Prototype-based Dataset Comparison

Nanne van Noord
University of Amsterdam
n.j.e.vannoord@uva.nl

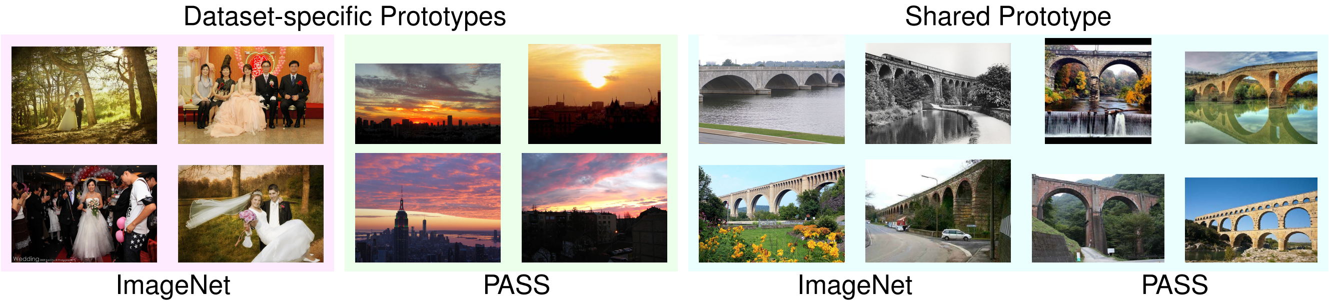

Figure 1: **Prototypes discovered with ProtoSim**. By training on both datasets ProtoSim is able to discover prototypes that are dataset specific or shared. From left to right: ImageNet specific "Wedding", PASS specific "sunset skyline", and shared "viaduct" prototypes.

Abstract

Dataset summarisation is a fruitful approach to dataset inspection. However, when applied to a single dataset the discovery of visual concepts is restricted to those most prominent. We argue that a comparative approach can expand upon this paradigm to enable richer forms of dataset inspection that go beyond the most prominent concepts.

To enable dataset comparison we present a module that learns concept-level prototypes across datasets. We leverage self-supervised learning to discover these prototypes without supervision, and we demonstrate the benefits of our approach in two case-studies. Our findings show that dataset comparison extends dataset inspection and we hope to encourage more works in this direction. Code and usage instructions available at https://github.com/Nanne/ProtoSim

1. Introduction

Image datasets are crucial for Computer Vision and due to the algorithms' need for more data they are ever growing in size. At the same time datasets are a major source of bias leading to negative social impact [29, 30]. Unfortunately, it is challenging to determine what is contained in a dataset as their large size combined with the visual nature of the data makes manual inspection infeasible. To support users and developers of large-scale datasets in ensuring that the datasets match their usage and design goals it is necessary to develop better tools for dataset inspection.

A promising direction for generic dataset inspection is found in highly effective approaches that have been proposed for summarisation [35, 9, 14, 44, 24]. A major benefit of these approaches is that they enable explorative dataset inspection without needing supervised pretraining. However, a limitation of these approaches is that they use frequency as a proxy for importance, and on a single dataset therefore only discover those visual concepts which are most prominent. As such, we argue that a comparative approach, which enables discovery of a wider and more diverse range of concepts, is necessary to effectively perform dataset inspection. For instance, the PASS dataset [1] is designed as an ImageNet [32] alternative whilst containing no people, as such it provides us with a testable hypothesis that is comparative in nature. Namely, when comparing these two datasets there should be a disjoint set of visual people-centric concepts that are only found in ImageNet. In a case study we will verify this hypothesis, and demonstrate that dataset comparison can lead to new insights.

Moreover, an additional limitation of existing summarisation approaches is that they decouple the summarisation process from representation learning, and treat these as two distinct steps by performing the summarisation on a pre-defined feature basis, such as GIST descriptors in [35] or LDA clusters in [31]. As recent work on incorporating prototypes into a network's reasoning process has been shown to aid interpretability [23, 6, 33, 13, 45], we propose that this may also be a promising direction for end-to-end dataset comparison. Therefore, we introduce a method for prototype-based dataset comparison, which discovers the visual concepts in a dataset in an end-to-end fashion.

Dataset prototypes are similar to cluster centroids in that they represent latent points in the feature space that are to be discovered. However, in clustering, similar to prior work on dataset summarisation, these centroids are discovered in a step that is decoupled from learning the feature representations. Whilst some recent works have explored forms of deep clustering [40, 11, 22], they still involve two separate optimisation goals. For example, by having one loss focused on the feature representation and another on clustering [11]. Instead, we propose a simple module *ProtoSim* that can be integrated into a deep network without changing how it is optimised. To demonstrate this, we add ProtoSim to a Vision Transformer (ViT) [10] and show that it can effectively discover visual concepts across datasets in a self-supervised setting.

Overall, we make the following contributions:

- We introduce dataset comparison, a new approach for inspecting datasets.
- To enable dataset comparison we present ProtoSim, a module for integrated learning of dataset prototypes.
- With two case-studies we demonstrate how dataset comparison can be used to gain new insight into datasets.

2. Related work

Dataset comparison through prototypes is a largely unexplored area of research, but it is similar in spirit to dataset distilliation [47, 5] which seeks to distill the knowledge from a large dataset into a small dataset. Most works in dataset distillation generate synthetic examples, as opposed to selecting a set of data instances, to construct the small dataset. These synthetic examples could be considered dataset prototypes, as they represent an aggregate of information that is central to the dataset. A key difference between such prototypes and the prototypes considered in this work, is that the distilled synthetic examples are instance-level prototypes (i.e., complete images that may represent multiple concepts), whereas in this work the aim is to discover concept-level prototypes, which may correspond to global concepts or part-level concepts.

In the remainder of this section we will further explore how dataset comparison relates to dataset summarisation, prototype learning, and deep clustering.

2.1. Dataset Summarisation

Manual browsing is a reliable manner to determine what is contained in a dataset, however, browsing even a subset of the data can be time-consuming, which has led to a line of work on dataset summarisation [31, 9, 14, 44, 24]. The aim of dataset summarisation is to find underlying structure or patterns that are shared between data instances to give an overview of what is contained in a dataset.

