reliable nodes to compute the similarities. Based on **+ M & R** setting, if we further consider edges between similarity nodes, we can get complete components of the proposed AVSL framework. In the **+ M (concat)** setting, we concatenated embeddings of all three layers (i.e., the final dimension equals to $3 \times 512 = 1536$) to compute similarities, which is a strong baseline to further demonstrate the effectiveness of the proposed AVSL framework.

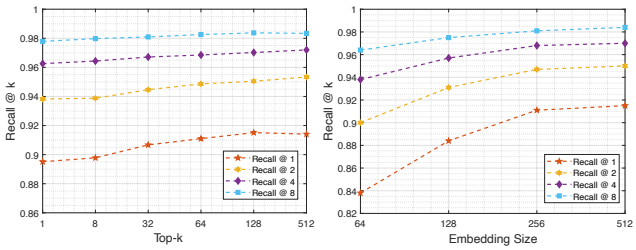
We observe that the proposed AVSL framework achieves

Table 1. Recall@K(%) on the test sets of CUB-200-2011, Cars196, and Stanford Online Products.

Datasets		CUB-200-2011				Cars196				Stanford Online Products		
Methods	Setting	R@1	R@2	R@4	R@8	R@1	R@2	R@4	R@8	R@1	R@10	R@100
HDC [49]	384G	53.6	65.7	77.0	85.6	73.7	83.2	89.5	93.8	70.1	84.9	93.2
DAML [6]	512BN	52.7	65.4	75.5	84.3	75.1	83.8	89.7	93.5	68.4	83.5	92.3
DVML [19]	512BN	52.7	65.1	75.5	84.3	82.0	88.4	93.3	96.3	70.2	85.2	93.8
Angular [39]	512G	53.6	65.0	75.3	83.7	71.3	80.7	87.0	91.8	67.9	83.2	92.2
DAMLRRM [43]	512G	55.1	66.5	76.8	85.3	73.5	82.6	89.1	93.5	69.7	85.2	93.2
DE-DSP [5]	512G	53.6	65.5	76.9	-	72.9	81.6	88.8	-	68.9	84.0	92.6
HDML [55]	512BN	53.7	65.7	76.7	85.7	79.1	87.1	92.1	95.5	68.7	83.2	92.4
A-BIER [25]	512G	57.5	68.7	78.3	86.2	82.0	89.0	93.2	96.1	74.2	86.9	94.0
ABE [14]	512G	60.6	71.5	79.8	87.4	85.2	90.5	94.0	96.1	76.3	88.4	94.8
MS [40]	512BN	65.7	77.0	86.3	91.2	84.1	90.4	94.0	96.5	78.2	90.5	96.0
SoftTriple [27]	512BN	65.4	76.4	84.5	91.6	86.1	91.7	95.0	97.3	78.3	90.3	95.9
Circle [34]	512BN	66.7	77.4	86.2	91.2	83.4	89.8	94.1	96.5	78.3	90.5	96.1
DCML [56]	512R	68.4	77.9	86.1	91.7	85.2	91.8	96.0	98.0	79.8	90.8	95.8
DIML [54]	512R	68.2	-	-	-	87.0	-	-	-	79.3	-	-
DRML [57]	512R	68.7	78.6	86.3	91.6	86.9	92.1	95.2	97.4	79.9	90.7	96.1
Margin [41]	512R	65.6	75.9	84.3	90.8	78.2	86.7	92.3	95.3	72.4	85.3	92.8
Margin-AVSL	512R	**68.8**	**79.2**	**87.3**	**92.7**	**81.1**	**88.8**	**93.4**	**96.4**	**76.8**	**89.2**	**95.4**
ProxyAnchor [13]	512R	69.7	80.0	87.0	92.4	87.7	92.9	95.8	97.9	78.4	90.5	96.2
ProxyAnchor-AVSL	512R	**71.9**	**81.7**	**88.1**	**93.2**	**91.5**	**95.0**	**97.0**	**98.4**	**79.6**	**91.4**	**96.4**

Table 2. Ablation study with different model settings.

