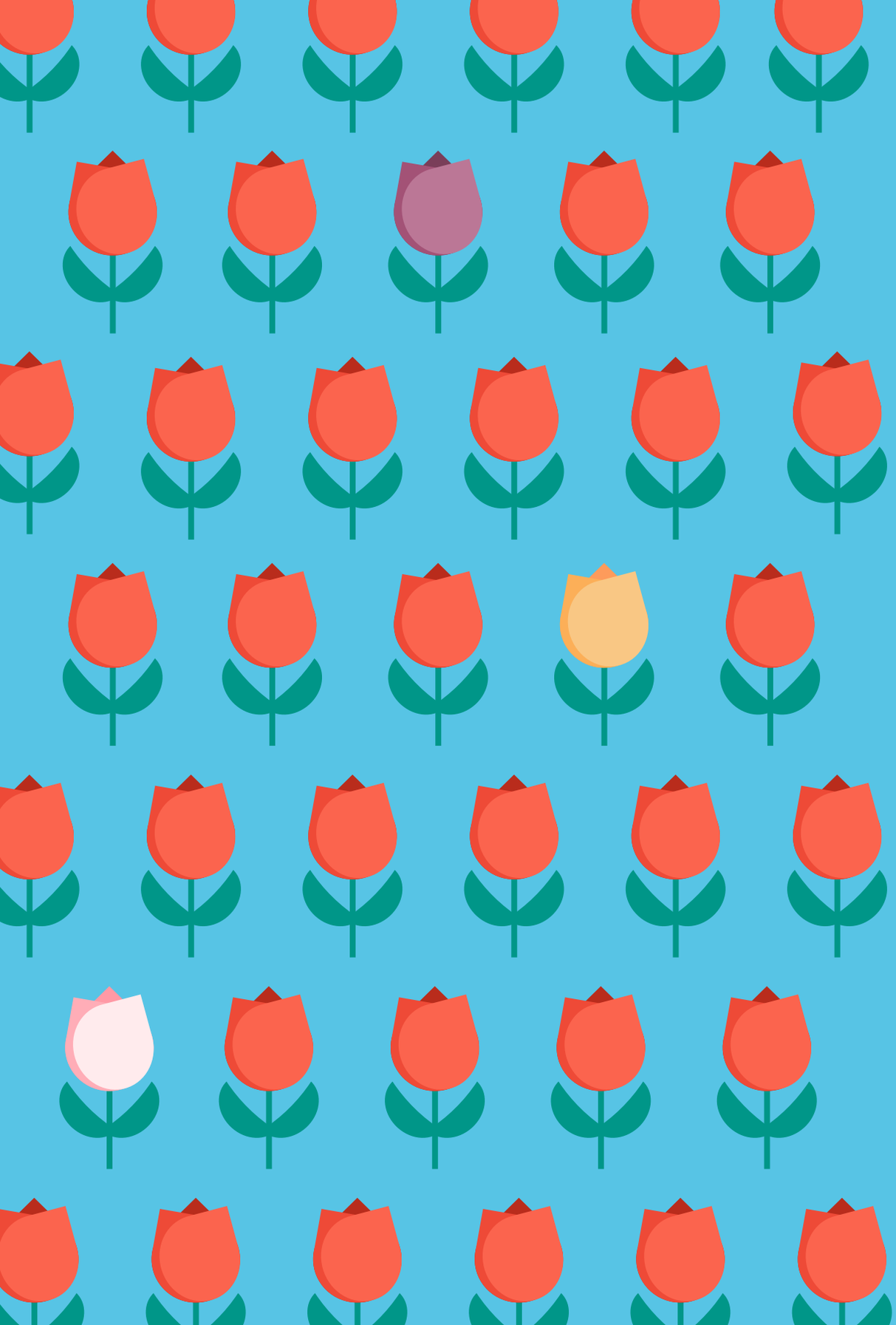
Learning visual similarities robust to bias

Thong, W.

Publication date
2022
Document Version
Final published version



Learning Visual Similarities Robust to Bias

William Thong

Learning Visual Similarities Robust to Bias

William Thong

Learning Visual Similarities Robust to Bias

William Eric Thong

This book was typeset by the author using LaTeX 2_ε.

Learning Visual Similarities Robust to Bias

Academisch Proefschrift

ter verkrijging van de graad van doctor
aan de Universiteit van Amsterdam
op gezag van de Rector Magnificus

prof. dr. G.T.M. ten Dam

ten overstaan van een
door het College voor Promoties ingestelde commissie,
in het openbaar te verdedigen in de Agnietenkapel
op woensdag 7 september 2022, te 10.00 uur

door

William Eric Thong

geboren te Rueil-Malmaison

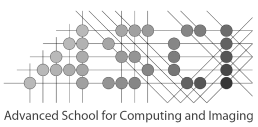
Promotiecommisie

Promotor:	prof. dr. C.G.M. Snoek	Universiteit van Amsterdam
Co-promotor:	prof. dr. ir. A.W.M. Smeulders	Universiteit van Amsterdam
Overige leden:	prof. dr. M. Worring	Universiteit van Amsterdam
	prof. dr. M. de Rijke	Universiteit van Amsterdam
	dr. S. Ghebreab	Universiteit van Amsterdam
	prof. dr. N. Sebe	Università degli studi di Trento
	dr. T.E.J. Mensink	Google Research

Faculteit der Natuurwetenschappen, Wiskunde en Informatica

UNIVERSITEIT VAN AMSTERDAM

The work described in this thesis has been carried out at the Video and Image Sense lab of the University of Amsterdam. Partial funding was provided by the Natural Sciences and Engineering Research Council of Canada (NSERC), with reference number BESD3-488844-2016. This thesis is supported by the Advanced School for Computing and Imaging (ASCI).



Contents

Chapter 1

Introduction

Our environment is constantly changing, and nothing is set in stone. This challenges known and preconceived assumptions, given that they are no longer enough to describe our environment. For example, when a new species is discovered, our scientific knowledge needs to incorporate this discovery and make associations with existing species. Models of our environment must, therefore, encompass the ability to generalize beyond fixed assumptions. Such generalization abilities favor a more robust understanding and interpretation of our changing environment.

Similarity associations with what is already familiar helps to cope with our changing environment [Bar, 2009]. Indeed, digesting a completely new concept without any mechanism to rely on prior knowledge can be overwhelming and inefficient. When encountering an unseen situation, we should then ask and learn to answer the question "what is this *like*?". There exists, however, a trade-off on how much we should rely on similarity associations to avoid being biased towards scenarios previously encountered. When an artist releases a new creation, it is always helpful to know to which art movement it belongs to and how it relates to existing creations in the art scene. Making these associations better helps to ascertain the motivation and impact of the new creation. Still, depending too much on these associations can damage judgement and lead to misinterpretations that would be unfair to the artwork.

In the computer vision context of this thesis, where machines make sense of images, similarity associations among images amount, nowadays, to learning an embedding space. Images exhibit a very high variability, and computing a similarity metric in this high-dimensional image space can lead to erroneous interpretations, originating from the curse of dimensionality [Goodfellow et al., 2016]. It becomes more adequate to learn a mapping function from the image space to a lower-dimensional space to build and infuse similarity associations. If two images refer to the same concept, they should then be close to each other in the embedding space, regardless of their appearances, properties, or backgrounds. Distances in the embedding space then act as a similarity association metric and should be robust to potential adverse biases that may arise in images.

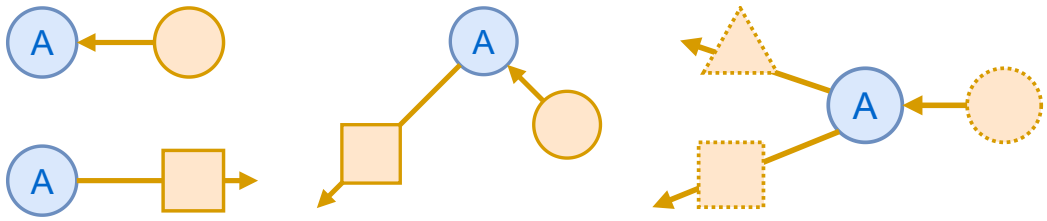
(a) A dog and a plush dog refer to the same concept *dog*.

(b) Coffees with different levels of milk refer to the same concept *coffee*.

(c) Hands with different skin colors refer to the same concept *hand*.

Figure 1.1: **Appearance changes for a similar concept.** Computer vision models should not be affected by appearance changes, and assimilate the fact that different appearances, attributes or properties refer to the same concept. In this thesis, we learn visual similarities to cope robustly with biases originating from these appearance changes. From left to right, photos by Camille Paralisan, Nathan Dumlao, and Shane Rounce on Unsplash (Unsplash license).

Learning visual similarities opens the door for novel opportunities and a different reasoning in categorization tasks. Instead of categorizing images into a fixed set of labels, visual similarities strive to learn semantically meaningful image associations through an embedding space. Consider Figure 1.1, which illustrates three examples where different appearances correspond to similar concepts. Figure 1.1a depicts two dogs with different forms: one is a life form while the other one is a plush toy. Even though they correspond to two very different domains, they still refer to the same concept *dog*. In this context, it is necessary to understand that different mediums or domains of representation can convey a similar message. Figure 1.1b showcases various coffee types, where the level of milk acts as an attribute of the concept *coffee*. While the level of milk modifies the visual appearance, all depicted cups remain coffee cups. In other words, these images exhibit a high variance in terms of the milk amount but should still be similar to each other as they refer to the same concept *coffee*. Different compositions can then refer to a similar concept. Furthermore, no sufficient description exists to encompass all variations that define a category [Wittgenstein, 1953]. There will always be exceptions to the definition, as a barista could come up with a new way to make coffee or a new drink that contains coffee. This requires an ability to generalize to unseen compositions or unseen concepts, given the similarities with seen



(a) Pairwise (b) Triplet (c) Prototype

Figure 1.2: **Methods for learning visual similarities.** A common solution relies on pairwise (a) or triplet (b) comparisons, where an anchor representation is compared with a positive (*same shape*) or a negative (*different shape*) representation. If the comparison is positive, they should be pulled together; otherwise, they should be pushed away from each other. One hurdle is the need for sampling positive and negative samples. In this thesis, we instead rely on prototype comparisons (c), where the anchor is compared with prototype representations in the embedding space (*dotted shape*). This removes the need for sampling and enables learning visual similarities with novel and robust properties.

ones. Figure 1.1c portrays the diversity in human skin color. When a protected attribute – such as skin color, age or gender – is present, it becomes necessary to ensure computer vision models do not suffer from algorithmic bias. Indeed, they shouldn't reproduce or amplify social biases present in the training dataset. Predictions should be fair and mitigate potential discrimination that may arise. This notably translates into having similar representations when referring to the same concept, regardless of the variation in protected attribute values.

The main challenge in learning visual similarities resides in deriving an objective function to learn a mapping function from the image space to the embedding space that is robust to biases. A common solution is to rely on pairwise [Chopra et al., 2005, Hadsell et al., 2006] (Figure 1.2a) or triplet [Schroff et al., 2015] (Figure 1.2b) comparisons, where an anchor image is compared to either a positive or a negative sample, or both, in the embedding space. Such an objective requires a sampling step, which limits the versatility of the training procedure as it can get quite convoluted when multiple similarities need to be taken into account. Furthermore, a bag of tricks is usually needed to facilitate the training and improve performance [Hermans et al., 2017]. We instead rely on prototype theory [Rosch, 1978], a theory in cognitive psychology which assumes that there exist central members of every category. These central members, referred to as prototypes, better capture the definition of a concept as they are easier and faster to recognize. In the visual similarity context, this amounts to learning or defining vector representation in the embedding, which acts as a prototype representation for every category (Figure 1.2c). Learning a meaningful embedding space then consists in mapping images of a similar concept to their corresponding prototype, which makes the training procedure much simpler compared with pairwise or triplet

comparisons. Indeed, the sampling step is now removed as prototypes provide a global view of all categories for comparisons. Different variations of prototype representations can also be derived to robustify the embedding space. For example, prototypes can be fixed to leverage inductive priors, or learned as any other parameters of the model to provide freedom for arranging the embedding space.

In this thesis, we rely on the learning of visual similarities to robustly cope with different appearance changes for a similar concept. We explore how learning visual similarities can benefit computer vision models, which need to work across domains, generalize to seen and unseen attribute compositions or categories, and overcome algorithmic bias. We formulate the following main research question:

How to learn visual similarities robust to bias?

In Chapter 2, we investigate the recognition of categories across domains. When learning a model for object classification or retrieval, a domain bias arises from the gap in the different domain representations. For example, the representation of a "dog" differs among photographs, paintings on a canvas, or pencil drawings. We formulate the following question:

How to learn visual similarities robust to domain bias?

Where previous works address this domain bias via a domain adaptation loss to make features domain invariant (*e.g.*, Shen et al. [2018], Yelamarthi et al. [2018]), we propose to learn domain-specific mapping functions to a common embedding space. The embedding space corresponds to a visual similarity space with a pre-trained semantic meaning where every category is represented by the word vector of its name. In other words, we fix the category prototype to a word vector representation. Training then consists of pulling inputs to their corresponding representation and pushing them away from other class representations with a cosine embedding loss. Experimentally, we confirm that a common embedding space benefits standard cross-domain search tasks with two domains. Furthermore, we show how this approach can open the search to multiple domains with novel search scenarios, which would have been intractable with previous methods. We conclude that relying on a visual similarity space common to all domains is an effective approach to bridge the domain bias.

In Chapter 3, we study the retrieval of specific attribute combinations for multiple categories. Designers create new products by composing attributes and categories. For example a new fashionable "vest" could be one with *oversized shoulderpads* and *pastel colors* attributes. We pose the research question:

How to learn visual similarities robust to compositional bias?

Products exhibit multiple similarities. Indeed, products are instances of particular categories while attributes characterize their visual properties. When searching for specific products, the visual representations need to capture instance, category and attribute similarities to retrieve seen and unseen product compositions. Where previous works usually address these similarities individually (*e.g.*, Frome et al. [2007], Kovashka et al. [2012], Song et al. [2016]), we integrate them altogether in an embedding space in an interrelated manner for product search. Training relies on a diverse supervision of instance, category and attribute labels for every visual sample. Every label has its own similarity loss function where interrelatedness is handled by spanning similarity comparisons either only in a single subspace or multiple ones. The evaluation reveals the importance of every similarity for retrieving product composition: attributes matter more for clothes images while categories are more important for car images. Furthermore, having such an interrelated visual similarity space with diverse labels enables the exploration of product trends to discover typical, atypical and eclectic products. We conclude that modeling the interrelation of instance, category and attribute through visual similarities facilitates the search of seen and unseen product compositions.

In Chapter 4, we investigate how to recognize novel categories without dampening the ability to recognize the ones seen during training. Image classifier exhibit a confidence bias as predictions tend to be overconfident towards the categories seen during training. In return, when shown a sample of an unseen category, the classifier most likely predicts it as a seen class. We formulate the research question:

How to learn visual similarities robust to confidence bias?

Scientists regularly discover new species. One way to describe them is to define a common set of characteristics common to all species but still discriminative enough to distinguish them. For example, while both "horses" and "zebras" have *hoofed feet*, *long heads* and *manes*, a "zebra" differs by the *striped coat*. We can then rely on this common set of characteristics to recognize new categories. However, relying too much on this set of characteristics for classification hurts the performance of existing models in generalized zero-shot learning. Where previous works address this bias by separating the classification for seen and unseen categories (*e.g.*, Atzmon and Chechik [2019], Liu et al. [2018]), we consider both jointly to address the confidence bias. We map inputs to a label embedding space where every category is represented by a fixed attribute vector. Visual similarity then consists in mapping inputs close to their corresponding attribute representation. We control the confidence of seen and unseen with temperature and a bidirectional entropy regularization of the probabilities. The evaluation shows the effectiveness of our approach to mitigate the confidence bias for several existing models, and

such for characteristics described via attributes or sentences. Furthermore, we show that the confidence bias is also dataset-dependent as not all datasets suffer to the same extent. We conclude that addressing the confidence bias with visual similarities benefits existing models in generalized zero-shot learning.

In Chapter 5, we address adverse predictions in image classifiers. As society becomes aware of new potential harms, models should also be assessed to understand whether their predictions can result in adverse decisions. For example while more *women* tend to wear "earrings" than *men*, an image classifier should not rely on the *gender* of the person to detect the presence of "earrings" in facial portraits. We pose the research question:

How to learn visual similarities robust to algorithmic bias?

The presence of spurious correlations creates an algorithm bias and they should be identified, mitigated, and measured in image classifiers to avoid any potential discrimination or amplification of biases. Where previous works consider either the feature space (*e.g.*, Alvi et al. [2018]) or the label space (*e.g.*, Wang et al. [2020b]) for algorithmic bias, we show that both matter for bias identification and mitigation. We identify a bias direction in the feature space, which indicates that common classifiers encode the bias implicitly. We mitigate the algorithmic bias by creating separate discriminative label embedding spaces for binary protected attributes. During training, inputs are mapped to their specific and separate embedding space, and visual similarities to their corresponding latent class representation are maximized with a cosine embedding loss. Once trained, we further apply a bias removal operation in the feature space. Experimentally, we show the effectiveness of our approach for algorithmic bias mitigation in both multi-class and multi-label classifications. We conclude that reducing the bias direction in the feature space, as well as deriving label embedding spaces for classification helps in mitigating the algorithmic bias from spurious correlations.

Tackling different biases through visual similarities makes computer vision models more robust. This is a step towards models able to adapt constantly to changing environment where changes can arise through multiple forms.

1.1 List of publications

For every chapter of this thesis, we here declare the authors' contributions.