An important modeling choice for dataset summarisation is the basis on top of which the patterns are found. Work on mid-level pattern mining aims to find discriminative and representative patches and use these as visual primitive for further analysis [34]. Mid-level pattern mining aims to distinguish itself from "visual word" approaches [36], which were found to mainly capture low-level patterns such as edges and corners [34, 9]. Two notable patch-based approaches are the dataset fingerprinting approach by [31] and the *what makes Paris look like Paris?* approach by [9]. Patch-based approaches aim to circumvent the limitations of visual words by starting with larger visual structure (i.e., patches at multiple scales). Due to the limitations of pixel-space representations patches cannot be used directly for analysis, instead initial patch-based approaches used feature extractors such as HOG [34, 9] and Convolutional Neural Network (CNN) activations in later works [24]. Using patches has been shown to help steer the results towards more semantically meaningful visual elements, yet they are still restrictive in representational power and in terms of shape (i.e., patches must be rectangular). Moreover, although the starting point is a patch, the features extracted largely determine what the pattern represents and the features are not intrinsically explainable, necessitating post-hoc explanations. In contrast, we propose to add ProtoSim to a ViT architecture, which can learn to recognise visual elements of arbitrary shapes and sizes, whilst still focusing on semantically meaningful concepts.

2.2. Prototype Learning

Prototype-based approaches can be categorised into two areas, the first aims to learn prototypes which represent high-level semantic concepts (*i.e.*, classes) to support zero or few-shot learning [18, 28, 37]. The second area is concerned with finding recurring visual elements and are increasingly used for model explainability [23, 6, 21, 39, 13, 20, 42, 45]. As opposed to post-hoc explainability methods, prototypes are integral to the network and provide explanations of the network's reasoning process [6]. Our approach is part of this second area and we will focus the discussion on integral prototypes for recurring visual elements.

Integral prototypes have been shown to be effective for supervised multi-class classification [6, 13, 20, 21], deepfake detection [39], and semantic segmentation [48]. In the literature we can recognise two branches of work on integral prototypes, the first builds on ProtoPnet [23, 6] and the second on Concept Bottleneck Models (CBM) [13, 20]. Common across these approaches is that they add a layer preceding the output layer that maps the extracted features to prototypes (or concepts) and produce the final output based on the affinity to these prototypes. The logic is thus that because the prototypes are recognisable entities, the prediction is explained by the affinity to the prototypes. The main difference between these branches is that in ProtoPnet-like approaches the prototypes are latent vectors in the embedding space that are learnt end-to-end, whereas CBM approaches use a pre-defined concept library.

However, these prior approaches all focus on a supervised setting and learn per-class prototypes or concepts, which is severely limiting for dataset summarisation. Instead, we propose to leverage a self-supervised loss to learn prototypes that are not class specific and can represent *any* visual concept that is present in the data. In particular, our prototypes may represent class-level concepts or segment-level concepts as in CBM [13], but we learn them without concept-level supervision. Previous works learned prototypes for the features at each location in the convolutional output of a Convolutional Neural Network (CNN) backbone [6, 21, 39]. This restricts the spatial extent of the prototypes to size of the receptive field. Additionally, in CNN the output is typically spatially averaged to achieve a global image representation, however, as this is not done for CNN-based prototype approaches they can only learn local and not global prototypes. Instead, we use Vision Transformers (ViT) [10], as they offer solutions for both aforementioned issues capturing local and global concepts of various spatial extents.

2.3. Deep Clustering

Clustering relates to learning prototypes or concepts in that it has been used to define the library of concept CBM [13], and to an extent the underlying principles match the goal of clustering. In particular, in deep clustering [40, 11, 22] the centroids are learned iteratively, resulting in them being conceptually close to prototypes or CBM-concepts. However, in prior works on deep clustering [40, 11, 22] the clustering objective is separate from the feature representation, whereas in prototype-based approaches the final output is determined by projecting the input to the prototype subspace.

Another key difference is that by making the prototypes integral to the reasoning process, as in CBM [13] or ProtoPnet [6], it is ensured that the prototypes are of meaningful to the final output. Post-hoc clustering, or as part of a separate objective, may lead to overrepresentation of clusters which capture meaningless information, *e.g.*, orthogonal to the objective that was used to obtain the feature representation.

3. Dataset Comparison

Given a collection of datasets $X = \{\chi_i\}_{i=1}^{|X|}$ the aim of prototype-based dataset comparison is to discover a set of prototypes $p \in \mathbb{R}^{K \times D}$ that represent visual concepts that may occur predominantly in only one dataset (*i.e.* dataset-specific prototypes) or visual concepts which occur in two or more datasets (*i.e.* shared prototypes). An illustration of the dataset comparison workflow is shown in Figure 2.

We argue that a comparative approach to dataset summarisation, *i.e.*, dataset comparison is vital for gaining insights into datasets. In particular, if C_* is the set of *all* possible visual concepts, and C_i the visual concepts in ImageNet, then $C_i \subset C_*$. Given these sets, C_i is a perfect description of what is contained in ImageNet, and the difference $C_* - C_i$ perfectly describes what is not contained in ImageNet. However, as we do not have access to C_* and we have no guaranteed manner of discovering C_i, we can at best learn a set of prototypes $\hat{C}_i$ that approximates C_i.

As it is an approximation it cannot be concluded from $\hat{C}_i$ that a concept is not in C_i. For example, PASS is designed to not contain humans (i.e., no humans in C_p), but based on $\hat{C}_p$ it cannot be stated with certainty that no humans are found in PASS, only that no human-centric concepts were discovered. To overcome this limitation we propose to jointly learn $\hat{C}_i$ and $\hat{C}_p$. Thereby, if we know a concept is in $\hat{C}_i$ and not in $\hat{C}_p$, we can reasonably conclude that the concept is not in C_p either.

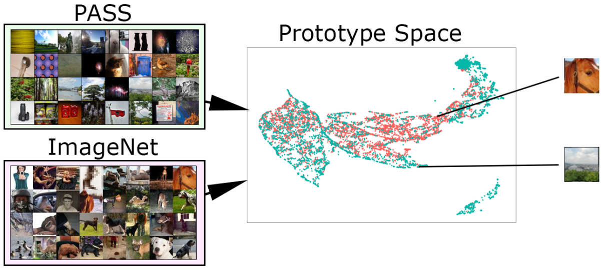

Figure 2: Overview of the dataset comparison workflow. Datasets are mapped to the prototype embedding space, from where we can inspect and compare individual prototypes.