Method	R@1	R@2	R@4	R@8
ProxyAnchor	87.7	92.9	95.8	97.9
ProxyAnchor + M	89.7	93.9	96.3	97.9
ProxyAnchor + M & R	89.9	94.0	96.4	98.1
ProxyAnchor + M (concat)	90.6	94.6	96.8	98.2
ProxyAnchor + AVSL	**91.5**	**94.8**	**96.9**	**98.4**



(a) Different top-k values. (b) Different embedding sizes.

Figure 5. Influence of hyperparameters.

better performance than all the compared counterparts and all modules contribute to the overall improvement. In particular, imposing loss constraints on the embeddings of hidden layers can boost the performance of the original method by 2.0%. It is also beneficial to employ reliabilities defined in (6) to guide the selection of informative nodes. Subsequently, exploiting the relations among similarity nodes and employing the hierarchy consistency for similarity inference can further improve the performance by 1.6%. We also see that our method suppresses **+ M (concat)**, which demonstrates that learning informative relations and reliabilities is crucial for effective inference.

Influence of hyperparameters: The k value defined in (7) controls how many adjacent related nodes participate in rectification. Figure 5a reveals the continuous improvement when increasing the k value. And we can further discover that the influence of k shows diminishing marginal effects, thus we fixed k to 128 to complete all other experiments. In addition, the dimension of embeddings significantly impacts the performance as shown in Figure 5b, and larger embedding size leads to higher performance. In particular, when only fixing the dimension to 128, our proposed AVSL could surpass all other methods with a recall@1 score of 88.4% on the Cars196 dataset, which further demonstrates the effectiveness of our framework.

4.4. Visualization

To verify the interpretability of the proposed AVSL framework, we randomly selected a triplet from CUB-200-2011 to show the attribution results, as shown in Figure 6. For the pairs in the triplet, we first selected the 128 most reliable similarity nodes, and then ranked those nodes according to their similarities. We observe that most of the CAMs of nodes focus on specific parts of images while some saliency maps are unrecognizable. We think that this phenomenon is due to the singularity of the relationships between spatial coordinates and concepts. Also, we discover that the dissimilarity distribution of the negative pair

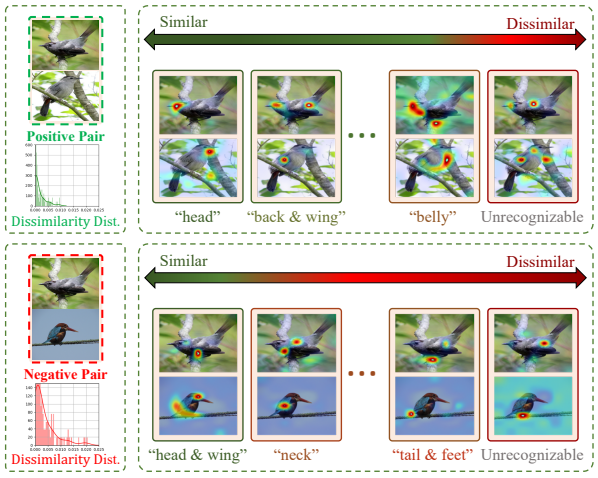


Figure 6. Visualization of the attribution result. We randomly select a triplet from CUB-200-2011 and rank the results according to the similarity among the 128 most reliable nodes for each sample pair. We use green and red boxes to denote positive and negative pairs, respectively. Best viewed in color.

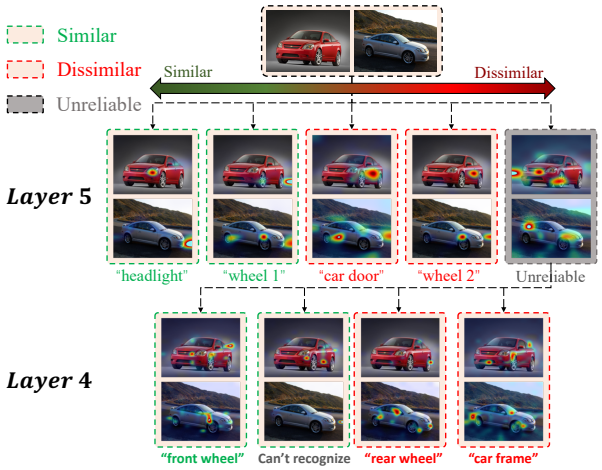


Figure 7. Visualization of the attribution process. We randomly select 2 samples from Cars196 and attribute the overall similarity to the specific similarity nodes in an undirected graph manner. Best viewed in color.

is more dispersed than the positive one as shown on the left of Figure 6. This means that the nodes of the negative pair are more likely to be dissimilar, which is beneficial to classifying samples from different classes.