Chapter 2

William Thong, Pascal Mettes and Cees G. M. Snoek (2020). "Open Cross-Domain Visual Search". *Computer Vision and Image Understanding*, 200:103045. [Thong et al., 2020]

- William Thong — All aspects
- Pascal Mettes — Guidance and technical advice
- Cees G. M. Snoek — Supervision and insight

Chapter 3

William Thong and Cees G. M. Snoek (2022). "Diversely-Supervised Visual Search". *ACM Transactions on Multimedia Computing, Communications, and Applications*, 18(1):1–22. [Thong and Snoek, 2022]

- William Thong — All aspects
- Cees G. M. Snoek — Supervision and insight

Chapter 4

William Thong and Cees G. M. Snoek (2020). "Bias-Awareness for Zero-Shot Learning the Seen and Unseen". In *British Machine Vision Conference.* [Thong and Snoek, 2020]

- William Thong — All aspects
- Cees G. M. Snoek — Supervision and insight

Chapter 5

William Thong and Cees G. M. Snoek (2021). "Feature and Label Embedding Spaces Matter in Addressing Image Classifier Bias". In *British Machine Vision Conference.* [Thong and Snoek, 2021]

- William Thong — All aspects
- Cees G. M. Snoek — Supervision and insight

The author has further contributed to the following publications during his doctoral studies:

- Mehmet Ozgur Turkoglu, William Thong, Luuk Spreeuwers and Berkay Kicanaoglu (2019). A Layer-Based Sequential Framework for Scene Generation with GANs. In *AAAI Conference on Artificial Intelligence.* [Turkoglu et al., 2019]

- Sarah Ibrahimi, Shuo Chen, Devanshu Arya, Arthur Camara, Yunlu Chen, Tanja Crijns, Maurits van der Goes, Thomas Mensink, Emiel van Miltenburg, Daan Odijk, William Thong, Jiaojiao Zhao, Pascal Mettes (2019). Interactive Exploration of Journalistic Video Footage through Multimodal Semantic Matching. In *ACM Multimedia.* [Ibrahimi et al., 2019]

- Pascal Mettes, William Thong and Cees G.M. Snoek (2021). Object Priors for Classifying and Localizing Unseen Actions. *International Journal of Computer Vision*, 129(6):1954–1971. [Mettes et al., 2021]

- William Thong, Jose Costa Pereira, Sarah Parisot, Ales Leonardis and Steven McDonagh (2022). Content-Diverse Comparisons Improve Image Quality Assessment Learning. *Submitted.* [Thong et al., 2022]

Chapter 2

Open Cross-Domain Visual Search

2.1 Introduction

This chapter aims for visual category search across domains. The task is to retrieve visual examples from a specific category in one domain, given a query from another domain. For example, we may want to retrieve *images* of an "airplane" from a quickly-drawn *sketch*. Cross-domain visual search has made considerable progress, showing the possibility to retrieve natural images [Eitz et al., 2010, Sangkloy et al., 2016] or 3D shapes [Li et al., 2013, 2014a,b] from sketches. Different from existing works, which emphasize retrieval from a single source domain to a single target domain, we open the search beyond two domains. The motivation for a search among many domains is that in practice, categories come in many forms [Li et al., 2017, Peng et al., 2019, Wilber et al., 2017]. Hence, we may have queries from several source domains, or want to search with any possible combination of source and target domains. For example, we may now want to combine a *sketch* and a *clipart* of an "airplane" to retrieve *photograph* samples, or use a *clipart* of an "airplane" to retrieve *3D shapes*. In this chapter, we strive for such an open setting: we visually search for categories from any source domain to any target domain, with the ability to search from and within multiple domains simultaneously.

Within cross-domain visual search, an important challenge is the gap between source and target domains [Chen et al., 2019, Dey et al., 2019, Dutta and Akata, 2019, Shen et al., 2018, Xie et al., 2017, Yelamarthi et al., 2018]. Given the inherent difference in representations, reducing the domain gap is an intuitive solution. Both Shen et al. [2018] and Yelamarthi et al. [2018] have highlighted the importance of domain adaptation losses for cross-domain search, especially when searching for unseen categories. Yet, relying on domain adaptation methods

Published in *Computer Vision and Image Understanding*, 200:103045. [Thong et al., 2020]

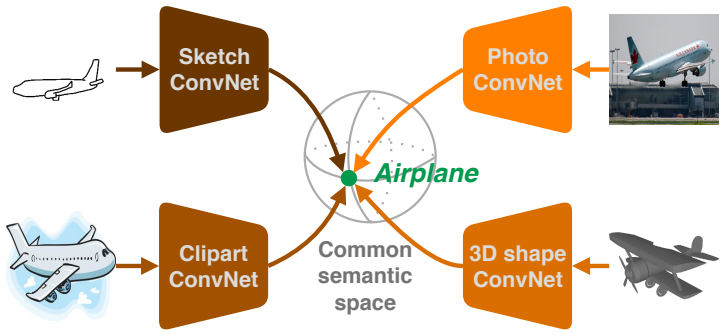


Figure 2.1: **Open cross-domain visual search.** We search for categories from any number of source domains to any number of target domains. Mapping examples to a common semantic space enables any possible combinations of domains when searching for categories.

makes the search unsuited for an open setting by design, due to the requirement of pairwise domain training. As a consequence, opening the search to many domains creates new challenges as (i) all domains should to be mapped to a unique embedding space, and (ii) new domains should be able to be added continuously in an efficient fashion. We address the challenges of open cross-domain visual search.

Inspired by recent works on prototype-based embedding spaces [Movshovitz-Attias et al., 2017, Snell et al., 2017, Wen et al., 2016], we introduce prototype learners for cross-domain visual search in an open setting. Prototype learning has shown to simplify model training and improve performance for image retrieval [Movshovitz-Attias et al., 2017, Wen et al., 2016] and classification [Snell et al., 2017] problems in a low-shot setting. In this work, we leverage prototype learners to perform visual search across multiple domains simultaneously. We define prototypes to unite all domains. Inputs from every domain are mapped to a common semantic space, where every learner is domain-specific and is trained separately. During training, the semantic space is defined by categorical prototypes, corresponding to word embeddings of category names. Learning then consists of regressing inputs to their corresponding categorical prototype in this common semantic space, as illustrated in Figure 2.1. Query representations for search are further refined with neighbours from other domains through a spherical linear interpolation operation. Once trained, the proposed formulation allows us to search among any pair of domains. Since all domains are now aligned in the common semantic space, this enables a search from multiple source domains or in multiple target domains. Lastly, new domains can be added on-the-fly, without retraining previous models.

Empirically, we first demonstrate the ability to perform open cross-domain visual search, highlighting new applications and search possibilities, *i.e.* (i) a search between any pair of source and target domains without hassle; (ii) a search from

multiple source domains; and (*iii*) a search in multiple target domains. Second, while designed for the open cross-domain setting, our approach also works in the conventional closed settings, allowing for comparisons to current approaches. We compare to sketch-based image and 3D shape retrieval, usually considered separately in the literature. We show the versatility of our approach to handle them. Across three well-established tasks totalling seven benchmarks, we obtain state-of-the-art results, highlighting the effectiveness of focusing solely on the semantic space for cross-domain search.

Contributions. Our main contribution is the introduction of open cross-domain visual search. We open the search to many domains, with the ability to retrieve categories from and among any number of domains. To achieve this, we introduce a simple prototype learner for each domain to learn a common semantic space efficiently. Empirically, solely relying on semantic prototypes turns into an effective solution for cross-domain visual search in both newly proposed open settings and existing closed settings. All code and setups are released to foster further research in open cross-domain visual search[1].

2.2 Related work

We first cover related work in cross-domain search, where a large body of works focuses on retrieving natural images or 3D shapes from sketches. We then review relevant work addressing multiple domains and on how to learn semantic spaces with prototype learners.

Cross-domain image search. Sketch-based image retrieval has been a topic of vision community interest for a long time [Jacobs et al., 1995, Kato, 1992]. The seminal work of Eitz et al. [2010] established the first benchmark for its evaluation, which led to the construction of common descriptors for sketches and images, such as bag-of-features [Eitz et al., 2010], bag-of-regions [Hu et al., 2011], histogram of oriented gradients [Hu and Collomosse, 2013], or specialized descriptors for edges [Saavedra, 2014]. With the resurgence of convolutional networks, the dominant approach has shifted towards the learning of a joint semantic space of sketches and images. Qi et al. [2016] learn a joint embedding with a Siamese network while Bui et al. [2017] rely on a triplet network. Bui et al. [2018] add a classification head with a multi-stage training to make features even more discriminative. In all these works, the semantic spaces model categories implicitly, as they rely on sample-based methods such as the Siamese [Chopra et al., 2005, Hadsell et al., 2006] or triplet [Schroff et al., 2015, Weinberger and Saul, 2009] losses to learn cross-domain visual similarities. In this chapter, we explicitly define semantic representations for every category in the embedding space. This

[1]Source code is available at https://github.com/twuilliam/open-search

removes the need for sampling and mining of cross-domain pairs, resulting in a much simpler training procedure.

Sketch-based image retrieval is also considered as a zero-shot learning problem [Shen et al., 2018, Yelamarthi et al., 2018]. In this context, a common approach is to bridge the domain gap between sketches and images. Shen et al. [2018] fuse sketch and image representations with a Kronecker product, while Yelamarthi et al. [2018] introduce domain confusion with generative models to produce domain-agnostic features. Dey et al. [2019] combine gradient reversal layers with metric learning losses to extract the mutual information from both domains. Dutta and Akata [2019] tie the semantic space with visual features from both domains by learning to generate them while Dutta and Biswas [2019] prefer to separate them explicitly. Alternatively, Liu et al. [2019] preserve the knowledge from a pre-trained model to avoid features to drift away during training. Hu et al. [2018a] have also explored how to synthesize classifiers derived from sketches for few-shot image classification. By focusing on domain adaptation, current approaches are optimized to map from a single specific source domain to a single specific target domain. Instead, we consider cross-modal image search from any number of source domains to any number of target domains.

Cross-domain 3D shape search. Searching for 3D shapes from a sketch has been accelerated by the SHREC challenges [Li et al., 2013, 2014a,b]. A common approach is to transform the 3D shape search into an image search problem by projecting the unaligned 3D shape into multiple 2D views [Su et al., 2015]. In this regard, the main methodological approach is to learn a joint embedding space of sketches and 2D view renderings of the unaligned 3D shapes. Wang et al. [2015] map both sketches and shapes in a similar feature space with a Siamese network, while Tasse and Dodgson [2016] learn to regress to a semantic space with a ranking loss. Dai et al. [2017] correlate both sketch and 3D shape representations to bridge the domain gap. Xie et al. [2017] employ the Wasserstein distance to create a barycentric representation of shapes. Qi et al. [2018] apply loss functions on the probabilistic label space rather than the feature space. Chen et al. [2019] propose an advanced sampling of 2D views for the unaligned shapes. Learning cross-domain visual similarities with Siamese or triplet losses typically requires a multi-stage training or negative sampling schemes. A prototype learner removes this requirement, and enables the addition of new domains without the need for retraining existing models.

Searching beyond two domains. Using multiple domains has been investigated in unsupervised domain adaptation [Csurka, 2017, Peng et al., 2017] and unsupervised domain generalization [Blanchard et al., 2011], where the task is to classify unlabeled target samples by learning a classifier on labeled source samples. As such, Peng et al. [2019] illustrate how challenging classification becomes when multiple domains are considered. A new challenge then arises as classifiers have to be designed to benefit from the inherent gap among multiple domains [Carlucci

et al., 2019, Dou et al., 2019, Peng et al., 2019, Xu et al., 2018, Zhuo et al., 2019]. In this chapter, we focus on a different multi-domain task: we consider cross-domain retrieval where category labels are present for both source and target domains, and where the main challenge is to learn a common embedding space for all domains.

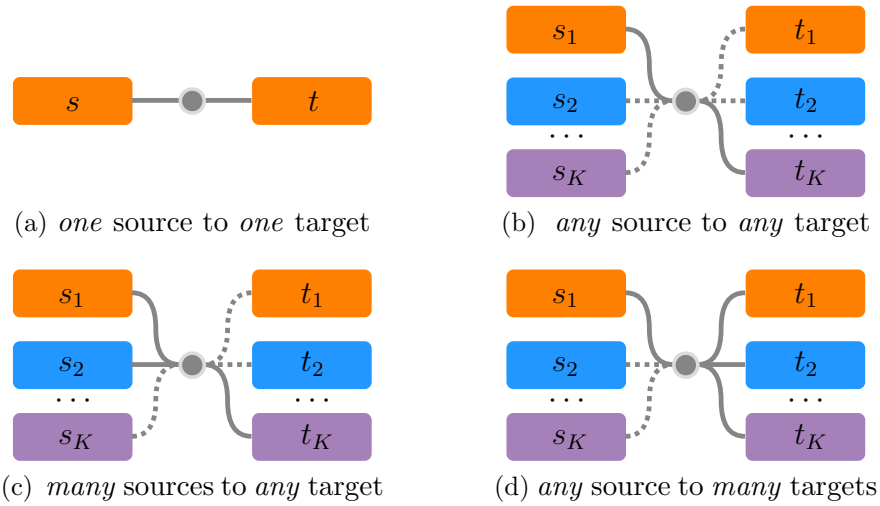
Prototype learners. Learning metric spaces with prototypes for image retrieval [Deng et al., 2019, Liu et al., 2017b, Movshovitz-Attias et al., 2017, Snell et al., 2017, Sohn, 2016, Wang et al., 2018, Wen et al., 2016, Zhai and Wu, 2019] and classification [Chintala et al., 2017, Mensink et al., 2013, Mettes et al., 2019, Snell et al., 2017] provides a simpler alternative to common contrastive [Chopra et al., 2005, Hadsell et al., 2006] or triplet [Schroff et al., 2015, Weinberger and Saul, 2009] loss functions. One line of work learns to regress to moving prototypical representations. Depending on the task, such prototypes can correspond to center [Wen et al., 2016], proxy [Movshovitz-Attias et al., 2017, Zhai and Wu, 2019], or support [Ren et al., 2018, Snell et al., 2017] representations. While the distance measure usually relies on a cosine or Euclidean distance, a margin has also been introduced in the distance measure [Deng et al., 2019, Liu et al., 2017b, Wang et al., 2018]. Another line of work regresses to fixed prototypical representations to avoid the simultaneous learning of prototypes and model parameters. Examples of fixed representations include class means [Mensink et al., 2013], one-hot representations [Chintala et al., 2017], or separated representations [Mettes et al., 2019]. We build on the latter approach for open cross-domain visual search. We formulate semantic prototypes to align examples from many domains simultaneously. Categories are represented by fixed semantic prototypes in the embedding space. We then define a prototype learner for every domain to map visual inputs to the common space where open cross-domain search occurs.

2.3 Method

2.3.1 Problem formulation

Figure 2.2 illustrates the search scenarios for open cross-domain search. While the *closed* cross-domain setting focuses on one pre-defined source s and one pre-defined target t, the *open* cross-domain setting searches for categories from any source domain s_k to any target domain t_k. As multiple domains now become available, this opens the door for combining multiple domains at both source and target positions. Thus, the main difference between the *closed* setting and the *open* setting lies in the ability to leverage multiple domains for categorical cross-domain visual search.

Formally, let $\mathcal{D}$ denote the set of all domains to be considered. Rather than making an explicit split of a dataset into source and target, we consider a large combined visual collection $\mathcal{T} = \{(\mathbf{x}_n^d, y_n)\}_{n=1}^N$, where $\mathbf{x}_n^d \in \mathcal{I}_d$ denotes an input



(a) *one* source to *one* target (b) *any* source to *any* target

(c) *many* sources to *any* target (d) *any* source to *many* targets

Figure 2.2: **Open cross-domain visual search configurations.** Cross-domain image search focuses on mapping (a) from one fixed source to one fixed target domain. In this chapter, we consider an open domain setting with K available domains. We search (b) from any source to any target domain, (c) from multiple source domains to any target domain, and (d) from any source domain to multiple target domains.

example from a visual domain $d \in \mathcal{D}$ of category $y_n \in \mathcal{Y}$. In other words, $\mathcal{Y}$ is common and shared among all domains $\mathcal{D}$ but is depicted differently from domain d_i to domain d_j, with $i \neq j$.

Categorical search consists in using a sample query $\mathbf{x}^{d_i}$ from domain d_i to retrieve samples of the same category y in the gallery of domain d_j. If $i \neq j$, this corresponds to a cross-domain categorical search as the search occurs across two different domains. A *closed* setting only considers $|\mathcal{D}| = 2$, *i.e.* with a pre-defined source domain and a pre-defined target domain. We define the *open* setting as comprising $|\mathcal{D}| > 2$. This stimulates novel search configurations. For example, we may want to combine two queries $(\mathbf{x}^{d_i}, \mathbf{x}^{d_j})$ of two different domains $i \neq j$ to search in the gallery of a third domain k. Conversely, given a sample query $\mathbf{x}^{d_i}$, we can search in the combined gallery of multiple domains.

2.3.2 Proposed approach

We pose open domain visual search as projecting any number of heterogeneous domains to prototypes on a common and shared hyperspherical semantic space. First, we outline how to represent categories in the semantic embedding space. Second, we propose a mapping function for every domain to the common semantic embedding space. Third, we outline how open cross-domain search occurs.

Categorical prototypes. We leverage the concept of prototypes to represent categories in a common semantic space. Every category is represented by a unique real-valued vector, corresponding to a categorical prototype. Hence, the objective is to align examples, coming from different domains but with the same category label, to the same categorical prototype in the semantic space. For every category $y \in \mathcal{Y}$, we denote its prototype on the semantic space as $\phi(y) \in \mathbb{S}^{D-1}$ for a D-dimensional hypersphere. Relying on semantic relations enables to search for unseen classes using models trained on seen categories [Frome et al., 2013, Palatucci et al., 2009]. In this work, we opt for word embeddings, *e.g.*, word2vec [Mikolov et al., 2013] or GloVe [Pennington et al., 2014], to represent categories, as these embeddings adhere to the semantic relation property.