Through dataset comparison we are thus able to answer questions that could not be answered by only considering a single dataset, thereby creating new possibilities for dataset inspection for hypothesis verification or comparative exploration.

3.1. Prototype Evaluation

How to evaluate the prototypes learnt for dataset comparison (or summarisation) is an open problem, as the ground-truth for which visual concepts are contained in the dataset

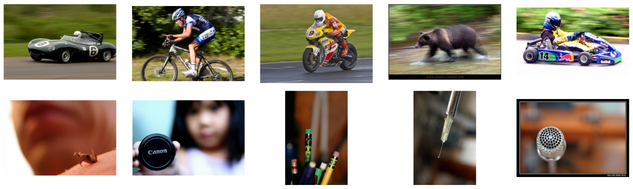

Figure 3: Examples of low-level visual properties discovered with ProtoSim: motion blur (top) and a shallow depth of field (bottom).

are unknown. Nonetheless, we can leverage the comparative approach by verifying whether the prototypes found match the design goals of the datasets. For instance, as PASS was designed not to contain humans we can explore whether the human-centric prototypes learnt from ImageNet indeed do not occur in PASS. This manner of evaluation forms the basis of our first case study.

Generally, we consider the prototypes p to be of high-quality when they are *distinct* and *meaningful*. Distinct prototypes are able to independently represent a visual concept, this is in contrast to, for example, sparse coding [46] where the aim is to find basis vectors that represent the input through linear combinations. The set of prototypes is meaningful when it can be used to discriminate images within and across the datasets from one another. Requiring that the prototypes are discriminative helps avoid trivial solutions. We can quantitatively measure how discriminative the prototypes are by evaluating them with downstream tasks.

To ensure that the prototypes are distinct we design our method ProtoSim to perform hard assignment of the prototypes, and by optimising the prototypes with contrastive SSL we ensure that they are discriminative. This alignment between the design of the method (more details in Section 4) and the goal of dataset comparison helps us make certain the prototypes are of high-quality.

3.2. Types of Prototypes

Within the set of prototypes we can recognise different types of prototypes depending on how they occur across the datasets. For instance, a prototype p_i may be considered *dataset-specific* when over 95% of its occurrences are only found in a single dataset. By definition, any prototype which is not dataset-specific would then be considered *shared*. However, various degrees of 'sharedness' may occur depending on the overlap between datasets.

In theory it may be possible for the datasets in X to be fully distinct, resulting in only dataset-specific prototypes. In practice, we find that prototypes may represent various basic visual properties (see Figure 3), resulting in shared prototypes even when the datasets differ semantically.

4. Prototype-based Learning

We propose *ProtoSim* a simple module that can inserted in ViT architectures to enable dataset comparison through integral prototype-based learning. ProtoSim differs from previous prototype layers in three ways, firstly, the prototypes learned with ProtoSim are distinct (as opposed to linear mixtures), secondly, ProtoSim is specifically designed for ViT architectures and learns prototypes from a set of tokens, rather than a single embedding vector. Lastly, the prototypes in ProtoSim are not class-specific, instead we learn a single pool of prototypes that may occur in any image.

4.1. ProtoSim

In order to learn which visual concepts are present in a dataset we aim to optimise a set of K learnable prototypes $p \in \mathbb{R}^{K \times D}$. By designing the prototypes to be part of a prototype layer (*i.e.* ProtoSim) we can directly optimise them in and end-to-end fashion as part of ViT that we add the layer to. Specifically, given the token embeddings $z \in \mathbb{R}^{N+1 \times D}$ produced by a ViT, ProtoSim maps these to the prototype embeddings $\hat{z} \in \mathbb{R}^{N+1 \times D}$, where D is the token vector size. During this mapping each token is replaced with the most similar prototype in p.

In [23, 6] the similarity to each of the class-specifics prototypes is determined by calculating the squared L^2 distance and selecting the closest prototype with a max operation. Because our aim is to find prototypes that recur in the entire dataset, instead of finding class-specific prototypes, we propose a surprisingly simple formulation of this process by using a reversed version of dot-product attention [25] to efficiently calculate attention across all the prototypes:

$$a = softmax(pz^{\intercal}) \tag{1}$$

$$\hat{z} = a^{\intercal} p. \tag{2}$$

Here the attention mask a represents the soft-assignment of prototypes to tokens, as such each token embedding $\hat{z}$ is a linear combination of prototypes. This is not desirable as our aim is to learn distinct prototypes which independently represent a visual concept, as this enables better summarisation and greater interpretability. To obtain a hard assignment we replace the softmax operation in Equation 1 with gumbel-softmax [17]:

$$a = gumbel\text{-}softmax(pz^{\intercal}). \tag{3}$$

For simplicity we drop the temperature parameter, as it empirically does not meaningfully influence performance and can be kept fixed at 1. The benefit of gumbel-softmax is that it maintains gradient-flow whilst enabling hard attention. Based on this modification a now represents a hard assignment matrix which is used to replace each token with a prototype. As a visual concept can occur across multiple spatial locations there is no constraint on how often a prototype can be assigned.

4.2. Backbone

Whilst ProtoSim functions as a generic network module, with the only requirement being an a two-dimensional input of $K \times D$, we specifically apply it to ViT in this work as these offer some benefits over CNN. In particular, in ViT all tokens can share information with all other tokens, as such there is no restriction on the spatial extent of the learned prototypes. Moreover, whilst most tokens in a ViT represent some spatial region of the image (*i.e.* patch tokens), the class token takes on the role of a global image representation. This allows us to learn prototypes that are local (*i.e.* only assigned to patch tokens) and global (*i.e.* only assigned to the class token) without modifying the architecture.

ProtoSim is placed at the end of the backbone ViT in the network architecture. Formally, the input $x \in \mathbb{R}^{H \times W \times C}$ is divided into flattened patch tokens $x_p \in \mathbb{R}^{N \times (P^2 \cdot C)}$, where H, W, and C are the height, width, and channel dimensionality of the input and P the patch size hyperparameter. In addition to N patch tokens, a learnable *class* token x_{class} is prepended to input. No class labels are used during training, but we retain the ViT terminology of referring to this as a class token. After the embedding phase the tokens are passed through a series of transformer blocks resulting in the token embeddings $z \in \mathbb{R}^{N+1 \times D}$, where D is the embedded token vector size. z is passed through the ProtoSim layer resulting in the prototype embeddings $\hat{z}$.