To further understand the underlying mechanism of the inference process, we also randomly selected a sample pair from Cars196 for similarity attribution, as shown in Figure 7 [2]. From top to bottom, we first selected top-128 reliable nodes with high p_i^l scores among 512 nodes and further displayed the two most similar nodes framed in the green dotted box as well as the two most dissimilar ones framed in the red dotted box. Subsequently, we decompose one unreliable node into adjacent related nodes. We observe that similarity nodes with higher sensitivity value λ_i^l are more likely positioned in a higher layer and correspond to clearer concepts such as "headlight", "wheel", and "door" and low-level saliency maps are difficult to distinguish concepts. This demonstrates that high features tend to encode object-level patterns while low features focus on pixel-level patterns. In addition, we discover that nodes and concepts may not correspond to each other one-to-one. For example, multiple nodes may focus on the "wheel" part of cars, which indicates that concepts extracted by CNNs are not well disentangled.

5. Limitations

During evaluation, our AVSL framework needs to maintain a similarity matrix with the spatial complexity of $O(N^2)$ to compute the similarity between two images, where N denotes the number of samples. When dealing with large-scale datasets, we use computation tricks such as matrix slicing to reduce the memory usage to achieve partially parallel computing. This also affects training of proxy-based methods (e.g., the ProxyAnchor loss) when the number of classes are large. On the Stanford Online Products dataset, it is impossible to maintain the similarities between samples in a mini-batch and 11,318 proxies simultaneously on a 24GB-memory device. We thus tailor the loss to only constrain the similarities between samples and positive proxies, which may lead to inferior performance.

[2]More detailed inference graphs of samples from both CUB-200-2011 and Cars196 are included in the supplemental material.

6. Conclusion

In this paper, we have presented an attributable visual similarity learning (AVSL) framework to learn a more accurate and interpretable similarity. We adopt a hierarchy consistency as the inductive bias and employ a bottom-up similarity construction and top-down similarity inference method to model the visual similarity, which first estimates the reliability of similarity nodes at a higher level and then rectifies the unreliable ones using the correlated ones in the adjacent lower level. We have conducted experiments on three widely used datasets to demonstrate the superiority of our framework on both accuracy and interpretability. While our framework is motivated by human visual similarity perception, we believe it can also be adapted to other modalities of information such as text and speech for better interpretability, which is an interesting future work.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China under Grant 62125603 and Grant U1813218, and in part by a grant from the Beijing Academy of Artificial Intelligence (BAAI).

References

[1] Timo Ahonen, Abdenour Hadid, and Matti Pietikainen. Face description with local binary patterns: Application to face recognition. *TPAMI*, 28(12):2037–2041, 2006. 1

[2] Binghui Chen and Weihong Deng. Hybrid-attention based decoupled metric learning for zero-shot image retrieval. In *CVPR*, pages 2750–2759, 2019. 1

[3] Lei Chen, Jianhui Chen, Hossein Hajimirsadeghi, and Greg Mori. Adapting grad-cam for embedding networks. In *WACV*, pages 2794–2803, 2020. 2

[4] Weihua Chen, Xiaotang Chen, Jianguo Zhang, and Kaiqi Huang. Beyond triplet loss: a deep quadruplet network for person re-identification. In *CVPR*, pages 403–412, 2017. 1

[5] Yueqi Duan, Lei Chen, Jiwen Lu, and Jie Zhou. Deep embedding learning with discriminative sampling policy. In *CVPR*, pages 4964–4973, 2019. 2, 7

[6] Yueqi Duan, Wenzhao Zheng, Xudong Lin, Jiwen Lu, and Jie Zhou. Deep adversarial metric learning. In *CVPR*, pages 2780–2789, 2018. 2, 7

[7] Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *CVPR*, volume 2, pages 1735–1742, 2006. 2