Mapping domains to categories. For every domain $d \in \mathcal{D}$, we learn a separate mapping function $f_d \in \mathbb{S}^{D-1}$ to the common and shared semantic space. Separate mapping functions are not only easy to train, they also enable us to incorporate new domains over time. Indeed, we only have to train the mapping of the new incoming domain without retraining previous mapping functions of existing domains. The mapping function is formulated as a convolutional network followed by an ℓ_2-normalization on the D-dimensional network outputs.

We propose the following function to map an example $\mathbf{x}^d$ of domain d to its categorical prototype $\phi(y)$ in the common semantic space:

$$p(y|\mathbf{x}^d, d) = \frac{\exp\Big(-s \cdot c\big(f_d(\mathbf{x}^d), \phi(y)\big)\Big)}{\sum_{y' \in \mathcal{Y}} \exp\Big(-s \cdot c\big(f_d(\mathbf{x}^d), \phi(y')\big)\Big)}, \tag{2.1}$$

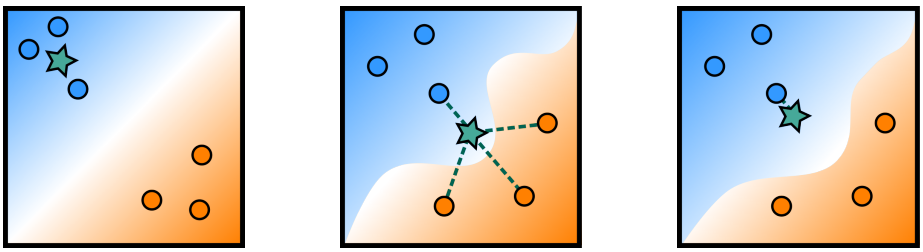
where $s \in \mathbb{R}_{>0}$ denotes a scaling factor, inversely equivalent to the temperature [Hinton et al., 2014]. Intuitively, the scaling controls how samples are spread around categorical prototypes. $c(\cdot, \cdot)$ is defined as the cosine distance:

$$c(f_d(\mathbf{x}^d), \phi(y)) = 1 - < f_d(\mathbf{x}^d), \phi(y) >, \tag{2.2}$$

where $<\cdot, \cdot>$ is the dot product. As both $f_d(\mathbf{x})$ and $\phi(y)$ lie on the hypersphere $\mathbb{S}^{D-1}$, they have a unit norm. Finally, learning every mapping function f_d is done by minimizing the cross-entropy loss over the training set:

$$\mathcal{L} = -\frac{1}{N} \sum_{n=1}^{N} \log p(y_n|\mathbf{x}_n^d, d). \tag{2.3}$$

In our approach, the representations of the categorical prototypes remain unaltered. Hence, we only take the partial derivative with respect to the mapping function parameters. When training the mapping function f_d for domain d, only examples $\mathbf{x}^d$ of domain d are used as inputs.



(a) Ideal. (b) Real. (c) Refined.

Figure 2.3: **Cross-domain query refinement**. (a) Ideally, the neighborhood of the query (star) is only close to examples from the same category. (b) In reality, variability causes noise in the semantic space. Hence, the query might also be close to samples from other categories. (c) We tackle this variability by refining the query representation.

Searching across open domains. In the search evaluation phase, similarity between source and target samples is measured with the cosine distance in the shared semantic space. Given one or more queries from different source domains, we first project all queries to the shared semantic space and average their positions into a single vector. Then, we compute the distance to all target examples to rank them with respect to the source query. As all domains map to the same common semantic space, domains can straightforwardly be combined either to search with queries from multiple domains or to search within a gallery of multiple domains.

2.3.3 Refining queries across domains

With our approach, a source query is close to target examples from the same category, regardless of the domains of the query and target examples. In practice, inherent variability in the hyperspherical semantic space can cause noise in the similarity measures. We then propose to refine the initial query representation using a nearby example from the target domain, as illustrated in Figure 2.3.

We refine the query representation p_0 by performing a spherical linear interpolation with a relevant representation p_1. The refined representation $\hat{p}$ is:

$$\hat{p}(p_0, p_1|\lambda) = \frac{\sin\big((1-\lambda)\Omega\big)}{\sin\Omega} p_0 + \frac{\sin\big(\lambda\Omega\big)}{\sin\Omega} p_1, \tag{2.4}$$

where $\Omega = \arccos(p_0 \cdot p_1)$ and $\lambda \in [0, 1]$ controls the amount of mixture in the refinement process. The higher the value of lambda is, the further away the refined representation is from the original representation p_0. Intuitively, the refinement performs a weighted signal averaging to reduce the noise present in the initial representation. In retrieval, we set p_1 as the 1-nearest neighbour of p_0 in the target set. This mixture doesn't require any label and relies on the fact that the

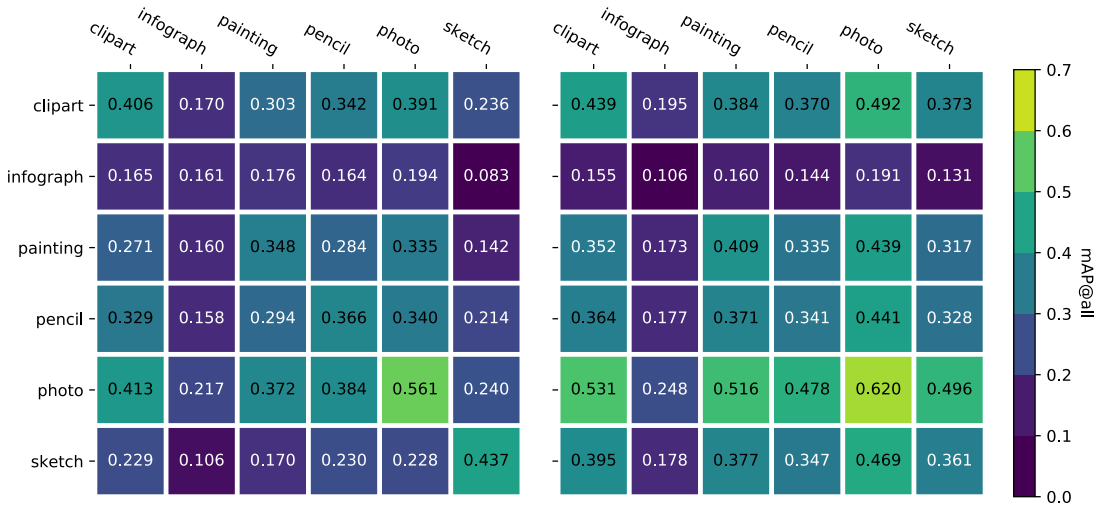
recall at one is usually very high. In classification, p_1 is the word embedding of the category name.

2.4 Open cross-domain visual search

In the first set of experiments, we demonstrate the ability to perform open cross-domain visual search in three ways. We note that this is a new setting, making direct comparisons to existing works infeasible. First, we demonstrate how we can search from any source to any target domain without hassle. Second, we show the potential and positive effect of searching from multiple source domains for any target domain. Third, we exhibit the possibility of searching in multiple target domains simultaneously.

Setup. We evaluate on the recently introduced *DomainNet* [Peng et al., 2019], which contains 596,006 images from 345 classes. Images are gathered from six visual domains: *clipart*, *infograph*, *painting*, *pencil*, *photo* and *sketch*. We consider retrieval in *zero-* and *many*-shot evaluations: (i) in the *zero*-shot evaluation, $\mathcal{Y}$ is split into $\mathcal{Y}_{train}$ and $\mathcal{Y}_{test}$, with $\mathcal{Y}_{train} \cap \mathcal{Y}_{test} = \emptyset$, *i.e.*, categories to be searched during inference have not been seen during training; (ii) the *many*-shot evaluation uses the same categories during both training and testing. The zero-shot evaluation randomly splits samples into 300 training and 45 testing classes. Following the zero-shot learning good practices in Xian et al. [2018a], we have verified the presence of the 345 categories of *DomainNet* [Peng et al., 2019] in *ImageNet* [Russakovsky et al., 2015], where we identify 188 separate categories. From this list of separate categories, we randomly sample 45 zero-shot categories with at least 40 samples per class in every domain. The many-shot evaluation follows the original splits from Peng et al. [2019]. We report the mean average precision (mAP@all).

Implementation details. Throughout the paper and unless stated otherwise, we use SE-ResNet50 [Hu et al., 2018b] pre-trained on ImageNet [Russakovsky et al., 2015] as a backbone, and word2vec trained on a Google News corpus [Mikolov et al., 2013] as the common semantic space. We remove the final classifier layer of SE-ResNet50, and replace it with a fully-connected layer of size $D = 300$ initialized with random weights. The new layer is followed by a linear activation and batch normalization [Ioffe and Szegedy, 2015]. We optimize the loss in Equation 2.3 with Nesterov momentum [Sutskever et al., 2013] by setting the coefficient to 0.9. We apply a learning rate of 1e$-$4 with cosine annealing without warm restarts [Loshchilov and Hutter, 2017] and a batch size of 128. We use a scaling factor s of 20, and decrease it to 10 for Sections 2.5.2 and 2.5.3. We set $\lambda = 0.7$ when evaluating on unseen classes (*i.e.*, zero-shot and few-shot evaluations) and to 0.4 when evaluating on seen classes (*i.e.*, many-shot evaluation). The implementation rests on the Pytorch [Paszke et al., 2019] framework



(a) *Zero*-shot evaluation (45 unseen classes), (b) *Many*-shot evaluation (all 345 classes).

Figure 2.4: **Demonstration 1** for visual search from any source (columns) to any target (rows) domain in mAP@all. Our approach can perform 36 cross-domain searches for both (a) *zero*-shot evaluation, and (b) *many*-shot evaluation, without any modifications as we bypass the need to align domains.

and image similarities are computed with the Faiss [Johnson et al., 2017] library. Word embeddings of class names are extracted with the Gensim [Řehůřek and Sojka, 2010] library.

2.4.1 From any source to any target domain

First, we demonstrate how searching from any source to any target domain in an open setting is trivially enabled by our approach. Figure 2.4 shows the result of 72 cross-domain search evaluations; corresponding to all six cross-domain pairs for both zero- and many-shot evaluations. In our formulation, such an exhaustive evaluation is enabled by training only six models, one for every domain. For comparison, a domain adaptation approach –the standard in current cross-domain search methods– requires a pair-wise training of all available domain combinations. Moreover, our formulation allows for an easy integration of new domains, as only the mapping from a new visual domain to the shared semantic space needs to be trained. While approaches based on pair-wise training scale with a quadratic complexity to the number of domains, we scale linearly.

In the zero-shot evaluation with an evaluation on the unseen classes (Figure 2.4a), the *photograph* domain provides the most effective search whether used as source or target. One reason is the number of available images, which is up to four times larger than other domains. On the other hand, *infographs* and *sketches* are very diverse in terms of scale and visual representations, which induces a much

target domain	***zero*-shot**		***many*-shot**	
	SAKE	*This work*	SAKE	*This work*
clipart	0.199	**0.236**	0.268	**0.373**
infograph	0.080	**0.083**	0.097	**0.131**
painting	0.118	**0.142**	0.203	**0.317**
pencil	0.181	**0.214**	0.230	**0.328**
photo	0.206	**0.240**	0.358	**0.496**

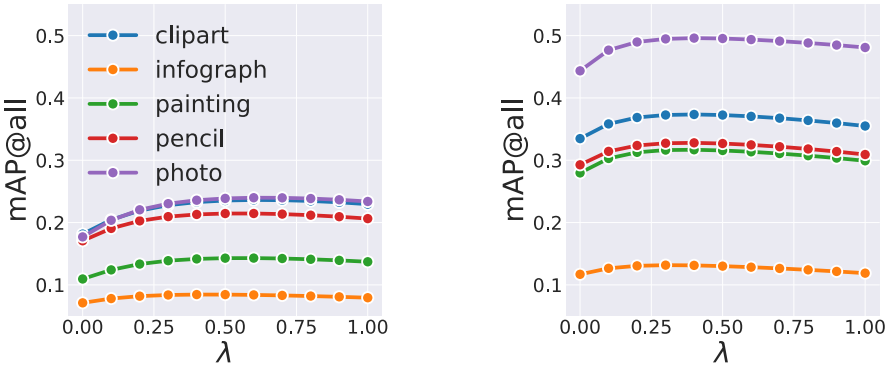
Table 2.1: **Visual search from sketches** as a source to any target domain comparison with SAKE [Liu et al., 2019] in mAP@all. Our formulation achieves competitive results in both *zero*- and *many*-shot evaluations.

more difficult search.

In the many-shot evaluation with an evaluation on all classes (Figure 2.4b), the *photograph* domain exhibits a similar behaviour. Though, in this case the search performance for *sketches* is at the same level as other considered domains, such as *clipart*, *painting* or *pencil*. Seeing all classes helps the prototype learner to better grasp the variability in *sketches*. The *infograph* domain remains the most challenging. We conclude from the first demonstration that search from any source to any target domain is not only feasible with our approach, it can be done easily for both zero- and many-shot evaluations since we bypass the need to align different domains.

We quantitatively compare with the state-of-the-art SAKE [Liu et al., 2019] on zero-shot sketch-based image retrieval. We run SAKE from the original source code provided by the authors. Table 2.1 presents the results when considering sketches as the source domain and retrieving images in any of the other domains. SAKE has been proposed with a zero-shot evaluation design from the start, which makes it strong in this setting. Indeed, results are close, we only observe an improvement of 0.3% (*infograph*) up to 3.7% (*clipart*). When the evaluation focuses on a large number of categories, we notice higher gains from 3.4% (*infograph*) up to 13.8% (*photograph*) in the many-shot evaluation. Our embedding space is better partitioned for all categories thanks to the semantic prototypes. Overall, our formulation provides competitive performance in both zero- and many-shot evaluations with a simpler training.

Finally, we also assess the importance of the proposed refinement module of Equation 2.4. Figure 2.5 illustrates the effect of our cross-domain prototypical refinement when searching in any target domain from the *sketch* domain. We create a mixture between the *sketch* query ($\lambda = 0$) and its nearest neighbour in the gallery ($\lambda = 1$) for retrieval. For both zero- and many-shot evaluations, refining the representations improves the performance. We observe a need for a lower mixture for the many-shot evaluation, as classes are all seen during training compared to the zero-shot evaluation. Refining the representations helps to bridge



(a) *Zero*-shot evaluation (b) *Many*-shot evaluation

Figure 2.5: **Ablation on cross-domain query refinement** on DomainNet, with *sketches* as a source. Refining the source representation always improves the retrieval performance.

the inherent cross-domain gap.

2.4.2 From multiple sources to any target domain

Second, we demonstrate the potential to search from multiple source domains. Due to the generic nature of our approach, we are not restricted to search from a single source. We show that a multi-source search benefits the search in any target domain.

For this experiment, we start from the *sketch* domain as a source and investigate the effect of including queries from the most effective source (*photographs*) and the least effective source (*infographs*). Table 2.2a highlights the positive effect of searching with an additional domain, rather than a single source domain. When using multiple sources, we simply average the positions in the common semantic space. For fairness, we also evaluate search using two *sketches*. Across all settings, we find that searching from multiple queries improves relative to using one single *sketch* query. In the zero-shot evaluation, including *infographs* and *photographs* improves upon sketch-based search only. In the many-shot evaluation, including *infographs* improves upon search by one *sketch*, but not by two *sketches*, which is not surprising given the low scores for infographs individually. *Photographs* with *sketches* obtain the highest scores, regardless of the target domain or the evaluation setting.

We also consider a more challenging multi-source search scenario where we search from the most informative source (*photograph*) and one of the least informative sources (*infograph* or *sketch*). Table 2.2b confirms the positive effect of searching with an additional domain. Adding *infographs* only improves the results marginally. Performance can even decrease when searching within one of the

target domain	***zero*-shot**			***many*-shot**		
	sk+sk	sk+in	sk+ph	sk+sk	sk+in	sk+ph
clipart	+.057	+.072	**+.211**	+.097	+.036	**+.178**
infograph	+.018	+.067	**+.107**	+.031	+.002	**+.075**
painting	+.035	+.080	**+.186**	+.079	+.029	**+.154**
pencil	+.054	+.060	**+.154**	+.083	+.043	**+.156**
photo	+.064	+.112	**+.328**	+.127	+.049	**+.185**

(a) Improving the less informative sketch representations

target domain	***zero*-shot**			***many*-shot**		
	ph+ph	ph+in	ph+sk	ph+ph	ph+in	ph+sk
clipart	**+.070**	+.012	+.048	**+.075**	+.002	+.067
infograph	**+.029**	-.035	+.005	**+.027**	-.062	+.018
painting	**+.052**	+.011	+.008	**+.061**	+.004	+.049
pencil	**+.054**	+.012	+.037	**+.066**	+.000	+.057
sketch	+.041	+.001	**+.202**	**+.075**	-.013	-.030

(b) Improving the more informative photograph representations

Table 2.2: **Demonstration 2** for visual search from multiple sources to any target domain (absolute improvement in mAP@all). In our approach, searching from multiple sources is as easy as using a single source, as we only have to average their positions in the common semantic space. Searching (a) from multiple diverse domains is preferred when the source is less informative, while (b) more examples from the same domain are preferred when the source is more informative.

least informative domains, because the combination creates a destructive noise that moves the initial representation to a wrong direction. Adding *sketches* can benefit searching within *sketches* when the uncertainty is high, as in a zero-shot evaluation, but slightly decreases the score when the uncertainty is low, as in a many-shot evaluation. In the other target domains, *sketches* are much more effective than *infographs* when added to *photographs*. Though, the improvement is lower than searching from two *photographs*. When searching from an informative source domain, combining it with itself improves more than a combination with a less informative domain for both zero- and many-shot evaluations.