4.3. Training Objective

The training objective determines the prototypes. For instance, a classification-based objective results in prototypes that are indicative of the classes (as in ProtoPnet [6] or CBM [20]). Yet, such an objective may result in a poor coverage of concepts that are not part of the annotated classes. Because we strive for good coverage we focus on a Contrastive Self-Supervised Learning (Contrastive SSL) objective. In particular, because Constrastive SSL optimises for fine-grained discrimination between images (rather than classes) and has been shown to be perform well on a variety of downstream tasks [7].

From the recently proposed Contrastive SSL methods [16, 15, 7, 8, 3, 4], only DINO [4] has been specifically designed to work with ViT. As such, we minimise the DINO loss as the training objective:

$$\min_{\theta_s} \sum_{x \in \{x_1^g, x_2^g\}} \sum_{\substack{x' \in V \\ x' \neq x}} H(P_t(x), P_s(x')), \tag{4}$$

where $H(a, b) = -a \log b$, x_1^g and x_2^g are two large crops from the input, V a set of data-augmented smaller crops, and P_t and P_s are the teacher and student networks respectively, each with a ProtoSim layer added to the backbone. We refer to the excellent work by [4] for further details.

5. Case Studies

In this section we present two case studies, the first is a comparison between two datasets (*i.e.*, two-way dataset comparison) aimed at determining how the PASS [1] dataset differs from ImageNet [32]. Because the design goal of PASS was to present an alternative for ImageNet that does not contain humans we test whether this is achieved successfully. The second case study focuses on a three-way dataset comparison scenario wherein we explore how three artwork datasets differ. These three datasets were designed for different tasks and contain images from different museum collections, as such based on these meta properties of the datasets it is challenging to determine how they actually relate. With these two case studies we put the two main applications of dataset comparison to the test, verifying a design goal of a dataset, and exploring unknown datasets. In addition, in Supplementary Section A we demonstrate how ProtoSim can also be used on a single dataset.

5.1. Datasets

For the first case study we focus on ImageNet and PASS, we briefly describe both:

- **ImageNet** [32] is a widely used dataset available in two versions: ImageNet-21K and ImageNet-1K. In this work we focus on ImageNet-1K consisting of approximately 1.2 million training images and 1000 classes. Despite how widely used it is, and how influential it has been, ImageNet-1K has also been criticised for its biased depictions of persons [30].
- **PASS** [1] is a recently proposed alternative for ImageNet with the intention of not containing persons to avoid the issues in ImageNet. PASS contains approximately 1.4 million unlabeled images and can only be used for self-supervised training.

In the second case study we focus on three artwork datasets that have been used for a variety of different tasks:

- **MET** [43] is a dataset designed for instance-level recognition, its training set contains over 400k images from more than 224k different objects held by the Metropolitan Museum of Art in New York.
- **Rijksmuseum** [27] presents a set of tasks which are museum-centric, focusing on predicting various types of metadata (*i.e.*, artist, object-type, material usage, and creation year). It consists of 110k images obtained from collections held by the Rijksmuseum in Amsterdam.
- **SemArt** [12] consists of images collected from the Web Gallery of Art a virtual museum with predominantly European fine-art. The SemArt dataset contains

21k images described with similar metadata as the Rijksmuseum dataset, but in addition also has textual descriptions in the form of artistic comments.

The available metadata for these three artwork datasets differs, but notably none of these datasets have been described with semantic classes as, for example, found in ImageNet. As such, we do not have prior information about what type of visual concepts are represented in the artwork datasets.

5.2. Experimental setup

The backbone used is the DeIT-S model [38] with a patch size of 16×16. The parameters of the backbone are fixed during prototype learning and training is done on a single NVIDIA RTX 3090 with a batch size of 128 and learning rate of $5\mathrm{e}{-5}$ for 20 epochs. For the first 15 epochs we train with soft gumbel-softmax, after which we switch the student to the hard 'straight-through' gumbel-softmax, to ensure more independent prototypes. In line with previous works (*e.g.*, [2, 26]) we set K to 8192.

For quantitative evaluation we follow the standard procedure of training a linear classifier on frozen features [4, 16]. We freeze the backbone and the ProtoSim layer and then train a linear classifier for 20 epochs with a learning rate of 0.001 and a batch size of 256.

5.3. Two-way Dataset Comparison

For the two-way dataset comparison case study we focus on the PASS and ImageNet datasets, to this end we learn prototypes on the union of these two datasets, which we refer to as the *PassNet* dataset, and present the results in this section. All results in this section are obtained with ProtoSim trained on top of the DINO backbone pretrained on ImageNet. A key finding of this comparison is that ProtoSim learns both dataset-specific prototypes (i.e., prototypes that predominantly activate for one dataset) and shared prototypes (i.e., prototypes that occur frequently in both datasets). In Figure 4 we show nine prototypes which are predominantly found in ImageNet, PASS, or are shared equally among both datasets. Additional examples of prototypes are shown in the supplementary material.

The prototypes that are predominantly found in ImageNet contain people (*i.e.*, over 99% of images for which they activate are found in ImageNet). However, despite its design goal we do find persons for these prototypes in PASS. The *Persons at the beach* prototype (first row) contains a man holding a beach-ball photographed from the back, and the *Uncovered legs/arms* prototype (third row) is found in (nude) statues and a faded Polaroid of a person. The only two PASS images for the *Sports-ball* prototype (second row) just depict the field and a ball, suggesting that the prototype captures more than just persons.

The prototypes that are predominantly found in PASS depict concepts or objects that are not found among the 1000 ImageNet categories. Nonetheless, these predominantly PASS prototypes can be found in some ImageNet images where the annotated object is a (small) part of the image. From left to right for row four to six we find the following ImageNet categories: European Gallinule, Ant, Wheelbarrow, Turnstile, Barn, and Headland. We would describe the prototypes in these rows as capturing *Flowers*, a *View down narrow alley*, and *Sunset with silhouetted foreground* respectively. In general, the PASS prototypes that predominantly concern vista or landscape images, which on occasion contain objects from ImageNet categories.