[8] Ben Harwood, Vijay Kumar BG, Gustavo Carneiro, Ian Reid, and Tom Drummond. Smart mining for deep metric learning. In *ICCV*, pages 2821–2829, 2017. 2

[9] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 6

[10] Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv*, abs/1703.07737, 2017. 1

[11] Junlin Hu, Jiwen Lu, and Yap-Peng Tan. Discriminative deep metric learning for face verification in the wild. In *CVPR*, pages 1875–1882, 2014. 1

[12] Chen Huang, Chen Change Loy, and Xiaoou Tang. Local similarity-aware deep feature embedding. *arXiv*, abs/1610.08904, 2016. 2

[13] Sungyeon Kim, Dongwon Kim, Minsu Cho, and Suha Kwak. Proxy anchor loss for deep metric learning. In *CVPR*, pages 3238–3247, 2020. 2, 6, 7

[14] Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based ensemble for deep metric learning. In *ECCV*, pages 736–751, 2018. 7

[15] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *ICCVW*, pages 554–561, 2013. 2, 6

[16] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *NeurIPS*, pages 1097–1105, 2012. 1

[17] Brenden M Lake, Ruslan Salakhutdinov, and Joshua B Tenenbaum. Human-level concept learning through probabilistic program induction. *Science*, 350(6266):1332–1338, 2015. 1

[18] Ji Lin, Liangliang Ren, Jiwen Lu, Jianjiang Feng, and Jie Zhou. Consistent-aware deep learning for person re-identification in a camera network. In *CVPR*, pages 5771–5780, 2017. 1

[19] Xudong Lin, Yueqi Duan, Qiyuan Dong, Jiwen Lu, and Jie Zhou. Deep variational metric learning. In *ECCV*, pages 689–704, 2018. 7

[20] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv*, abs/1711.05101, 2017. 6

[21] David G Lowe. Distinctive image features from scale-invariant keypoints. *IJCV*, 60(2):91–110, 2004. 1

[22] Jiwen Lu, Gang Wang, Weihong Deng, Pierre Moulin, and Jie Zhou. Multi-manifold deep metric learning for image set classification. In *CVPR*, pages 1137–1145, 2015. 1

[23] Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *ICCV*, pages 360–368, 2017. 2

[24] Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *CVPR*, pages 4004–4012, 2016. 2, 6

[25] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. Deep metric learning with bier: Boosting independent embeddings robustly. *TPAMI*, 42(2):276–290, 2018. 2, 7

[26] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. Pytorch: An imperative style, high-performance deep learning library. *arXiv*, abs/1912.01703, 2019. 6

[27] Qi Qian, Lei Shang, Baigui Sun, Juhua Hu, Hao Li, and Rong Jin. Softtriple loss: Deep metric learning without triplet sampling. In *ICCV*, pages 6450–6458, 2019. 2, 7

[28] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *KDD*, pages 1135–1144, 2016. 2

[29] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *IJCV*, 115(3):211–252, 2015. 6

[30] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, pages 815–823, 2015. 1, 2

[31] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *ICCV*, pages 618–626, 2017. 2

[32] Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *NeurIPS*, pages 1857–1865, 2016. 2

[33] Abby Stylianou, Richard Souvenir, and Robert Pless. Visualizing deep similarity networks. In *WACV*, pages 2029–2037, 2019. 2

[34] Yifan Sun, Changmao Cheng, Yuhan Zhang, Chi Zhang, Liang Zheng, Zhongdao Wang, and Yichen Wei. Circle loss: A unified perspective of pair similarity optimization. In *CVPR*, pages 6398–6407, 2020. 2, 7

[35] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level perfor-

mance in face verification. In *CVPR*, pages 1701–1708, 2014. 1

[36] Nakul Verma, Dhruv Mahajan, Sundararajan Sellamanickam, and Vinod Nair. Learning hierarchical similarity metrics. In *CVPR*, pages 2280–2287, 2012. 2

[37] Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge J Belongie. The Caltech-UCSD Birds-200-2011 dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011. 2, 6

[38] Alvin Wan, Lisa Dunlap, Daniel Ho, Jihan Yin, Scott Lee, Henry Jin, Suzanne Petryk, Sarah Adel Bargal, and Joseph E Gonzalez. Nbdt: neural-backed decision trees. *arXiv*, abs/2004.00221, 2020. 2