This demonstration shows the potential of searching from multiple sources. It is better to diversify the search by using multiple diverse domains when the source is less informative while more queries from the same domain are preferred when the source is more informative. Similar to the first demonstration, this evaluation is a trivial extension to our approach, as we only have to average positions in the shared semantic space, regardless of the domain the examples come from.

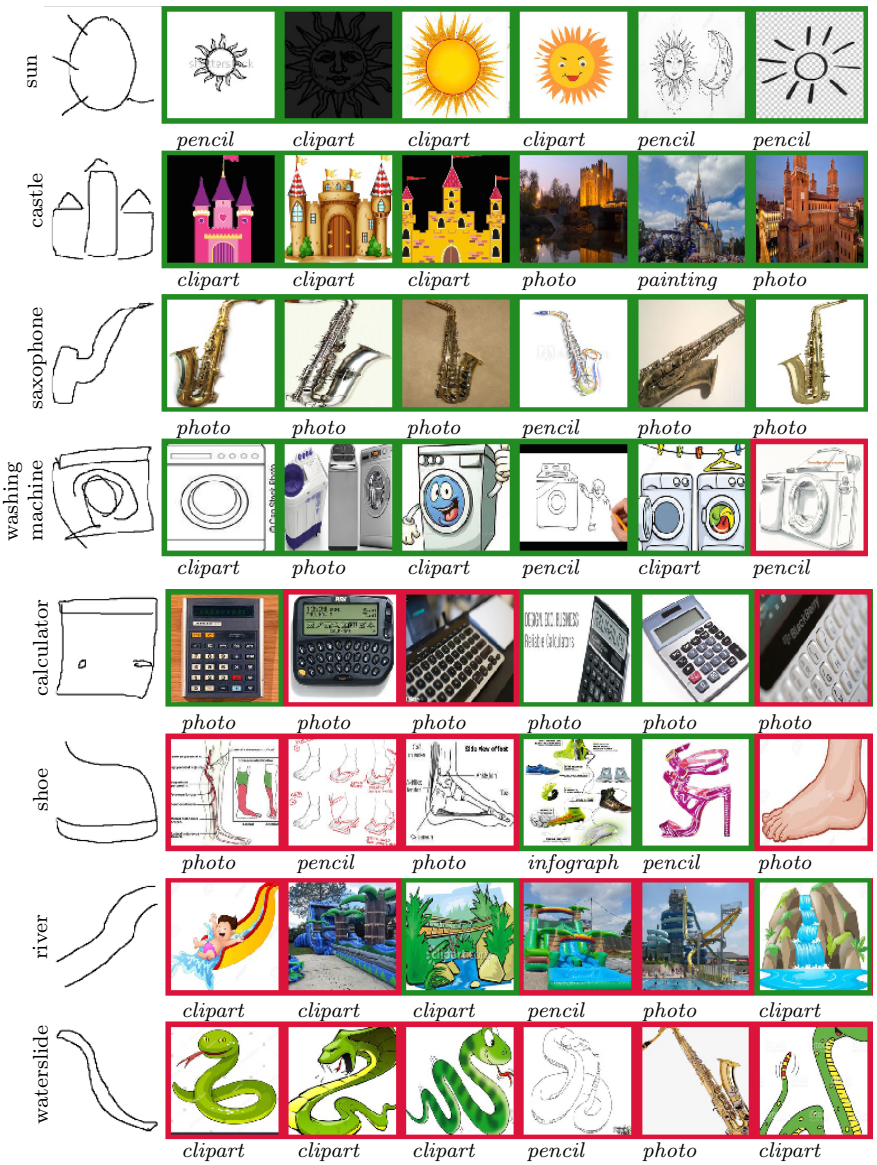


Figure 2.6: **Demonstration 3** for visual search from any source to multiple target domains. Correct results are in *green*, incorrect in *red*. For abstract categories such as "sun", abstract domains such as *clipart* or *pencil* drawings tend to be retrieved first. When *sketches* are more ambiguous such as "shoe", some retrieved results are incorrect but resemble the shape.

2.4.3 From any source to multiple target domains

Third, we demonstrate our ability to search in multiple domains simultaneously. This setting has potential applications for example in untargeted portfolio browsing, where a user may want to explore all possible visual expressions of a category. Exploring in multiple domains also highlights whether certain categories have a

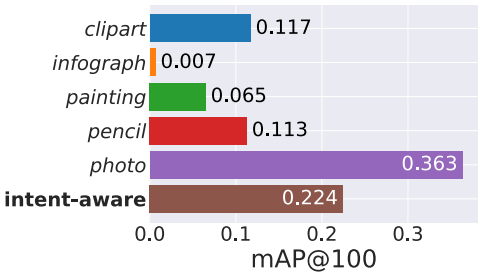


Figure 2.7: **Intent-aware evaluation** for visual search from sketches to the other five target domains. Correct retrieved images in the top-ranked results more likely come from the *photograph* than the *infograph* domain.

preference towards specific domains, which offers an insight on how to best depict those categories. Note that this setting can also be easily extended to include also multiple domains as a source. For the sake of clarity, we use *sketch* as the source domain and search in the other five domains in a many-shot evaluation.

Figure 2.6 provides qualitative results for eight *sketches* from different categories. We first observe that the results come from multiple target domains, without being explicitly told to do so. We do not need to align results from different target domains, since we measure distance in the common semantic space. For categories such as "sun", we have a bias towards retrieving abstract depictions, such as *pencil* drawings and *cliparts*, as the "sun" is a category with a clear abstract representation. "Castle" on the other hand has a bias towards both distinct *cliparts*, as well as *photographs* and *paintings*. In both cases, all top results are relevant. For categories with more ambiguous *sketches*, such as "river" or "calculator", retrieved examples resemble the shape of the provided *sketch*, but do not match the category. Overall, we conclude that searching in multiple domains is not only trivial in our approach, but is also an indicator of the presence of preferential domains for depicting categories.

We also quantitatively measure the retrieval performance when searching from sketches to the other five target domains simultaneously. When computing the mAP@100, we obtain a score of 0.565. Though, this measure does not take into account the differences and diversity among domains, as it considers all of them as similar. As such, we report the intent-aware mAP [Agrawal et al., 2009]. Extending the mAP to an intent-aware formulation provides an estimate of the result diversity by: (i) computing the mAP per domain, and (ii) summing them with a weighting that corresponds to the occurrences of every category within each domain. Figure 2.7 shows the per domain and intent-aware mAP@100. The *photograph*-mAP@100 is the highest score, which indicates correct *photographs* are in the top-ranked results compared with other target domains. The *infograph*-mAP@100 obtains the lowest score, which means that there are very few correct *infographs* in the top-ranked results. When the differences among domains are taken into consideration, the intent-aware mAP@100 results in 0.224. In a search within multiple domains, the informativeness of each domain influences the top-ranked results.

2.5 Closed cross-domain visual search

2.5.1 Zero-shot sketch-based image retrieval

Setup. Zero-shot sketch-based image retrieval focuses on retrieving natural images (target domain) from a sketch query (source domain). We evaluate on two datasets. *TU-Berlin Extended* [Eitz et al., 2012, Zhang et al., 2016] contains 20,000 sketches and 204,070 images from 250 classes. Following Shen et al. [2018], we select 220 classes for training and 30 classes for testing. *Sketchy Extended* [Liu et al., 2017a, Sangkloy et al., 2016] contains 75,481 sketches and 73,002 images from 125 classes. Similarly, following Shen et al. [2018], we select 100 classes for training and 25 classes for testing. For fair comparison with Liu et al. [2019], we select the same unseen classes for both datasets. Following recent works [Dutta and Akata, 2019, Liu et al., 2019, Shen et al., 2018], we report the mAP@all and the precision at 100 (prec@100) scores.

Our approach is geared towards open cross-domain visual search, as demonstrated in the previous section. To get insight in the effectiveness of our approach for cross-domain visual search in general, we also perform an extensive comparative evaluation on standard cross-domain settings, which search between two domains. In total, we compare on three of the most popular cross-domain search tasks, namely zero-shot sketch-based image retrieval [Sangkloy et al., 2016, Shen et al., 2018], few-shot sketch-based image classification [Hu et al., 2018a], and many-shot sketch-based 3D shape retrieval [Li et al., 2013, 2014b]. For our approach, we simply train one mapping function for the source domain, and one for the target domain using the examples provided during training. Below, we present each comparison separately.

Results. Table 2.3a compares to six state-of-the-art baselines on both datasets. Baselines mostly focus on bridging the domain gap between sketches and natural images with domain adaptation losses [Ganin et al., 2016, Gonzalez-Garcia et al., 2018]. On Sketchy Extended, our approach outperforms other baselines. On TU-Berlin Extended, we obtain the highest mAP@all score, while the recently introduced SAKE by Liu et al. [2019] obtains a higher prec@100 score. SAKE is better at grouping images from the same category together thanks to the preservation module that produces tightly distributed representations. Our method is better at retrieving relevant images in the first ranks as the refinement module reduces the noise in the query representations.

Following previous work in zero-shot sketch-based image retrieval [Dutta and Akata, 2019, Liu et al., 2019, Lu et al., 2018, Shen et al., 2018], we also report the retrieval performance on binary representations. As previously proposed in [Dutta and Akata, 2019, Liu et al., 2019], real-valued representations are projected to a low-dimensional space and quantized with iterative quantization [Gong et al., 2012]. We compute the transformation on the training set and apply it on both sketch and image testing sets. Note that we first refine the representations,

Method	**TU-Berlin Extended**		**Sketchy Extended**	
	mAP@all	prec@100	mAP@all	prec@100
EMS [Lu et al., 2018]	0.259	0.369	n/a	n/a
CAAE [Yelamarthi et al., 2018]	n/a	n/a	0.196	0.284
ADS [Dey et al., 2019]	0.110	n/a	0.369	n/a
SEM-PCYC [Dutta and Akata, 2019]	0.297	0.426	0.349	0.463
SG [Dutta and Biswas, 2019]	0.254	0.355	0.376	0.484
SAKE [Liu et al., 2019]	0.475	**0.599**	0.547	0.692
This work	**0.517**	0.557	**0.649**	**0.708**

(a) Real-valued representations

Method	**TU-Berlin Extended**		**Sketchy Extended**	
	mAP@all	prec@100	mAP@all	prec@100
EMS [Lu et al., 2018]	0.165	0.252	n/a	n/a
ZSIH [Shen et al., 2018]	0.220	0.291	0.254	0.340
SEM-PCYC [Dutta and Akata, 2019]	0.293	0.392	0.344	0.399
SAKE [Liu et al., 2019]	0.359	0.481	0.364	0.487
This work	**0.404**	**0.517**	**0.466**	**0.618**

(b) Binary representations

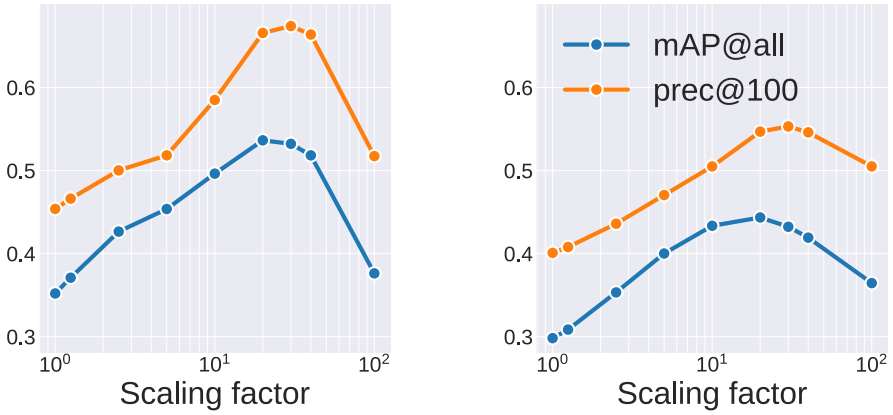
Method	**TU-Berlin Extended**		**Sketchy Extended**	
	mAP@all	prec@100	mAP@all	prec@100
ZSIH [Shen et al., 2018]	0.142	0.218	0.219	0.296
SEM-PCYC [Dutta and Akata, 2019]	0.192	**0.298**	0.307	0.364
SG [Dutta and Biswas, 2019]	0.149	0.226	0.331	0.381
This work	**0.211**	0.224	**0.397**	**0.421**

(c) Generalized setting

Table 2.3: **Comparison 1** to zero-shot sketch-based image retrieval on TU-Berlin Extended and Sketchy Extended. Aligning solely the semantics improves cross-domain image retrieval.

then apply iterative quantization. Table 2.3b compares the proposed formulation with binary representations of 64 dimensions. Compared to real-valued representations in Table 2.3a, we notice a higher drop in the mAP@all score compared to prec@100 score. Compared to other baselines, our semantic space based on word embeddings better preserves the information when compressed to a low-dimensional space.

As recently introduced by Dutta and Akata [2019], we also evaluate on a generalized setting in Table 2.3c, where the gallery set also includes images from seen classes. Following their protocol, we reserve 20% of the samples from the seen



(a) Sketchy Extended (b) TU-Berlin Extended

Figure 2.8: **Scaling hyper-parameter** ablation. We evaluate the scaling of the softmax function. $s = 20$ yields the best results for both datasets, especially for the mAP@all score.

classes for evaluation and use VGG16 [Simonyan and Zisserman, 2014] in this experiment for fair comparison. On Sketchy Extended, our approach also outperforms other baselines. On TU-Berlin Extended, we obtain the highest mAP@all score, while SEM-PCYC by Dutta and Akata [2019] obtains a higher prec@100 score. Similar to the zero-shot evaluation, our method is better at ranking images than grouping them together. Overall, focusing solely on semantic alignment outperforms alternatives on domain adaption or knowledge preservation across three different settings derived from two datasets.

To understand the effect of the distance scaling hyper-parameter defined in Equation 2.1, we vary its value on both datasets in Figure 2.8. We observe the same behaviour on both datasets. When $s = 1$ as in a common softmax function, it yields the lowest results. A higher scaling helps to narrow the probability distribution, resulting in a better retrieval performance. There is a tipping point around $s = 20$, after which performance decreases. Calibrating the softmax with a high distance scaling factor improves the retrieval performance.

Qualitative analysis. To understand which sketches trigger the performance of natural image retrieval, we provide several qualitative sketch queries with their top retrieved images in Figure 2.9. Our approach works well for typical sketches of categories. For example, the "cup" or "parrot" sketches exhibit a typical definition of their respective categories. In return, the search is very effective despite the variation in image appearance and viewpoints. Results degrade when sketches are ambiguous or in non-canonical views. For example, the "tree" sketch can easily be confused with the smoke ring of a "volcano" or the shape of a "windmill". Typical shape drawings of sketches matter for zero-shot image retrieval.

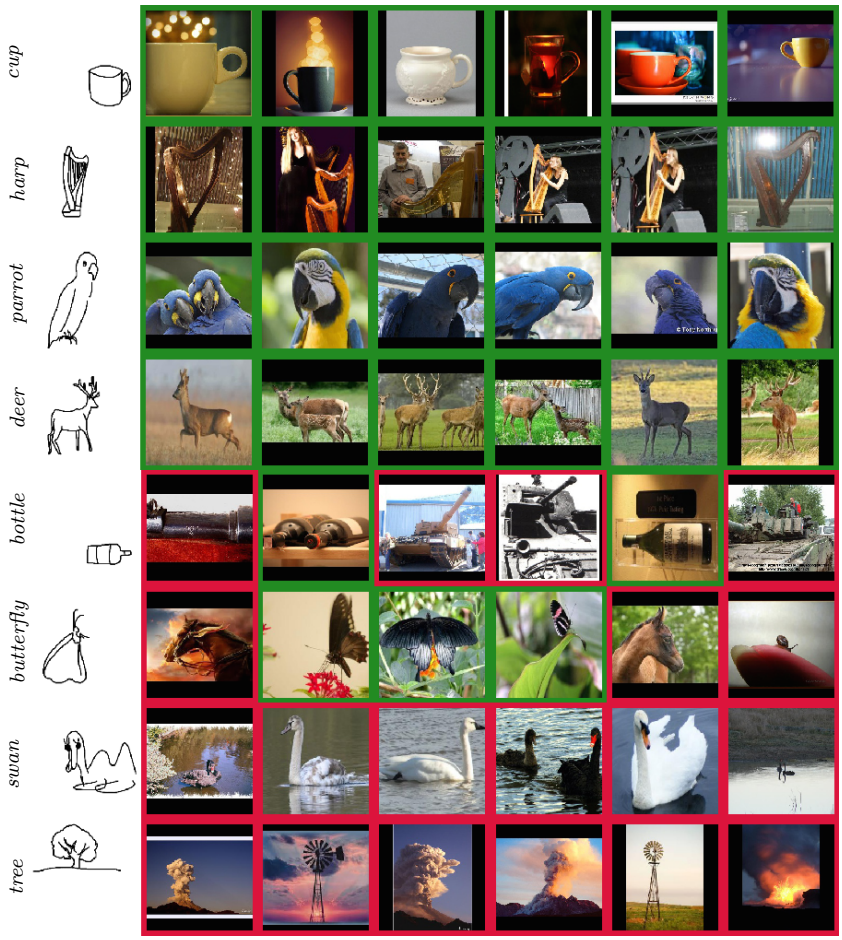


Figure 2.9: **Qualitative analysis** of zero-shot sketch-based image retrieval. We show eight sketches of Sketchy Extended, with correct retrievals in *green*, incorrect in *red*. For typical sketches (*e.g.*, "cup"), the closest images are from the same category. For ambiguous sketches (*e.g.*, "tree") or non-canonical views (*e.g.*, "butterfly"), our approach struggles.