For the prototype in the second row of Figure 4 we measure how it is spatially distributed and visualise this for the four ImageNet images in Figure 5. Visualising the attention maps of this prototype shows that the main activations are indeed for the leg and arm areas of the persons depicted. Other parts of the persons receive much less attention than these uncovered areas. Although this is a predominantly ImageNet prototype, and is therefore commonly found in instances of the 'miniskirt' and 'bikini' categories, it is also found in PASS, including in images of mannequins, artworks, and anime of primarily feminine characters.

Overall, we can conclude that the two-way comparison is highly successful, not only are we able to verify that depictions of humans in PASS are practically non-existent, but by edge-cases we are able to demonstrate the strength of the prototypes. Moreover, based on the discovered prototypes we also gain insights into how these datasets differ, with PASS containing many more vistas and landscapes, whereas ImageNet is strongly object-centric. As such we can conclude that while there is an overlap, PASS differs in more respects from ImageNet than just the presence of humans - which may cast doubt on its status as an ImageNet replacement.

Table 1: Performance for linear classifier on ImageNet.

Train set	Backbone	K	Accuracy
ImageNet	DINO [4]	8192	64.8
PASS	PASS-DINO [1]	8192	20.4
PassNet (I+P)	DINO [4]	8192	53.2
PassNet (P+I)	PASS-DINO [1]	8192	15.9
ImageNet	DINO [4]	1024	54.7
ImageNet	DINO [4]	4096	63.7

Quantitative analysis. We perform two quantitative analyses. Firstly, we explore the loss value of each model during training and compare it to the diversity of the prototypes, as measured by the average cosine distance between the prototype weights. Secondly, we train a linear classifier on top of each of the backbones to perform ImageNet clas-

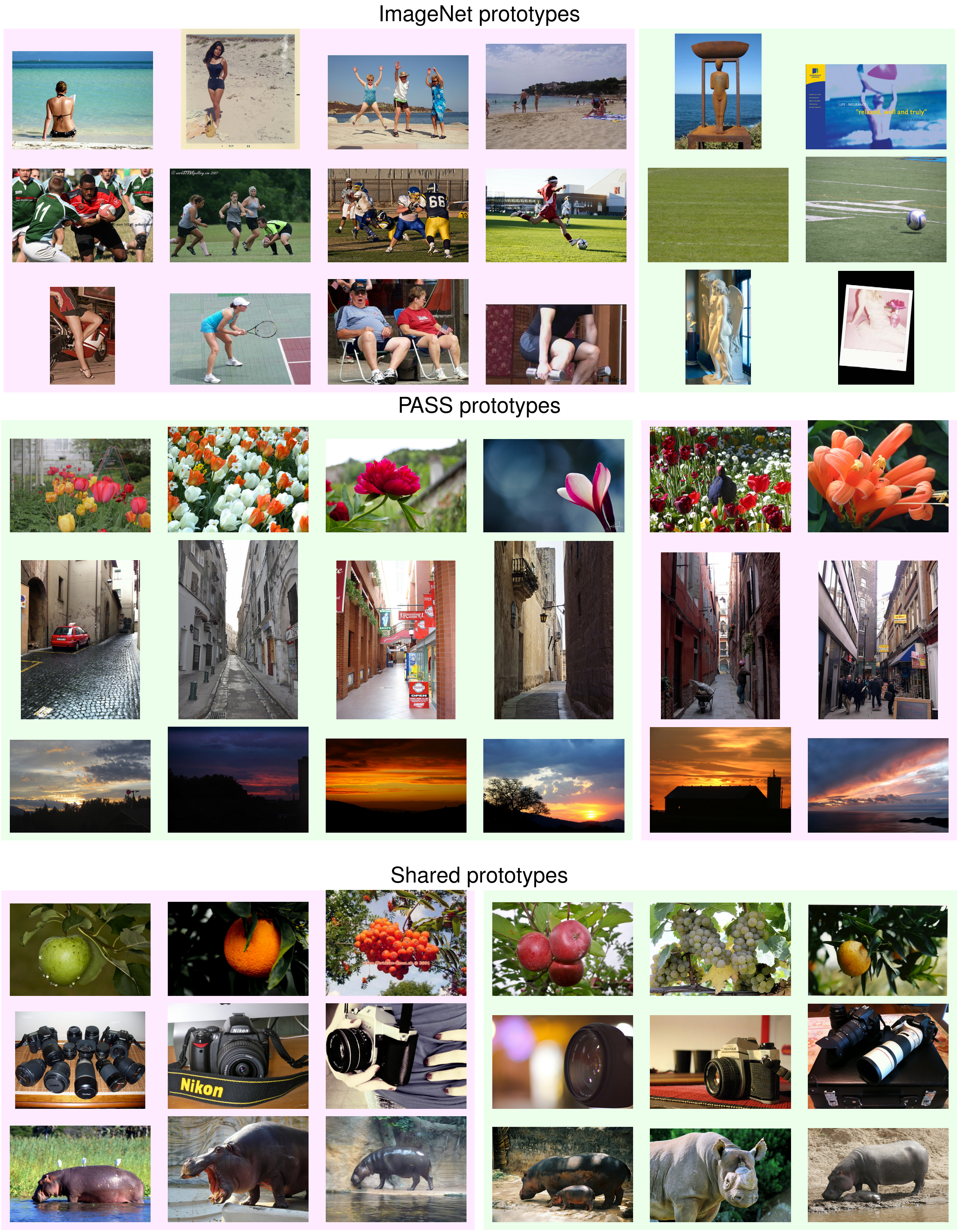

Figure 4: Example images for nine prototypes, each three rows showing predominantly ImageNet, predominantly PASS, and Shared prototypes respectively. The last two columns for rows one through six show some of the few examples found in the other dataset. The last three rows show prototypes common in both with examples of ImageNet in the left three columns, and examples in the right three.