[39] Jian Wang, Feng Zhou, Shilei Wen, Xiao Liu, and Yuanqing Lin. Deep metric learning with angular loss. In *ICCV*, pages 2593–2601, 2017. 7

[40] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *CVPR*, pages 5022–5030, 2019. 2, 7

[41] Chao-Yuan Wu, R Manmatha, Alexander J Smola, and Philipp Krahenbuhl. Sampling matters in deep embedding learning. In *ICCV*, pages 2840–2848, 2017. 2, 6, 7

[42] Mike Wu, Michael Hughes, Sonali Parbhoo, Maurizio Zazzi, Volker Roth, and Finale Doshi-Velez. Beyond sparsity: Tree regularization of deep models for interpretability. In *AAAI*, volume 32, 2018. 2

[43] Xinyi Xu, Yanhua Yang, Cheng Deng, and Feng Zheng. Deep asymmetric metric learning via rich relationship mining. In *CVPR*, pages 4076–4085, 2019. 2, 7

[44] Han-Jia Ye, De-Chuan Zhan, Xue-Min Si, Yuan Jiang, and Zhi-Hua Zhou. What makes objects similar: A unified multi-metric learning approach. In *NeurIPS*, pages 1235–1243, 2016. 2

[45] Jason Yosinski, Jeff Clune, Yoshua Bengio, and Hod Lipson. How transferable are features in deep neural networks? *arXiv*, abs/1411.1792, 2014. 2

[46] Jason Yosinski, Jeff Clune, Anh Nguyen, Thomas Fuchs, and Hod Lipson. Understanding neural networks through deep visualization. *arXiv*, abs/1506.06579, 2015. 2

[47] Rui Yu, Zhiyong Dou, Song Bai, Zhaoxiang Zhang, Yongchao Xu, and Xiang Bai. Hard-aware point-to-set deep metric for person re-identification. In *ECCV*, pages 188–204, 2018. 1

[48] Tongtong Yuan, Weihong Deng, Jian Tang, Yinan Tang, and Binghui Chen. Signal-to-noise ratio: A robust distance metric for deep metric learning. In *CVPR*, pages 4815–4824, 2019. 2

[49] Yuhui Yuan, Kuiyuan Yang, and Chao Zhang. Hard-aware deeply cascaded embedding. In *ICCV*, pages 814–823, 2017. 2, 7

[50] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *ECCV*, pages 818–833, 2014. 2

[51] Matthew D Zeiler, Dilip Krishnan, Graham W Taylor, and Rob Fergus. Deconvolutional networks. In *CVPR*, pages 2528–2535, 2010. 2, 3

[52] Quanshi Zhang, Ruiming Cao, Feng Shi, Ying Nian Wu, and Song-Chun Zhu. Interpreting cnn knowledge via an explanatory graph. In *AAAI*, volume 32, 2018. 2, 3

[53] Quanshi Zhang, Ying Nian Wu, and Song-Chun Zhu. Interpretable convolutional neural networks. In *CVPR*, pages 8827–8836, 2018. 2

[54] Wenliang Zhao, Yongming Rao, Ziyi Wang, Jiwen Lu, and Jie Zhou. Towards interpretable deep metric learning with structural matching. In *ICCV*, pages 9887–9896, 2021. 2, 7

[55] Wenzhao Zheng, Zhaodong Chen, Jiwen Lu, and Jie Zhou. Hardness-aware deep metric learning. In *CVPR*, pages 72–81, 2019. 2, 7

[56] Wenzhao Zheng, Chengkun Wang, Jiwen Lu, and Jie Zhou. Deep compositional metric learning. In *CVPR*, pages 9320–9329, 2021. 7

[57] Wenzhao Zheng, Borui Zhang, Jiwen Lu, and Jie Zhou. Deep relational metric learning. In *ICCV*, pages 12065–12074, 2021. 2, 7

[58] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 2, 4

[59] Sijie Zhu, Taojiannan Yang, and Chen Chen. Visual explanation for deep metric learning. *arXiv*, abs/1909.12977, 2019. 2