2.5.2 Few-shot sketch-based image classification

Setup. Few-shot sketch-based image classification focuses on classifying natural images from one or a few labeled sketches. The few-shot categories have not been observed during training. Different from the zero-shot retrieval scenario, the few-shot classification evaluation has access to the labels of the unseen classes in the evaluation phase. For example, this comes through the form of sketches or word embeddings. We report results on the *Sketchy Extended* dataset [Liu et al., 2017a, Sangkloy et al., 2016]. For fair comparison with Hu et al. [2018a], we subsample the Sketchy Extended to match the size of their private split. We select the same 115 classes for training and 10 classes for testing. We also rely on VGG19 [Simonyan and Zisserman, 2014] as a backbone. We evaluate the

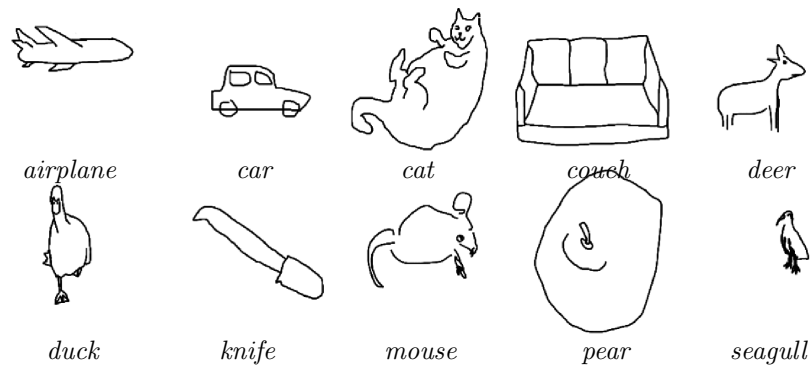
Method	**w2v**	**sketch**		**image**	
		one-shot	*five*-shot	*one*-shot	*five*-shot
M2M [Hu et al., 2018a]	n/a	n/a	79.93	n/a	93.55
F2M [Hu et al., 2018a]	35.90	68.16	83.01	84.12	93.89
This work	**80.39**	**82.19**	**85.13**	**90.63**	**94.63**

Table 2.4: **Comparison 2** to few-shot sketch-based image classification on a subsampled Sketchy Extended (multi-class accuracy). Our metric learning approach outperforms model regression approaches.

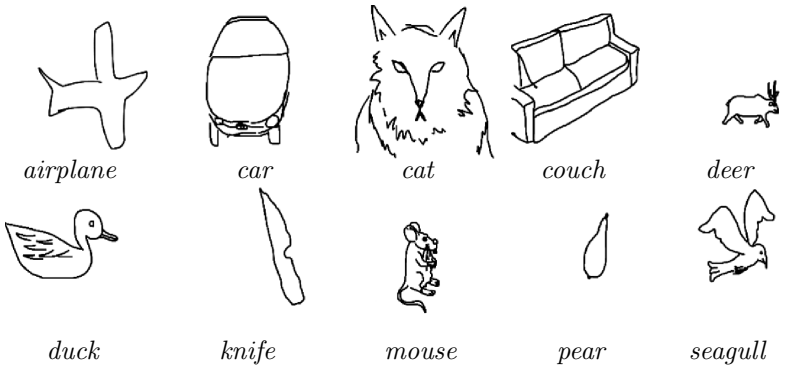
performance with the multi-class accuracy. Classification is done by measuring the distance to the class prototypes. Following Hu et al. [2018a], we evaluate on three different modes by setting the prototypes of the unseen classes to: (i) word vectors (w2v), (ii) *one* or *five* sketch representations, and (iii) *one* or *five* image representations. The latter is considered as an upper-bound of this cross-domain task. Following Hu et al. [2018a], the model is trained once and we report the average classification accuracy over 500 runs with different sets of sketches or images in the few-shot evaluation.

Results. Table 2.4 compares our formulation to two baselines introduced by Hu et al. [2018a]: M2M regresses weights for natural image classification from the weights of the sketch classifier while F2M regresses weights from sketch representations. For the first evaluation mode, we obtain an accuracy of 76.73%, compared to 35.90%, which reiterates the importance of a semantic alignment for categorical cross-domain search. In the few-shot evaluation, the biggest relative improvement is achieved in the one-shot evaluation. It is also interesting to compare the w2v and one-shot sketch evaluation modes. As the one-shot sketch exhibits a higher score, it means that sketch representations capture visual details that cannot be described with word representations only. Our approach is also effective for cross-domain classification, especially with low shots.

Qualitative analysis. To understand how to best employ our approach for few-shot sketch-based image classification, we provide the most and least effective sketches for image classification in Figure 2.10. Since categories are condensed to a single prototypical sketch, our approach desires sketches with details and in canonical configurations. Results are degraded when such assertions are not met. For example, Figure 2.10a shows a well sketched "cat" in one of the canonical positions while Figure 2.10b exhibits a "cat" without any whiskers and in a strange view as we only see the face. Another important assertions is the sketch separability. For example, the "airplane" sketch in Figure 2.10b could be confused with a "knife". Appearance, viewpoint and separability matter when relying on sketches for few-shot image classification.



(a) Most effective set of sketches (86.07% accuracy),



(b) Least effective set of sketches (43.82% accuracy).

Figure 2.10: **Qualitative analysis** of few-shot sketch-based image classification on a subsampled Sketchy Extended. (a) Since our approach condenses examples of category to a single prototype in the shared space, we obtain high scores when source sketches are detailed and in canonical views (*e.g.*, "deer" or "couch"). (b) The accuracy decreases when sketches are drawn badly (*e.g.*, "airplane"), or in non-canonical views (*e.g.*, "car" or "cat").

2.5.3 Many-shot sketch-based 3D shape retrieval

Setup. Sketch-based 3D shape retrieval focuses on retrieving 3D shape models from a sketch query, where both training and testing samples share the same set of classes. We evaluate on three datasets. *SHREC13* [Li et al., 2013] is constructed from the TU-Berlin [Eitz et al., 2012] and Princeton Shape Benchmark [Shilane et al., 2004] datasets, resulting in 7,200 sketches and 1,258 3D shapes from 90 classes. The training set contains 50 sketches per class, the testing set 30. *SHREC14* [Li et al., 2014b] contains more 3D shapes and more classes, resulting in 13,680 sketches and 8,987 3D shapes from 171 classes. The training and testing splits of sketches follow the same protocol as SHREC13. We also report on *Part-SHREC14* [Qi et al., 2018], which contains 3,840 sketches and 7,238 3D shapes from 48 classes. The sketch splits also follow the same protocol, while

Method	NN	FT	ST	E	DCG	mAP
Siamese [Wang et al., 2015]	0.405	0.403	0.548	0.287	0.607	0.469
Shape2Vec [Tasse and Dodgson, 2016]	0.620	0.628	0.684	0.354	0.741	0.650
DCML [Dai et al., 2017]	0.650	0.634	0.719	0.348	0.766	0.674
LWBR [Xie et al., 2017]	0.712	0.725	0.785	0.369	0.814	0.752
DCA [Chen and Fang, 2018]	0.783	0.796	0.829	0.376	0.856	0.813
SEM [Qi et al., 2018]	0.823	0.828	0.860	0.403	0.884	0.843
DSSH [Chen et al., 2019]	**0.831**	0.844	0.886	0.411	0.893	0.858
This work	0.825	**0.848**	**0.899**	**0.472**	**0.907**	**0.865**

(a) SHREC13

Method	NN	FT	ST	E	DCG	mAP
Siamese [Wang et al., 2015]	0.239	0.212	0.316	0.140	0.496	0.228
Shape2Vec [Tasse and Dodgson, 2016]	0.714	0.697	0.748	0.360	0.811	0.720
DCML [Dai et al., 2017]	0.272	0.275	0.345	0.171	0.498	0.286
LWBR [Xie et al., 2017]	0.403	0.378	0.455	0.236	0.581	0.401
DCA [Chen and Fang, 2018]	0.770	0.789	0.823	0.398	0.859	0.803
SEM [Qi et al., 2018]	**0.804**	0.749	0.813	0.395	0.870	0.780
DSSH [Chen et al., 2019]	0.796	0.813	0.851	0.412	0.881	0.826
This work	0.789	**0.814**	**0.854**	**0.561**	**0.886**	**0.830**

(b) SHREC14

Method	NN	FT	ST	E	DCG	mAP
Siamese [Wang et al., 2015]	0.118	0.076	0.132	0.073	0.400	0.067
SEM [Qi et al., 2018]	**0.840**	0.634	0.745	0.526	0.848	0.676
DSSH [Chen et al., 2019]	0.838	0.777	0.848	0.624	0.888	0.806
This work	0.816	**0.799**	**0.891**	**0.685**	**0.910**	**0.831**

(c) Part-SHREC14

Table 2.5: **Comparison 3** to many-shot sketch-based 3D shape retrieval on SHREC13, SHREC14, and Part-SHREC14. Having a metric space revolving around semantic prototypes benefits five out of six metrics.

the 3D shapes are now split into 5,812 for training and 1,426 for testing to avoid overlap.

Following previous works [Chen and Fang, 2018, Su et al., 2015, Xie et al., 2017], we generate 2D projections for all 3D shape models using the Phong reflection model [Phong, 1975]. Similarly, we render 12 different views by placing a virtual camera evenly spaced around the unaligned 3D shape model with an elevation of 30 degrees. We only aggregate the multiple views during testing to reduce complexity. We report six retrieval metrics [Li et al., 2014a]. The nearest

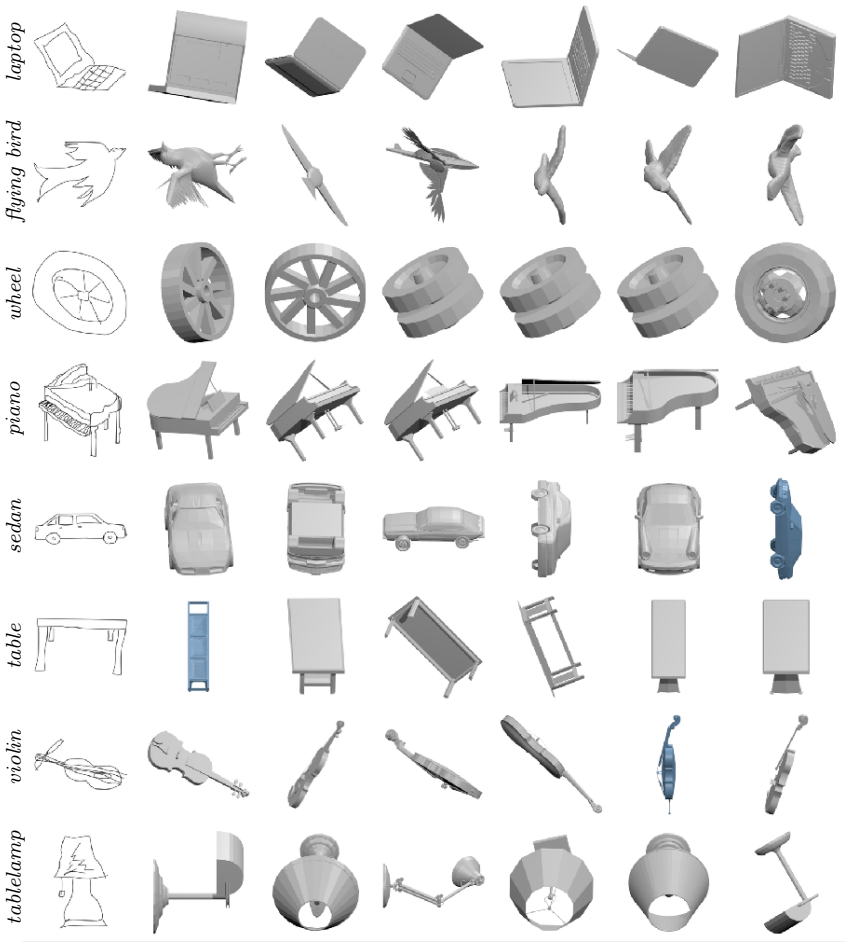


Figure 2.11: **Qualitative analysis** of many-shot sketch-based 3D shape retrieval on Part-SHREC14. Incorrect results are shown in *blue*. Our approach handles the unaligned shapes by projecting all views to the same semantic prototype in the shared space. An open problem remains the confusion with categories that are close both in semantics and in appearance (*e.g.*, "violin" *vs.* "cello").

neighbour (NN) denotes precision@1. The first tier (FT) is the recall@K, where K is the number of 3D shape models in the gallery set of the same class as the query. The second tier (ST) is the recall@$2K$. The E-measure (E) is the harmonic mean between the precision@32 and the recall@32. The discounted cumulated gain (DCG) and mAP are also reported.

Results. Table 2.5 shows the results on all three benchmarks and six metrics. We compare to seven state-of-the-art baselines, which mostly focus on learning a joint feature space of sketches and 3D shapes with metric learning [Chopra et al., 2005, Hadsell et al., 2006, Schroff et al., 2015]. Across all three benchmarks, we observe the same trend, where we obtain the highest scores for five out of the six baselines. Only for the precision@1 metric (NN) do the recent approaches of Chen

et al. [2019] and Qi et al. [2018] obtain higher scores on all three benchmarks. A first reason for this behaviour is that both approaches directly optimize for the nearest neighbour metric. Qi et al. [2018] search in the label space while Chen et al. [2019] perform a learned hashing. A second reason comes from their usage of more complex 3D shape representations. Qi et al. [2018] work with point clouds while Chen et al. [2019] sample 2D views from various viewpoints. Our approach, while simple in nature, provides competitive results compared to the current state-of-the-art in many-shot sketch-based 3D shape retrieval.

Qualitative analysis. To gain insight in our approach for retrieving 3D shapes from sketches, we provide qualitative examples in Figure 2.11. Rotations of unaligned shapes can be handled. For example, 3D shapes of "laptop" or "piano" are retrieved despite the large differences in rotation angles. Yet, confusion remains with visually similar categories. This happens when the search needs to differentiate among fine-grained categories. For example, differences are subtle between "sedan cars" and "sports cars", or between "violin" and "cello". Although errors can appear with semantically similar categories, our method can retrieve highly variable 3D shapes from sketches.

2.6 Conclusion

In this chapter, we open visual search beyond two domains to scale to any number of domains. This translates into a search between any pair of source and target domains, a search from a combination of multiple sources, or a search within a combination of multiple targets. This creates new challenges as all domains should map to the same embedding space, while new domains should be able to be incorporated efficiently. To achieve open cross-domain visual search, we propose a simple approach based on domain-specific prototype learners to align the semantics of multiple visual domains in a common space. Learning a mapping to a common space enables a visual search among any number of source or target domains. The addition of new domains consists in the training of a new prototype learner, without the need to retrain previous models. Empirical demonstrations on novel *open* cross-domain visual search tasks present how to search across multiple domains. State-of-the-art results on existing *closed* cross-domain visual search tasks show the effectiveness of our approach.

Chapter 3

Diversely-Supervised Visual Product Search

3.1 Introduction

This chapter strives to retrieve specific images of products, such as cars or clothes. Searching for product images has a long tradition in computer vision and multimedia, covering query-by-instance [Bell and Bala, 2015, Huang et al., 2015, Kiapour et al., 2015, Liu et al., 2016, Song et al., 2016], query-by-category [Bergamo et al., 2011, Chechik et al., 2009, Deselaers and Ferrari, 2011, Frome et al., 2007], query-by-attribute value [Kovashka et al., 2012, Parikh and Grauman, 2011, Veit et al., 2017, Yu and Grauman, 2014, Zhao et al., 2018b], or query-by-description [Karpathy and Fei-Fei, 2015, Lee et al., 2018, Wang et al., 2019c]. A more targeted search strategy has been proposed recently, in which a query-by-sentence aims to modify attribute values [Ak et al., 2018, Han et al., 2017, Vo et al., 2019, Zhao et al., 2017a] or to generate product instances [Ak et al., 2019, Zhu et al., 2017]. While these previous works consider the similarity of instance, category and attribute labels individually, we aim to integrate them altogether to enable a more expressive product search.

We are inspired by recent works on diverse supervision [Ruder et al., 2019, Ye et al., 2018], which define auxiliary labels in separate branches to benefit a primary task. Ruder et al. [2019] show the benefits of part-of-speech tagging as auxiliary labels for several natural language processing problems. Ye et al. [2018] leverage image-level, box-level and pixel-level annotations jointly for instance segmentation. Encouraged by these seminal works, we introduce diverse supervision to visual product search. We define the search for a given diverse set of labels as our primary task. To achieve this, we learn visual representations for attribute,

Published in *ACM Transactions on Multimedia Computing, Communications, and Applications*, 18(1):1–22. [Thong and Snoek, 2022]

instance, and category labels altogether in an integrated embedding space for a product retrieval task.

Our main contribution is the introduction of diversely-supervised visual product search, where the objective is to search for product images that match to a specific set of diverse labels. For example, we may want to retrieve images of "a *shirt* with *long* sleeves and a *stripe* print", which composes a set of three different labels. For this purpose, we derive an embedding space where interrelations among labels result in interrelated representations. Training relies on a diverse supervision of attribute, instance and category labels to describe images through a diverse representation. For every label, we compute a representation by aggregating the corresponding items in the training set. We propose an evaluation based on composite queries for diversely-supervised product search. We represent composite queries by averaging the aggregated visual representations of each diverse label of the specific set. As such, we create two diversely-labeled datasets, which build upon existing clothes [Liu et al., 2016] and cars datasets [Krause et al., 2013, Yang et al., 2015]. Evaluation on these two datasets shows the benefits of our embedding for diversely-supervised product search in seen and unseen settings, and for discovering the typicality effect of product styles. All source code and setups are released to foster further research in diversely-supervised visual product search [1].