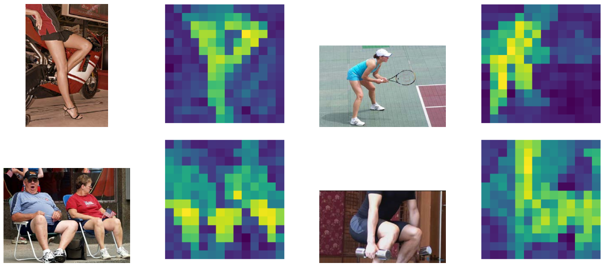

Figure 5: Attention maps for the for the four ImageNet images for the *Uncovered legs/arms* prototype in Figure 4.

sification. For these evaluations we compare between two different backbones, one pre-trained on ImageNet [4] and one pre-trained on PASS [1]. Results reported on PassNet are described as either (I+P) for the ImageNet pre-trained backbone, or P+I for the PASS pre-trained backbone.

Table 1 shows the performance on the ImageNet validation set for a linear classifier trained on top of fixed backbones. We observe that pre-training on ImageNet (unsurprisingly) leads to the best downstream performance on ImageNet, and that training on PASS leads to lower performance. Similarly, for the two backbones we find that the ImageNet pre-trained backbone performs much better. Nonetheless, we see degraded performance for training on PassNet versus ImageNet. The last two rows of Table 1 show that increasing the K value benefits performance, due to computational complexity and diminishing returns no values above 8192 were tested. In conclusion, our quantative evaluation confirms our modelling choices of using the ImageNet pre-trained backbone with 8192 prototypes, and we will use this configuration for the second case study as well.

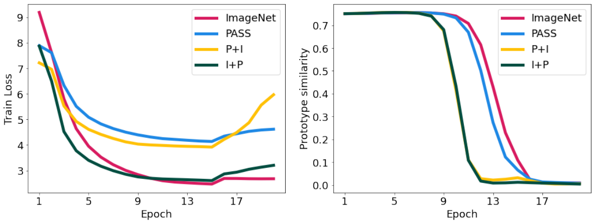

Figure 6: Training loss (left) and average cosine similarity (right) during prototype training. PassNet results P+I are obtained PASS-DINO backbone and I+P with the DINO backbone.

The results in Figure 6 show the training loss and the average cosine similarity between prototypes as training progresses. In both plots we see two groupings of two lines that follow the same pattern. For all models we can see that switching to the hard gumbel-softmax leads to a bump in the loss, but except for P+I this bump levels off quickly.

5.4. Three-way Dataset Comparison

In the second case study we perform a three-way dataset comparison between the MET, Rijksmuseum, and SemArt datasets. The aim of this comparison is highly exploratory, unlike ImageNet, none of these datasets have pre-defined classes, as such it is not obvious what visual concepts are represented. In Figure 7 we show nine prototypes discovered with ProtoSim, the first three pairs of prototypes are dataset-specific and are predominantly found in the MET, Rijksmuseum, and SemArt respectively. These prototypes were randomly selected from the most dataset-specific prototypes and give a good indication of what is contained in these datasets. The MET dataset contains many 3D objects and often multiple photographs are taken from different angles, as such the first prototype represents *Top-down views of vases*. The second MET prototype depicts *Egpytian wall carvings*, which were photographed under raking light to emphasise the height differences across the surface.

The Rijksmuseum dataset contains many prints or drawings on paper, whilst this is something that can be learnt from the material type overview in [27], the two prototypes depicted show two different groups of visual concepts. The first Rijksmuseum prototype depicts *natural history drawings*, whereas the second is found in *side view portraits*. SemArt on the other hand consists of mostly fine-art paintings, with the first prototype capturing a stylistic visual concept of *Renaissance paintings*, the second prototype is a more concrete semantic concept, namely *dogs*.

Just by exploring the dataset-specific datasets we already get a much better understanding of what is contained in these datasets and what makes them unique. In addition, when we then analyse the shared prototypes we can see that these differences largely persists across different visual concepts. The first shared prototype is of various *animals*, notably we see that the prototype is not sensitive to material differences, returning both drawings and paintings. The same observation is made for the second shared prototype of *baskets*, which is matched to drawings, paintings, and photographs of actual baskets. This ability to generalise across modalities is also observed in the third shared prototype of *brass instruments*, which matches actual instruments and realistic paintings containing similar instruments.

Based on the three-way dataset comparison we find that there are degrees of overlap between the datasets, generally we would see that the MET and Rijksmuseum datasets are most similar in types of objects, where MET has a greater focus on physical objects and Rijksmuseum on prints and drawings. Because these two datasets have a relatively small proportion of paintings the similarity with the SemArt dataset is less, nonetheless, we can observe that in terms of visual concepts these datasets represent similar things. In Appendix Section C we show additional prototypes discovered for these artwork datasets.