3.2 Related work

Visual product search has attracted a lot of interest from social media platforms [Jing et al., 2015, Zhai et al., 2017, 2019] or online e-retailers [Yang et al., 2017, Zhang et al., 2018b], as they need to recommend products to users. In multimedia and computer vision applications, this interest in visual product search has been translated into different retrieval problems. Each problem comes with its own challenge and offers new ways to search for products.

One line of work follows the traditional instance retrieval problem where an example image is used as a query. The objective is to retrieve images of the same product in a gallery set within the same domain [Liu et al., 2016, Song et al., 2016] or across domains [Bell and Bala, 2015, Huang et al., 2015, Kiapour et al., 2015, Liu et al., 2012, 2016]. Product categories can also be related to each other to retrieve complementary products for recommendation by capturing a global description of style [Hsiao and Grauman, 2018, Kiapour et al., 2014, McAuley et al., 2015, Veit et al., 2015].

Another line of work covers image captioning where a description is matched to an image [Karpathy and Fei-Fei, 2015, Lee et al., 2018, Wang et al., 2019c]. The idea is to learn a multimodal embedding where text and image representations are aligned together [Karpathy and Fei-Fei, 2015]. Grounding words in the image is

[1] Source code is available at https://github.com/twuilliam/diverse-search

particularly important to capture the interactions between both modalities [Lee et al., 2018, Wang et al., 2019c]. In this chapter, the search task is complementary to text-image retrieval as we consider an unordered set with a varying number of labels instead of a fixed description sentence.

Finally, another line of work explores relevance feedback to integrate input from the user. This can consist of a comparison of product pairs to assess the relative strength of attributes [Kovashka et al., 2012, Parikh and Grauman, 2011, Yu and Grauman, 2014], to verify that they exhibit the same attribute value [Veit et al., 2017, Zhao et al., 2018b], or indicate a location of attribute interest [Huang et al., 2014]. Alternatively, the user can manipulate one attribute value to retrieve [Ak et al., 2018, Han et al., 2017, Vo et al., 2019, Zhao et al., 2017a] or to generate [Ak et al., 2019, Zhu et al., 2017] the targeted product. In this chapter, we introduce a complementary problem: we search for products that match to a specific, yet diverse, set of labels.

Diverse labels. Searching for a diverse set of labels has mainly focused on describing images with multiple binary attributes. Multi-attribute queries are used to search for images of faces [Kumar et al., 2011, Scheirer et al., 2012, Siddiquie et al., 2011], by describing the absence or presence of facial traits. The conjunction of positive binary attribute values has also proven to be useful in animal categorization, in a retrieval setting [Rastegari et al., 2013] or a zero-shot classification setting [Akata et al., 2016, Farhadi et al., 2009, Lampert et al., 2014]. While attributes are important to describe objects, they are not specific enough for producing a product search [Ferrari and Zisserman, 2008]. Different from these works, we aim to learn (a) image similarities through a diverse set of labels which go beyond attributes by including category and instance labels; and (b) an embedding space that encodes every label with real-valued vector representations rather than binary representations.

Structured queries have also been proposed to capture a diverse set of relations for complex scene retrieval. Sentence queries go beyond simple keywords to capture relations among objects [Gordo and Larlus, 2017, Sadeghi and Farhadi, 2011, Vo et al., 2019, Wang and Hebert, 2016]. Graph queries structure explicitly these relations [Chaudhary et al., 2020, Johnson et al., 2015, Lan et al., 2012]. Paragraph queries enable the retrieval of an image sequence to illustrate a story [Kim et al., 2015, Ravi et al., 2018]. In this work, we rely on a diverse set of label vocabularies to structure product retrieval. We form composite query representations by averaging over the visual representations of the desired labels to search for.

Diverse representations. Encoding multiple labels into an embedding space is usually done through two different approaches. One approach is to learn a global representation of images [Liu et al., 2016, Yang et al., 2015] to classify categories and attribute values. An alternative approach is to learn a subspace for each attribute to create distinct and disentangled similarities [Veit et al., 2017]. Vari-

ants of this approach enhance the backbone network to modulate channels either with a learned real-valued vector to promote constructive interference [Zhao et al., 2018b], or by a fixed binary mask to model task relationships in a non-parametric manner [Strezoski et al., 2019]. Yet, these approaches are restricted to comparing attribute [Veit et al., 2017] or instance [Liu et al., 2016, Yang et al., 2015] labels. In this chapter, we propose to encode attribute, instance and category labels in an integrated manner, by explicitly establishing their interrelationships.

3.3 Method

3.3.1 Problem statement

During the training, we are given a training set of product images $\mathcal{X}_{train}$. Each image $\mathbf{x}$ in the training set comes along with a diverse set of labels. In particular, we are interested in the category label $y \in \mathcal{C}$, the label v of attribute $k \in \mathcal{A}_k$ and the instance label $i \in \mathcal{I}_{train}$. $\mathcal{C}$ is the category vocabulary of C product categories. As products can express multiple attributes, we consider K different attribute vocabularies $\mathcal{A}_k$ with A_k attribute values each. Hence, images also have multiple attribute labels, forming multiple tuples (k, v) with $k = 1, \cdots, K$ and $v = 1, \cdots, A_k$. $\mathcal{I}_{train}$ is the set of instances in the training set. Instances are an integral part of visual products. Images of the same instance usually differ by a different viewpoint or background. Hence, the instance labels enforce images of the same product to be close to each other. Overall, we leverage all $\{\mathcal{C}, \mathcal{A}_1, \cdots, \mathcal{A}_K, \mathcal{I}_{train}\}$ labels to provide a diverse supervisory signal to the model during training.

During the evaluation, we are given a gallery set of images $\mathcal{X}_{gal}$, which originates from a separate set of products. Formally, $\mathcal{I}_{train} \cap \mathcal{I}_{gal} = \emptyset$. The gallery set $\mathcal{X}_{gal}$ shares the same category vocabulary $\mathcal{C}$ and K attribute vocabularies $\mathcal{A}_k$ with the training set $\mathcal{X}_{train}$. As such, these vocabularies serve to build a set of labels for describing composite queries used for retrieving product images. An example of such a search is to retrieve clothes images that match "a *shirt* with *long* sleeves and a *stripe* print", where the set of labels comprises one product category and values for two different attributes. Separating the instances in the gallery set $\mathcal{X}_{gal}$ from the training set $\mathcal{X}_{train}$, allows to evaluate the generalization ability of the model on new products which express both seen and unseen combinations of categorical and attribute values.

3.3.2 Diversely-supervised embedding

We propose to learn a diversely-supervised embedding space where Euclidean distances capture label similarities. The embedding space is motivated by the definition of attribute, instance, and category for describing products: (a) prod-

ucts are instances of particular categories, and (b) attributes characterize visual properties of products. For example, a "3-Series sedan" is an instance of the "BMW" car category with "4 doors" and "5 seats" attributes. In this context, attribute and instance labels are highly interrelated to each other because attributes qualify instances. Our technical contribution lies in how to explicitly encode these label definitions in the diversely-supervised embedding space.

To learn a representation for each label, we rely on a cross-entropy loss with softmax embedding [Liu et al., 2017b, Movshovitz-Attias et al., 2017, Snell et al., 2017]. While originally proposed for either instance retrieval [Liu et al., 2017b, Movshovitz-Attias et al., 2017] or few-shot learning [Snell et al., 2017], we develop a variant for learning a representation from a diverse set of labels. Different from the commonly used contrastive [Chopra et al., 2005, Hadsell et al., 2006] or triplet [Schroff et al., 2015, Weinberger and Saul, 2009] losses, the proposed loss doesn't require any intricate sampling, which makes the training with diverse supervision much simpler. We derive below how to learn representations for each label type in the embedding space.

Attribute representations. We encode attribute labels in subspaces, one per attribute. A dataset with K attributes results in an embedding with K subspaces. Let $\mathbf{h} = f_\theta(\mathbf{x})$ be the features $\mathbf{h}$ of an image $\mathbf{x}$ from a convolutional network f with trainable parameters θ. The idea is to learn a linear projection of the features $\mathbf{h}$ in multiple separate subspaces to encode the representation for each attribute $k = 1, \cdots, K$ in a representation $\mathbf{z}_{A_k} \in \mathbb{R}^d$:

$$\mathbf{z}_{A_k} = \mathbf{W}_k \mathbf{h} + \mathbf{b}_k \tag{3.1}$$

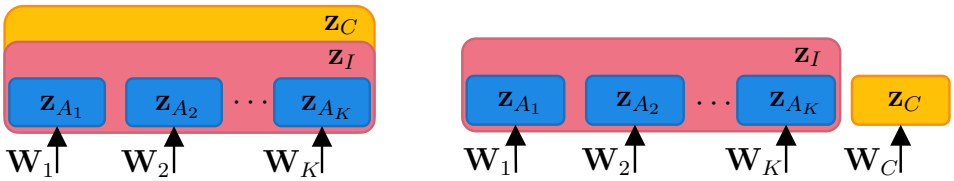
where $\mathbf{W}_k$ and $\mathbf{b}_k$ are the weights and biases, respectively. We learn the attribute representation based on the cross-entropy loss with softmax embedding:

$$\mathcal{L}_{A_k} = -\log \frac{\exp(-\|\mathbf{z}_{A_k} - \mathbf{a}_{k,v}\|)}{\sum_{z \in \mathbb{Z}_{A_k}} \exp(-\|\mathbf{z}_{A_k} - \mathbf{a}_{k,z}\|)}, \tag{3.2}$$

where $\|\cdot\|$ is the Euclidean distance, $\mathbb{Z}_{A_k}$ denotes the set of all the latent prototypes $\mathbf{a}_{k,v} \in \mathbb{R}^d$ of attribute k. The softmax embedding function provides a probability of the attribute representation $\mathbf{z}_{A_k}$ to be recognized as the value v of attribute k. At each step, the model pulls $\mathbf{z}_{A_k}$ to its corresponding latent prototype $\mathbf{a}_{k,v}$, and pushes it away from the prototypes of other values $\mathbf{a}_{k,z}$.

Instance representations. We establish an interrelation between instance and attribute representations. As attribute labels qualify product instances, we encode this property in the embedding space. The instance representation $\mathbf{z}_I \in \mathbb{R}^D$ with $D = K \cdot d$ corresponds to the concatenation of all attribute subspaces:

$$\mathbf{z}_I = \bigcup_{k=1}^{K} [\mathbf{z}_{\mathrm{A}_k}], \tag{3.3}$$



(a) *Triple* grouping (b) *Dual* grouping

Figure 3.1: **Diversely-supervised embeddings.** We consider attributes (*blue*), category (*yellow*) and instance (*pink*) representations. Given the output features of a convolutional network, we learn multiple linear projections $\mathbf{W}$ to an embedding space. (a) The *triple* grouping makes the embedding axis-aligned on attributes for both instances and categories. Average representations of instances form category representations. (b) The *dual* grouping treats category representations in a separate subspace.

where $\bigcup[\cdot]$ is the vector concatenation operator. Similarly, we learn the instance representation based on the cross-entropy loss with softmax embedding:

$$\mathcal{L}_I = -\log \frac{\exp(-\|\mathbf{z}_{\mathrm{I}_k} - \mathbf{p}_i\|)}{\sum_{z \in \mathbb{Z}_I} \exp(-\|\mathbf{z}_{\mathrm{I}_k} - \mathbf{p}_z\|)}, \tag{3.4}$$

where $\mathbb{Z}_I$ denotes the set of all the latent instance prototypes $\mathbf{p}_i \in \mathbb{R}^D$.

Category representations. We propose two different variants to encode the category labels, as illustrated in Figure 3.1: (a) the *triple* grouping ensures that representations from instances of the same category are close to each other (Figure 3.1a), while (b) the *dual* prefers to encode the category labels separately (Figure 3.1b). We also learn the category representation based on the cross-entropy loss with softmax embedding:

$$\mathcal{L}_C = -\log \frac{\exp(-\|\mathbf{z}_C - \mathbf{c}_y\|)}{\sum_{z \in \mathbb{Z}_C} \exp(-\|\mathbf{z}_C - \mathbf{c}_z\|)}, \tag{3.5}$$

where $\mathbb{Z}_C$ denotes the set of all latent category prototypes $\mathbf{c}_y$. In the *triple* grouping, category representations are a concatenation of attribute subspaces $\mathbf{z}_C \in \mathbb{R}^D$. Hence, $\mathbf{z}_C$ is also a concatenation of a series of $\mathbf{z}_{A_k}$, just like $\mathbf{z}_I$. Though, we impose a constraint on Z_C such that grouping instance representations form category representations. In other words, without the loss on $\mathbf{z}_C$ instance representations would be free to organize themselves in the embedding space. Formally, the category representation corresponds to:

$$\mathbf{c}_y = \frac{1}{|\mathbb{Y}|} \sum_{i \in \mathbb{Y}} \mathbf{p}_i, \tag{3.6}$$

where $\mathbb{Y}$ is the set of all latent instance prototypes of the category y. In the *dual* grouping, they are linearly projected to their own subspace $\mathbf{z}_C \in \mathbb{R}^d$.

The grouping motivation differs by the assumptions on how to relate instance, category and attribute labels. We assume that attributes qualify instances, and categories emerge by grouping instances. This leads to the *triple* grouping, where all three types of labels are interrelated. The *dual* grouping relaxes the category assumption, by only interrelating attributes and instances. The former incorporates the fact that categories and instances play opposite roles: categories force the embedding to be agnostic to instances, while instances force the embedding to focus on fine-grained differences making categories harder to learn.

Training. The training objective of the diversely-supervised embedding corresponds to a minimization of a weighted sum of representations for each type of labels:

$$\mathcal{L} = \lambda_I \mathcal{L}_I + \frac{\lambda_A}{K} \sum_k \mathcal{L}_{A_k} + \lambda_C \mathcal{L}_C + \lambda_R \|\mathbf{z}\|^2, \tag{3.7}$$

where λ_I, λ_A, and λ_C denote trade-off hyperparameters to control the contribution of each type of label. Some images might not express all attributes K defined in the dataset, *e.g.*, a *skirt* doesn't have a *sleeves length* attribute. In this case, the contribution of the missing attribute in Eq. 3.7 is ignored. We also apply an ℓ_2 regularization on the final representation $\mathbf{z}$, which encodes all label types. In the *triple* grouping, the final representation is $\mathbf{z} = \mathbf{z}_I \in \mathbb{R}^D$ while in *dual*, $\mathbf{z} = [\mathbf{z}_I; \mathbf{z}_C] \in \mathbb{R}^{(K+1)\cdot d}$.

Prototype updates. To design the probabilistic model, we take inspiration from the prototype literature [Liu et al., 2017b, Movshovitz-Attias et al., 2017, Snell et al., 2017], where the general idea is to apply a softmax over distances to prototypes. Different from prototypical networks [Snell et al., 2017], we consider prototypes as latent parameters, which are initialized randomly and updated throughout the training like any other neural network parameters. In other words, the backward pass also includes the partial derivative of the loss with respect to all latent prototypes. This differentiates us from prototypical networks [Snell et al., 2017]. Indeed, rather than defining prototypes as the average of support image representations, our prototypes are latent representations that are updated during training. Compared to a classification setting [Snell et al., 2017], no support images are present in retrieval that is why we design prototypes as latent representations as usually done in instance retrieval [Movshovitz-Attias et al., 2017, Zhai and Wu, 2019].

Implementation details. The backbone network relies on ResNet50 [He et al., 2016], pre-trained on ImageNet [Russakovsky et al., 2015]. To produce the embedding space, the classification layer is removed and replaced by the multiple linear projections with a random weight initialization. Latent prototypes are also initialized with random weights. During training, the model minimizes the loss function described in Eq. 3.7 using the Adam stochastic optimizer algorithm [Kingma and

Ba, 2015]. Images are cropped given their bounding box labels and resized to 224×224, and augmented with horizontal flipping. Hyper-parameters are the following: minibatch size of 128, learning rate of $1e{-}4$, $\beta_1 = 0.9$, $\beta_2 = 0.999$, weight decay of $5e{-}5$, and subspaces are of size $d = 50$. We set the trade-offs to $\lambda_I = \lambda_C = \lambda_A = 1$ and $\lambda_R = 1e - 3$. Updates of the latent prototypes operate at a learning rate $10\times$ higher. The learning rate undergoes a cosine annealing decay without restart [Loshchilov and Hutter, 2017]. We set hyper-parameters according to the classification accuracy of attributes on the validation set. The implementation relies on the PyTorch framework [Paszke et al., 2019].

3.3.3 Composite queries representations

During the evaluation, we query the gallery set $\mathcal{X}_{gal}$ with composite queries derived from the training set. We represent composite queries by a real-valued vector $\mathbf{q} \in \mathbb{R}^D$ of M diverse labels. In other words, given a composite query $\mathbf{q}$, the idea is to retrieve product images in the gallery set $\mathcal{X}_{gal}$ from their visual representations $\mathbf{z}$ that match a specific set of M labels. To form composite query representations, we average the representations from the training set of each $m \in M$ label individually and take the overall average. Formally, this corresponds to a per-label averaging:

$$\mathbf{q} = \frac{1}{M} \sum_{m=1}^{M} \frac{1}{|\mathcal{M}_m|} \sum_{n \in \mathcal{M}_m} \mathbf{z}^{(n)} \tag{3.8}$$

where $\mathcal{M}_m$ is the set of training images that exhibits label m with $m = 1, \cdots, M$. The inner sum averages the representations $\mathbf{z}$ of all images $n \in \mathcal{M}_m$ for each label m. The outer sum calculates an average of averages to create a composite query representation $\mathbf{q}$ that includes all M labels. If normalization is done globally (*i.e.*moving $1/|\mathcal{M}_m|$ to the outer sum), it corresponds to a per-sample averaging.