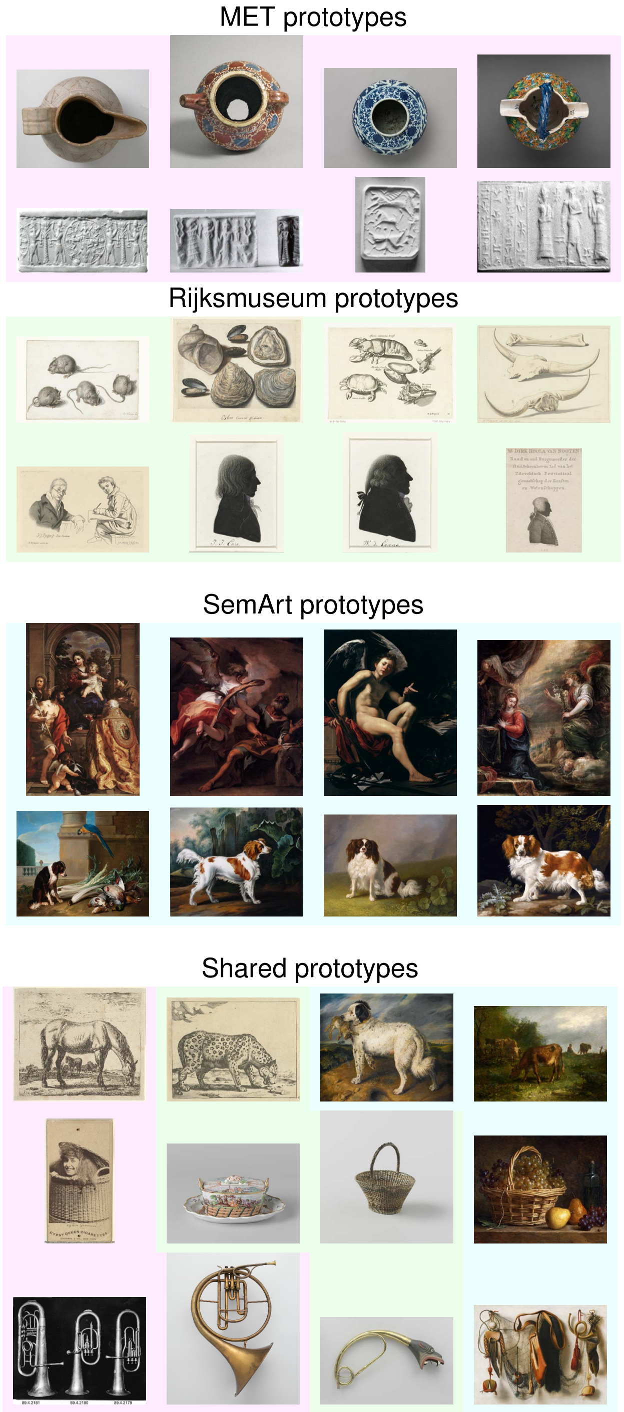

Figure 7: Example images for nine prototypes, the first six rows are divided in three pairs of prototypes predominantly found in MET, Rijksmuseum, or SemArt respectively. The last three rows show prototypes common across the datasets. Examples with a pink, green, and blue background belong to the MET, Rijksmuseum, or Semart datasets respectively.

6. Discussion

The prototypes obtained with ProtoSim enable dataset comparison, nonetheless, there are two limitations. Firstly, although explanation methods based on visual examples are preferred for visual data [19], there is a mandatory manual inference step to determine what a prototype represents. Whilst previous methods suffer from the same limitation [6, 21] ProtoSim does offer an advantage in that the attention maps can be visualised (i.e., Figure 5). Based on example images and their attention maps it becomes possible to be relatively certain about what the prototype represents.

Secondly, whilst the ViT backbone can be replaced, for our experiments we found the best results with an ImageNet pre-trained backbone. Since this dataset is one of the ones being compared, but also because this dataset has known biases [41] the choice for a ImageNet pre-trained backbone may influence the prototypes learned. During experiments no such influence was observed, instead it appears ProtoSim is flexible in learning diverse prototypes, nonetheless further investigation is necessary.

7. Conclusion

In this work we presented dataset comparison as a new direction for dataset inspection. To enable dataset comparison across large-scale datasets we introduce ProtoSim which leverages integral prototype-based learning and self-supervised learning to discover visual concepts across datasets. To evaluate our proposed approach we perform two case studies, in the first we compare the PASS and ImageNet datasets, and in the second we we perform a three-way comparison between artwork datasets.

Based on these case studies we find that we can gain new insights into the datasets. In the first case study we find that ImageNet indeed contains many more images with persons, which is in line with the design goal of PASS. However, we still discovered partial and non-photorealistic depictions of persons in PASS, which were not discovered in its person-focused dataset curation process. When comparing the artwork datasets we find that each has a unique focus, but also that various semantic concepts are shared between them.

Overall, we presented an initial exploration of dataset comparison that will hopefully lead to greater attention for this topic, as it is becoming increasingly necessary to improve dataset inspection techniques.

References

[1] Yuki M Asano, Christian Rupprecht, Andrew Zisserman, and Andrea Vedaldi. PASS: An ImageNet replacement for self-supervised pretraining without humans. In *NeurIPS Track on Datasets and Benchmarks*, page 26, 2021. 1, 5, 6, 8

[2] Hangbo Bao, Li Dong, and Furu Wei. BEiT: BERT Pre-Training of Image Transformers. In *Iclr*, 2022. 6

[3] Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. Unsupervised Learning of Visual Features by Contrasting Cluster Assignments. In *NeurIPS*, 2020. 5

[4] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging Properties in Self-Supervised Vision Transformers. In *Iccv*, 2021. 5, 6, 8

[5] George Cazenavette, Tongzhou Wang, Antonio Torralba, Alexei A. Efros, and Jun-Yan Zhu. Dataset Distillation by Matching Training Trajectories. In *Cvpr*, 2022. 2

[6] Chaofan Chen, Oscar Li, Chaofan Tao, Alina Jade Barnett, Jonathan Su, and Cynthia Rudin. This Looks Like That: Deep Learning for Interpretable Image Recognition. In *NeurIPS*, Dec. 2019. 2, 3, 4, 5, 9

[7] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A Simple Framework for Contrastive Learning of Visual Representations. In *Icml*, page 11, 2020. 5

[8] Xinlei Chen and Kaiming He. Exploring Simple Siamese Representation Learning. In *Cvpr*, 2021. 5

[9] Carl Doersch, Saurabh Singh, Abhinav Gupta, Josef Sivic, and Alexei A Efros. What Makes Paris Look like Paris? *ACM Transactions on Graphics (SIGGRAPH)*, 31(4):101:1–101:9, 2012. 1, 2

[10] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations*, Sept. 2020. 2, 3

[11] Boyan Gao, Yongxin Yang, Henry Gouk, and Timothy M. Hospedales. Deep clustering with concrete k-means, Oct. 2019. 2, 3

[12] Noa Garcia and George Vogiatzis. How to Read Paintings: Semantic Art Understanding with Multi-modal Retrieval. In Laura Leal-Taixé and Stefan Roth, editors, *Computer Vision – ECCV 2018 Workshops*, volume 11130, pages 676–691. Springer International Publishing, Cham, 2019. 5

[13] Amirata Ghorbani, James Wexler, James Zou, and Been Kim. Towards Automatic Concept-based Explanations. In *NeurIPS*, Oct. 2019. 2, 3

[14] Sonal Goel, Sarthak Ahuja, A V. Subramanyam, and Ponnurangam Kumaraguru. #VisualHashtags: Visual Summarization of Social Media Events Using Mid-Level Visual Elements. In *Proceedings of the 25th ACM International Conference on Multimedia*, pages 1434–1442, Mountain View California USA, Oct. 2017. Acm. 1, 2

[15] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre H Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap Your Own Latent A New Approach to Self-Supervised Learning. In *NeurIPS*, page 14, 2020. 5

[16] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum Contrast for Unsupervised Visual Representation Learning. In *Cvpr*, Mar. 2020. 5, 6

[17] Iris A. M. Huijben, Wouter Kool, Max B. Paulus, and Ruud J. G. van Sloun. A Review of the Gumbel-max Trick and its Extensions for Discrete Stochasticity in Machine Learning. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 01:1–1, Oct. 2021. 4

[18] Saumya Jetley, Bernardino Romera-Paredes, Sadeep Jayasumana, and Philip Torr. Prototypical Priors: From Improving Classification to Zero-Shot Learning. In *Bmvc*, 2015. 2

[19] 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. In *NeurIPS*, page 12, 2020. 9

[20] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept Bottleneck Models. In *Icml*, Dec. 2020. 2, 3, 5

[21] Stefan Kraft and IT-Designers Group. SPARROW: Semantically Coherent Prototypes for Image Classification. In *Bmvc*, page 12, 2021. 2, 3, 9

[22] Junnan Li, Pan Zhou, Caiming Xiong, and Steven C. H. Hoi. Prototypical Contrastive Learning of Unsupervised Representations, Mar. 2021. 2, 3

[23] 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 *Aaai*, page 8, 2018. 2, 3, 4

[24] Yao Li, Lingqiao Liu, and Chunhua Shen. Mid-Level Deep Pattern Mining. In *Cvpr*, page 10, 2015. 1, 2

[25] Thang Luong, Hieu Pham, and Christopher D. Manning. Effective Approaches to Attention-based Neural Machine Translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal, 2015. Association for Computational Linguistics. 4

[26] Chengzhi Mao, Lu Jiang, Mostafa Dehghani, Carl Vondrick, Rahul Sukthankar, and Irfan Essa. Discrete Representations Strengthen Vision Transformer Robustness. In *Iclr*, 2022. 6

[27] Thomas Mensink and Jan Van Gemert. The rijksmuseum challenge: Museum-centered visual recognition. In *Proceedings of International Conference on Multimedia Retrieval*, pages 451–454, 2014. 5, 8

[28] Pascal Mettes. Hyperspherical Prototype Networks. In *NeurIPS*, page 11, 2019. 2

[29] Amandalynne Paullada, Inioluwa Deborah Raji, Emily M. Bender, Emily Denton, and Alex Hanna. Data and its (dis)contents: A survey of dataset development and use in machine learning research. *Patterns*, 2(11):100336, Nov. 2021. 1

[30] Vinay Uday Prabhu and Abeba Birhane. Large image datasets: A pyrrhic win for computer vision? In *Wacv*, 2021. 1, 5

[31] Konstantinos Rematas, Basura Fernando, Frank Dellaert, and Tinne Tuytelaars. Dataset fingerprints: Exploring image collections through data mining. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4867–4875, Boston, MA, USA, June 2015. Ieee. 2

[32] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy,

Aditya Khosla, and Michael Bernstein. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115(3):211–252, 2015. 1, 5

[33] Dawid Rymarczyk, Łukasz Struski, Jacek Tabor, and Bartosz Zieliński. ProtoPShare: Prototype Sharing for Interpretable Image Classification and Similarity Discovery. In *Acm Sigkdd*, 2021. 2

[34] Saurabh Singh, Abhinav Gupta, and Alexei A. Efros. Unsupervised Discovery of Mid-Level Discriminative Patches. In Andrew Fitzgibbon, Svetlana Lazebnik, Pietro Perona, Yoichi Sato, and Cordelia Schmid, editors, *Eccv*, volume 7573, pages 73–86. Springer Berlin Heidelberg, Berlin, Heidelberg, 2012. 2

[35] Pinaki Sinha, Sharad Mehrotra, and Ramesh Jain. Summarization of personal photologs using multidimensional content and context. In *Proceedings of the 1st ACM International Conference on Multimedia Retrieval - ICMR '11*, pages 1–8, Trento, Italy, 2011. ACM Press. 1, 2

[36] Sivic and Zisserman. Video Google: A text retrieval approach to object matching in videos. In *Proceedings Ninth IEEE International Conference on Computer Vision*, pages 1470–1477 vol.2, Nice, France, 2003. Ieee. 2

[37] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical Networks for Few-shot Learning. In *NeurIPS*, page 11, 2017. 2

[38] 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, 2021. 6

[39] Loc Trinh, Michael Tsang, Sirisha Rambhatla, and Yan Liu. Interpretable and Trustworthy Deepfake Detection via Dynamic Prototypes. In *2021 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1972–1982, Waikoloa, HI, USA, Jan. 2021. Ieee. 2, 3

[40] Bo Yang, Xiao Fu, Nicholas D Sidiropoulos, and Mingyi Hong. Towards K-means-friendly Spaces: Simultaneous Deep Learning and Clustering. In *International Conference on Machine Learning*, 2017. 2, 3

[41] Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, page 547–558, Jan 2020. arXiv:1912.07726 [cs]. 9

[42] Chih-Kuan Yeh, Been Kim, Sercan O. Arik, Chun-Liang Li, Tomas Pfister, and Pradeep Ravikumar. On Completeness-aware Concept-Based Explanations in Deep Neural Networks. In *NeurIPS*, 2020. 2

[43] Nikolaos-Antonios Ypsilantis, Noa Garcia, Guangxing Han, Sarah Ibrahimi, Nanne Van Noord, and Giorgos Tolias. The Met Dataset: Instance-level Recognition for Artworks. In *NeurIPS Track on Datasets and Benchmarks*, 2021. 5

[44] Licheng Yu, Mohit Bansal, and Tamara L. Berg. Hierarchically-Attentive RNN for Album Summarization and Storytelling. In *Emnlp*, Aug. 2017. 1, 2

[45] Mert Yuksekgonul, Maggie Wang, and James Zou. Post-Hoc Concept Bottleneck Models. In *Iclr*, 2023. 2

[46] Tianzhu Zhang, Bernard Ghanem, Si Liu, Changsheng Xu, and Narendra Ahuja. Low-Rank Sparse Coding for Image Classification. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 281–288, 2013. 4

[47] Bo Zhao and Hakan Bilen. Dataset Condensation with Distribution Matching. *arXiv:2110.04181 [cs]*, Oct. 2021. 2

[48] Tianfei Zhou, Wenguan Wang, Ender Konukoglu, and Luc Van Goo. Rethinking Semantic Segmentation: A Prototype View. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2572–2583, New Orleans, LA, USA, June 2022. Ieee. 2