3.4 Experimental details

3.4.1 Diversely-labeled datasets

We introduce two datasets for diversely-supervised visual product search: *Diverse – Cars* and *Diverse–Clothes*. Both datasets include instance, category and multiple attributes labels. Figure 3.2 illustrates some diversely-labeled examples for each dataset.

Diverse–Cars. We build upon Cars196 by Krause et al. [2013] and CompCars by Yang et al. [2015] to create Diverse–Cars. The original datasets intend to tackle fine-grained categorization and verification, we merge them for the task of diversely-supervised product search. This creates a dataset that covers car models

Instance: 3 Series sedan
Category: BMW
Attributes
·*Doors*: 4
·*Seats*: 5
·*Type*: sedan

(a) **Diverse–Cars**

Instance: 6096
Category: Jacket
Attributes
·*Fabric*: denim ·*Print*: distress
·*Front*: buttons·*Shoulder*: n.a.
·*Hemline*: n.a. ·*Sleeves*: long
·*Neckline*: hood·*Silhouette*: cropped

(b) **Diverse–Clothes**

Figure 3.2: **Diversely-labeled examples** from Diverse–Cars and Diverse–Clothes.

sold in both North American and South-Pacific regions. We manually annotate Diverse–Cars to merge car model duplicates and to provide clean annotations for car makers and car attributes. Diverse–Cars defines 97 car makers and 3 car attributes. Every attribute is further defined with the specific values: 4 *number of doors*, 4 *number of seats* and 12 *type* values. In total, Diverse–Cars contains 28,423 images from 386 car models for training and 22,450 images from 305 separate car models for evaluation.

We manually re-annotate the images to ensure the quality of the *category* and *attribute* labels. Besides the new category and attribute labels, we also ensure that similar car models between the two original datasets are merged. The new labels will be made public. In the newly proposed labels, *category* and *attribute value* labels are annotated. Original *instance* labels are preserved. We adopt the same three attribute vocabularies as initially defined in CompCars [Yang et al., 2015]. Figure 3.3 shows one sample for every attribute value of every attribute.

Overall, a total of 691 unique instances are annotated. Every image in the dataset receives an instance, a category and three attribute value labels. Note that some categories are very scarce. We ensure that there are at least one or two models per car maker in the training set, which in return can result in the absence of some car makers in the gallery set. In other words, not all car makers are present in the gallery set. For hyper-parameters search, we create a separate validation set from the training set. We randomly sample 17 car models, for a total of 1,169 images. We keep the validation separate. There is no re-training on both training and validation sets once hyper-parameters are fixed.

Diverse–Clothes. We build upon In-Shop Clothes by Liu et al. [2016] to create Diverse–Clothes. The original dataset provides a large number of clothing products along with multiple views and a rich description of several sentences, but the provided labels are known to contain scarce attribute values, duplicates and incoherencies [Zakizadeh et al., 2018]. Hence, we manually re-annotate the dataset to provide clean annotations for clothes categories and clothes attributes. Diverse–Clothes defines 12 clothes categories and 8 clothes attributes. Every attribute comes with specific attribute values: 6 *fabric*, 7 *frontal feature*, 6 *hemline*, 13 *neckline*, 15 *print*, 4 *shoulder line*, 6 *sleeves length* and 2 *silhouette* values.

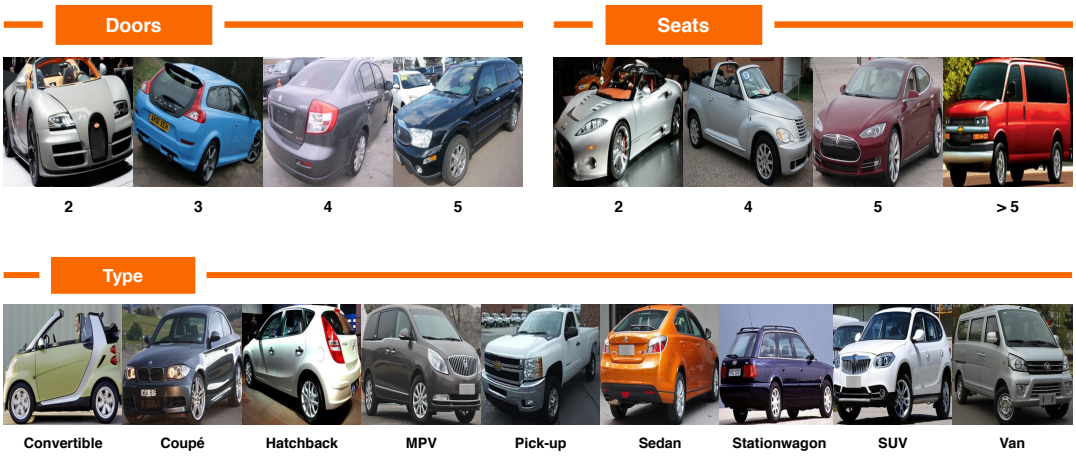


Figure 3.3: **Image samples for every attribute value** in Diverse–Cars.

In total, Diverse–Clothes contains 25,862 images from 3996 fashion products for training and 26,797 images from 3982 separate fashion products for evaluation.

Diverse–Clothes builds upon In-Shop Clothes by Liu et al. [2016], which provides a large number of clothing products. Every product comprises multiple images from several viewpoints and a rich description of several sentences. However, the labeling of the original In-Shop Clothes dataset was done in a weakly-supervised manner, which can result in scarce attribute values, duplicates, or incoherencies [Zakizadeh et al., 2018].

We manually re-annotate the images to ensure the quality of the *category* and *attribute* labels. Besides the new category and attribute labels, other cleaning tasks are also performed: (1) instance and image duplicates are removed; (2) instances with two different category labels are merged. The new labels will be made public. In the newly proposed labels, *category* and *attribute value* labels are re-annotated. Original *instance* labels are preserved. Eight different new *attributes* are defined. Figure 3.4 shows one sample for every value of every attribute.

Overall, a total of 7,978 unique instances are re-annotated. While a category label and an instance label are assigned to all instances, not all attribute labels are necessarily assigned to all instances. For example, a *skirt* does not have a *sleeves length* attribute label. For hyper-parameters search, we create a separate validation set from the training set. We sample 59 clothes items, for a total of 352 images. We keep the validation separate and do not re-train on it once hyper-parameters are fixed.

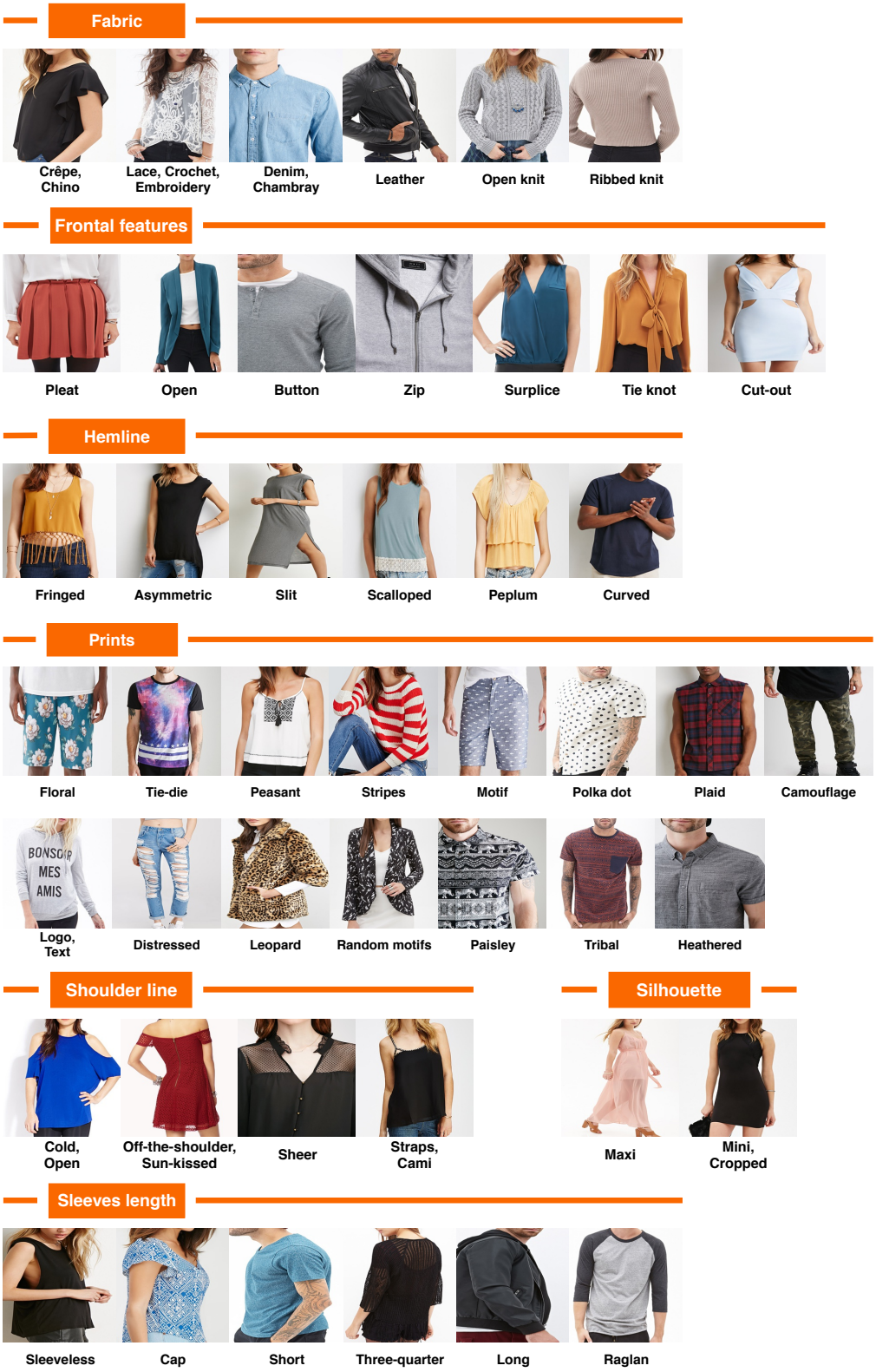
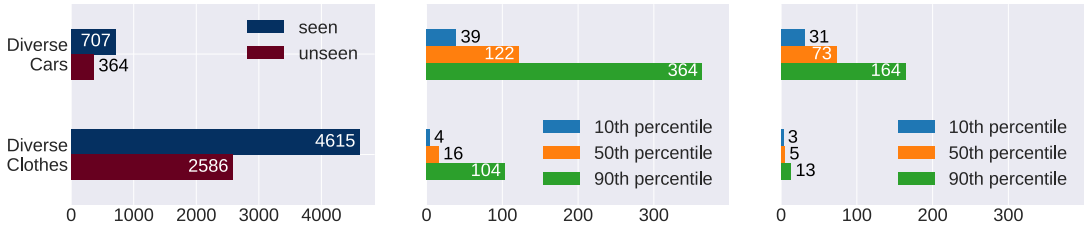


Figure 3.4: **Image samples for every attribute value** in Diverse–Clothes.



(a) Numbers of queries (b) Numbers of images per *seen* query (c) Numbers of images per *unseen* query

Figure 3.5: **Composite queries distribution** per dataset. Unseen queries represent a third of all queries. Diverse–Clothes appears to be more challenging than Diverse–Cars given the very few images per query (in both *seen* and *unseen* scenarios).

3.4.2 Composite queries

We leverage composite queries to retrieve images in the gallery set $\mathcal{X}_{gal}$ that share the same set of labels. In this paper, we define a composite query as a composite between a category label and one or multiple attribute labels, for a total of M types of label. An example of composite query for cars can be "a *BMW* with *2* doors, *5* seats and with a *coupé* type" ($M = 4$) while an example for clothes can be "a *shirt* with *long* sleeves and a *stripe* print" ($M = 3$). To avoid searching a needle in a haystack, we limit the number of attribute labels in composite queries to a maximum of three.

During the evaluation, we separate *seen* from *unseen* composite queries. If there is at least one image in the training set $\mathcal{X}_{train}$ that corresponds to the composite query, then the composite query is identified as *seen*. If the combination of category and attributes doesn't exist in the training set $\mathcal{X}_{train}$, then the composite query is *unseen*. Unseen composite queries are more realistic and more challenging than seen composite queries because their combination has never been encountered by the model during training.

For each dataset, we generate composite queries by considering all possible category and attribute combinations and select the valid ones. A query is valid if there is at least one image in the gallery set $\mathcal{X}_{gal}$ with this specific combination. Figure 3.5 presents the distribution of seen and unseen composite queries for both datasets. Unseen composite queries constitute more than a third of the total queries, which illustrates the difficulty of both benchmarks. Diverse–Clothes provides a more challenging evaluation than Diverse–Cars. Indeed, there is a high number of queries and very few images per query considering the large gallery size. For example, given an unseen query, the median number of images per query is only 5 while the the gallery size is 26,797 images. Searching for new clothes products emerges as a more difficult task than searching for new car models, given the large diversity of fashion items.

3.4.3 Diversely-supervised search

Evaluation is performed on the gallery set $\mathcal{X}_{gal}$ that contains separate instances from the training set $\mathcal{X}_{train}$, as defined in Section 3.3.1. In other words, while a diverse set of labels defined by the composite query might have been *seen* or *unseen* during training, instances in the gallery have never been seen before. This is the protocol commonly used in zero-shot instance retrieval (*e.g.*, [Liu et al., 2016, Song et al., 2016]), where no overlap exists in terms of images nor instances between the training and the gallery sets. An ℓ_2 normalization is applied to the representations before measuring distances between the composite query $\mathbf{q}$ and the gallery $\mathcal{X}_{gal}$. A retrieved image is considered as a hit if it shares the set of labels with the composite query. We report the mean average precision (mAP) [Manning et al., 2008] across *seen*, *unseen* combinations for composite queries, and the *overall*, to measure the performance.

3.5 Results

3.5.1 Comparison with alternatives

We adapt four existing methods, designed for a different purpose, in such a way that they become applicable to our setting. For fair comparisons, we apply the same procedure on these alternative models for both training and evaluation. We also use the same similarity loss based on the softmax embedding loss with prototypes. Below, we detail how each selected method is repurposed:

- **Global** maps an image $\mathbf{x}$ to a global representation $\mathbf{h}$, of the same dimension as our model with partial grouping. Inspired by Liu et al. [2016], we add common softmax classification heads on top of the global embedding space to predict values for every K attributes and for categories. In other words, this corresponds to a multitask model with multiple heads. An additional similarity loss on the global embedding space models instance representations. In our setting, the final embedding used for evaluation is the global representation space.

- **Conditional** gives every label its own metric subspace z, as originally introduced for attributes by Veit et al. [2017]. Compared with our proposed method, conditional does not include any grouping mechanism. We add for every subspace label a loss that measures image similarities. In our setting, the final embedding used for evaluation concatenates attribute, category and instance subspaces.

- **Modulation** controls the amount of feature sharing for every type of labels, as originally proposed for attributes by Zhao et al. [2018b]. Similar to

Method	seen	unseen	all
Global	33.20	19.45	28.53 ± 0.20
Conditional	33.83	19.83	29.00 ± 0.13
Modulation	32.80	17.57	27.63 ± 0.46
Routing	29.87	16.02	25.17 ± 0.14
This work	**37.61**	**21.03**	**31.98** ± 0.30

(a) **Diverse–Cars**

Method	seen	unseen	all
Global	8.58	4.15	6.88 ± 0.08
Conditional	8.01	3.60	6.33 ± 0.13
Modulation	8.61	4.12	6.89 ± 0.08
Routing	6.61	2.56	5.06 ± 0.11
This work	**9.67**	**4.56**	**7.72** ± 0.13

(b) **Diverse–Clothes**

Table 3.1: **Comparison with alternatives**. We adapt four existing methods, designed for a different purpose, in such a way that they become applicable to our setting (details provided in Section 3.5.1). We report the average over three runs. Our embedding outperforms these alternatives in mAP (in %) on both Diverse–Cars and Diverse–Clothes datasets. Integrating attribute, instance and category representations altogether in the embedding space with interrelated representations helps to model a diverse set of labels.

conditional, every type of label representation is also delimited to its subspace. Though, the main difference with conditional lies in the backbone network, which produces different features per label. Instead of having an explicit subspace per label during training, the idea is to encode the label information by transforming the activations of the backbone with a learned real-valued vector to weight every channel. This offers a compelling and efficient way to have label-specific feature representations without the need to train label-specific models. Following Zhao et al. [2018b], modulation occurs after the last two residual blocks (*i.e.*, block3 and block4). In our setting, the final embedding concatenates the modulated attribute, category and instance representations.

- **Routing** zeroes out channels given a type of labels, as originally proposed for many task learning by Strezoski et al. [2019]. Routing is in the same spirit as modulation, and the difference lies in the usage of fixed binary masks to transform the activations of the backbone rather than learned real-valued vectors. Following Strezoski et al. [2019], we generate binary masks by sampling a binomial distribution with a probability of success of 0.6. Similar to modulation, we apply the routing module after the last two residual blocks. In our setting, the final embedding concatenates the routed attribute, category and instance representations.

Results on Diverse–Cars. Table 3.1a shows that our diversely-supervised embedding outperforms alternative ways to combine attribute, category and instance subspaces. Interestingly, channel-modulated methods based on a real-valued or binary masks achieve a lower retrieval score than the non-modulated conditional

counterpart. As cars depict clear attribute values, their representation doesn't really benefit from creating a feature weighting. Indeed, there is no middle ground between *3* and *4* doors while there might exist a debate to decide whether the sleeves length is *long* or *three-quarter*. When comparing with the conditional embedding, the diversely-supervised embedding shows a large improvement. Integrating attribute, instance and category representations altogether in the embedding space, rather than separating them all, helps to capture the diverse set of labels needed for diversely-supervised search.

Note that for fair comparison, we implement alternatives with the same prototype loss as our method. For example, the conditional alternative of Veit et al. [2017] has been initially proposed with a triplet loss. When training conditional with a triplet loss [Schroff et al., 2015], the mAP drops by 9.74% on Diverse-Cars. As triplets only capture one label at a time, results degrade in a multiple labels setting. Our proposed loss with latent prototypes allows us to capture all labels simultaneously, which results in an increased performance for the alternatives and our proposed model.

Results on Diverse–Clothes. Table 3.1b confirms the benefits of the diversely-supervised embeddings on this more challenging dataset. When products exhibit more subjective attribute values, modulation has an edge over the non-modulated conditional counterpart. The routing module struggles the most as zeroing out channels destroys information needed when measuring distances in the embedding space. When comparing the inference time, we notice the channel modulated methods have a linear complexity to the number of subspaces as every labels comes with a modulated representation. This is different from global, conditional and ours that have a constant complexity, as they do not need to be channel-modulated. Our integration of attribute, instance and category representations in the embedding space, also captures these more subtle attribute changes without the need to modulate the backbone.

3.5.2 Ablations

Per-sample *vs.* per-label averaging. We study two alternatives to represent composite queries in the embedding space, as defined in Eq. 3.8. Recall that we collect all visual representations corresponding to every label and average them either per-sample or per-label to form a representation for composite queries. Table 3.2 shows that a per-label averaging outperforms a per-sample averaging on Diverse–Cars. When averaging per-sample, all sample images are considered equally in the composite query. If a label is over-represented in the training set, a per-sample averaging will then result in a composite query biased towards this dominant label. When averaging per-label, all labels are instead considered equally. If a label is over-represented in the training set, a per-label averaging will mitigate the imbalance effect as an equal weight is put to each label representation

Averaging	seen	unseen	all
Per-sample	6.44	5.06	5.97
Per-label	**35.17**	**19.28**	**29.77**

Table 3.2: **Per-sample *vs.* per-label averaging** on Diverse–Cars. Weighting per-sample biases in the query, which degrades the mAP (in %) score.

Grouping	seen	unseen	all
Triple	35.17	19.28	29.77
Dual	**38.10**	**20.71**	**32.19**

Table 3.3: **Triple *vs.* dual grouping** on Diverse–Cars. Separating the category representation leads to an mAP (in %) improvement.

to produce the composite query. For the remaining experiments, we then rely on a per-label averaging for composite queries to avoid a strong bias towards the dominant label.

Triple *vs.* dual grouping. In this experiment, we evaluate the difference between the triple and dual grouping in the embedding (Figure 3.1). The grouping motivation differs by the assumptions on how to relate instance, category and attribute labels; and the practical application. In the triple grouping, attributes qualify instances, and grouped instances form categories. With all three types of labels interrelated in one single embedding space, this allows to explore the dataset to discover trends, as illustrated in Figure 3.11. The dual grouping relaxes the category assumption, as categories are now in a separate subspace. This avoids the duality where the embedding focuses on fine-grained instance differences while trying to group them for form categories at the same time. Table 3.2 shows that the dual variant outperforms the triple one on Diverse–Cars. A competing duality appears between instances and categories: focusing on categories pushes the embedding to be agnostic to instances differences. The triple variant allows an interrelated exploration of products, as all diverse label representations are axis-aligned. Yet, putting the category representations in another subspace better helps the diversely-supervised search. Additionally, we evaluate a variant where category labels are treated like any other attribute labels. In this variant, we obtain a 29.34% mAP. This reinforce the observation that category labels are then different from attributes and need to be treated accordingly. Depending on the application, it can be advantageous to separate instance and category representations. For the remaining experiments in this section, we use the dual grouping as it yields the best scores for both seen and unseen queries.

Pre-training. We explore the effect of self-supervised pre-training on our model. We rely on MoCo v2 [Chen et al., 2020, He et al., 2020] for the self-supervision training, and use the same hyper-parameters as proposed originally. Once trained, we use these weights to initialize the ResNet50 backbone of our model. Table 3.4 compares a pre-training on ImageNet [Russakovsky et al., 2015] with self-supervision on both Diverse-Cars and Diverse-Clothes. On both datasets, pre-training on ImageNet outperforms a pre-training with self-supervision. Dur-

Pre-training	Cars	Clothes
Self-supervised	16.21	3.53
ImageNet	**32.19**	**7.74**

Table 3.4: **Pre-training comparison** on Diverse–Cars and Diverse–Clothes. Pre-training on ImageNet improves by a factor two on the diverse search of all queries (mAP, in %) compared with a self-supervised pre-training. ImageNet acts as a regularizer.

Swapping	Cars	Clothes
✓	0.86	0.33
	32.19	**7.74**

Table 3.5: **Swapping backbones** between Diverse–Cars and Diverse–Clothes. When swapping the backbones, the diverse search of all queries yields a very low performance (mAP, in %). Backbone features are specific to each dataset.

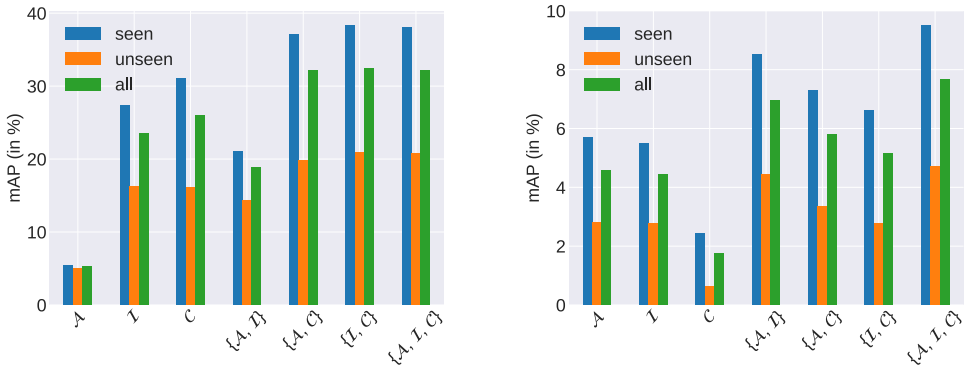
Search space	Model	Fine-tuning	Cars	Clothes
Features	Self-supervision		1.35	0.45
Features	Pre-trained		0.91	0.59
Features	Pre-trained	✓	22.02	3.64
Embedding	Pre-trained	✓	**32.19**	**7.74**

Table 3.6: **Search space comparison** on Diverse–Cars and Diverse–Clothes. Fine-tuning on the respective datasets yields a significant mAP (in %) improvement over models trained in a supervised or self-supervised setting on ImageNet. Diversely-supervised search benefits significantly when the search occurs in the embedding space, which captures all label types as opposed to the feature space.

ing training, we notably observe an overfitting effect with models initialized with self-supervision. Indeed, the training set of both datasets is several orders of magnitude smaller than ImageNet. Thus, a pre-training on ImageNet acts as a regularizer to help models generalize to diversely-supervised search.

Swapping backbones. To understand the importance of backbone features in the generalization performance on diversely-supervised search, we swap the backbone network trained on Diverse–Cars with the one trained on Diverse–Clothes, and vice-versa. Concretely, $\mathbf{h}$ in Eq. 3.1 for Diverse–Cars comes from the backbone f_θ of Diverse–Clothes, and vice-versa. Table 3.5 shows the negative effect of swapping backbones. In either scenario, swapping the backbone drops the performance close to zero. This means that the backbone features, as well as the linear projections, are dataset-specific as they cannot generalize across datasets.

Search space. We assess the importance of the embedding space for diversely-supervised product search by comparing with a search in the feature space. Concretely, we compute the composite query representation in Eq. 8 from $\mathbf{h}^{(n)}$ instead



(a) **Diverse–Cars** (b) **Diverse–Clothes**

Figure 3.6: **Influence of diverse labels.** In both datasets, the instance supervision $\mathcal{I}$ is essential. For Diverse–Cars, the category supervision $\mathcal{C}$ matters the most while for Diverse–Clothes the attribute supervision $\mathcal{A}$ is the most important. A combination of supervision results in an improvement for both seen and unseen composite queries.

of $\mathbf{z}^{(n)}$, where $\mathbf{h}^{(n)}$ corresponds to the output of the backbone convolutional network for the n-th sample and $\mathbf{z}^{(n)}$ to the output of the embedding layer. Table 3.6 shows the benefits of diversely-supervised search in the embedding space. When relying on a backbone model without fine-tuning, we obtain very low scores when trained either in supervised or self-supervised settings on ImageNet [Russakovsky et al., 2015]. For the self-supervised model, we rely on MoCo v2 [Chen et al., 2020, He et al., 2020]. When fine-tuning the model on the respective datasets, the diversely-supervised search improves considerably. Searching in the embedding space is the most effective as it captures all label similarities, and also the most efficient as the dimension is lower than the feature space. For example in Diverse-Cars, the dimensionality of the embedding space is 200 compared with 2048 in the feature space. When swapping backbones and searching in the feature space, we observe a similar behaviour as in Table 3.5 where the performance drops close to zero. Diversely-supervised search benefits from a retrieval operation in an embedding space that captures all label types.

Influence of diverse labels. We investigate the influence of each diverse label as a supervision source during training in Figure 3.6. In particular, we evaluate the effect of the instance, category and all attributes labels individually and their combination. When leveraging all types of labels, it achieves the best overall scores. In general, the instance labels always matters and combining two types of labels leads to an improvement. Though, both product datasets exhibit different behaviours. Figure 3.6a shows that the model benefits the most from category labels on Diverse–Cars. Category labels alone yield a high retrieval score and combining them with other types of labels results in even higher scores. Indeed,

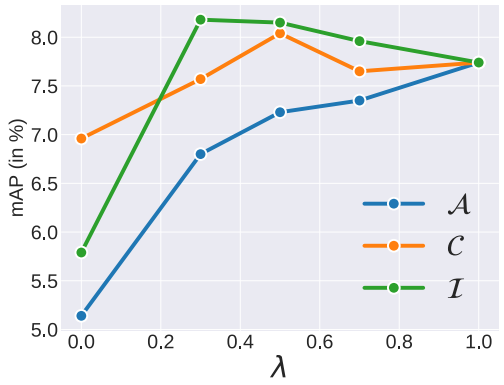


Figure 3.7: **Weighted diverse labels** effect on the diverse search of all queries (mAP) on Diverse–Clothes. Performance can be slightly improved by reducing the contribution of the attributes or the category labels. For simplicity, we set all contributions to one.

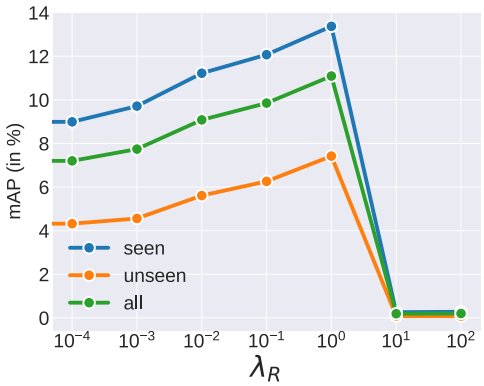


Figure 3.8: **Embedding regularization** on Diverse–Clothes. The stronger the regularization, the higher the performance on diverse search is for both seen and unseen queries (mAP). Though when the regularization is too strong, the model does not learn properly as everything is pushed to zero.

car makers usually distill a similar design to all their car models. Being able to represent categories is then the most important. Figure 3.6b rather depicts the importance of attribute-label supervision on Diverse–Clothes. Attribute labels alone yield a high retrieval score and their combination with other labels is always beneficial. Contrary to car makers, fashion designers focus more on attribute combinations to create new products. Indeed compared to cars, clothes have more attributes, which makes this supervision the most important. In case of scarce resources, we then recommend to collect annotations on instances and categories for cars, and on instances and attributes for clothes.

Weighted diverse labels. While Figure 3.6 switches on and off the contribution of every lambda, Figure 3.7 evaluates these trade-off hyper-parameters with real values. When evaluating every lambda individually, we fix the others to one. All settings improve over the absence of a label, which indicates that all labels are important to diversely-supervised search. It is possible to slightly improve the performance by reducing the contribution of attributes or category labels on Diverse–Clothes rather than setting them all to one. Though, as the search space for the lambda triplet is vast, we recommend to simply set all three to one. Notably, this enables a simple, non-exhaustive, and fair comparison with alternative methods.

Embedding regularization. Figure 3.8 varies the amount of regularization λ_R on the embedding space on Diverse–Clothes. The higher the regularization, the better the performance of diversely-supervised search is. Interesting, this affects

#A	seen	unseen	all
1	42.15	23.50	37.33
2	36.76	20.67	30.89
3	30.91	17.41	24.93

(a) **Diverse–Cars**

#A	seen	unseen	all
1	20.44	4.29	18.79
2	10.05	4.40	8.53
3	7.86	4.61	6.34

(b) **Diverse–Clothes**

Table 3.7: **Influence of the number of attributes**. We examine the influence of the number of attributes in composite queries and report the mAP (in %). The more specific the composite query is, the harder is gets to retrieve relevant images. Unseen queries for clothes remain at the same level because they are equally challenging as the median number of images per query is the same.

both seen and unseen queries positively. There is a cliff in performance after λ_R=1 where the performance drops drastically. Indeed, when the regularization is too strong, the representation is pushed towards zero, which annihilates the model learning.

Improving the performance. While comparisons in Table 3.1 are done with λ_R=0.001, Figure 3.8 shows that increasing this value can greatly benefit the diversely-supervised search in our proposed models. Indeed, when applying a λ_R=1 during training, we improve the mAP for all composite queries to 34.24 $\pm$ 0.23 for Diverse–Cars, and to 11.34 $\pm$ 0.21 for Diverse–Clothes. Though, applying such a high regularization for the alternatives can be detrimental. For example on Diverse–Cars, modulation drops to an mAP below one while conditional drops by five points. The fact that our model incorporates a grouping mechanism helps to benefit from higher regularization on the embedding space as alternatives without any grouping suffer to various extents.

Influence of the number of attributes. We examine the influence of the number of attributes in the composite queries on the retrieval performance. As described in Section 3.4.2, we create composite queries with up to three attributes. For example, "a *DS* with *3* doors and *5* seats" is a composite query with a category and two attributes, for a total of three labels. Table 3.7 shows that increasing the number of attributes in the composite queries leads to a more challenging task. The more specific the search is, the harder it gets to find the needle in the haystack. On both datasets there is a drop of about 12 mAP when switching from one to three attributes. In particular, Table 3.7a exhibits a drop of only 6 mAP points for unseen queries but 11 mAP points for seen queries on Diverse–Cars. Table 3.7b shows a constant performance for unseen queries while scores decrease more importantly for seen queries on Diverse–Clothes. This is explained by the fact that the median number of images per unseen composite query for all levels of detail is the same, making them equally challenging. Figure 3.9 depicts

Representation	Cars	Clothes
Binary	21.47	7.58
Real-valued	**32.19**	**7.74**

Table 3.8: **Binary representations** on the diverse search of all queries (mAP) on Diverse–Cars and Diverse–Clothes. While a binary representation has a large gap to real-valued representations on Diverse–Cars, it provides a compelling alternative with a close score on Diverse–Clothes.

Representation	Cars	Clothes
Sentences	5.51	3.96
Subspaces	**32.19**	**7.74**

Table 3.9: **Sentence representations** on the diverse search of all queries (mAP) on Diverse–Cars and Diverse–Clothes. Sentences cannot capture the diversity of all labels in composite queries, as they lack the flexibility of subspaces to represent every label.

composite query examples with an increasing number of attributes. For Diverse–Cars, there can exist multiple car models matching the query. The model can retrieve correct images regardless of the viewpoint. Yet, confusion can happen when cars are of the same color or shape. For Diverse–Clothes, the search is more challenging as there are usually one clothes item with very few images to retrieve. Items can be rare or exhibit original combinations of labels. Future work on product search should emphasize the retrieval performance of (a) composite queries with several attributes as distinguishing products on a fine-grained level requires a higher amount of attributes, and (b) unseen composite queries as designers usually create products with an unseen combination of labels.

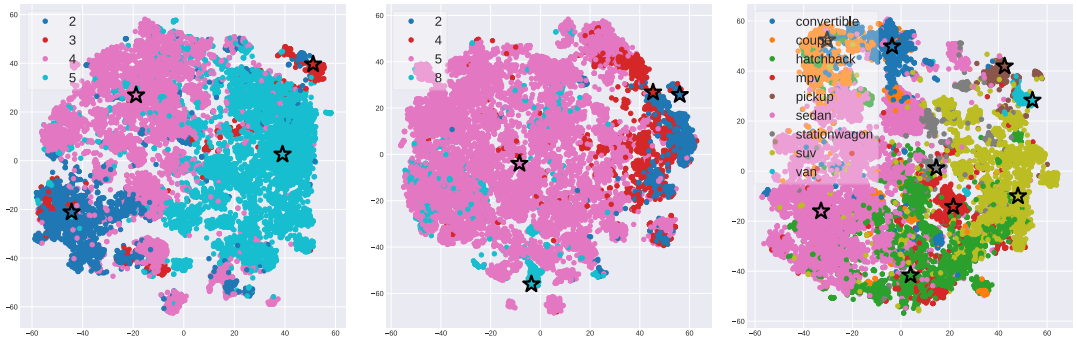
Attribute subspace visualization. Figure 3.10 plots the t-SNE [Maaten and Hinton, 2008] visualization of every attribute subspace on the test set of Diverse–Cars, as well as the latent prototype visualization for every attribute value. For the number of doors attribute, the prototypes are well separated, with a prototype at each extremity. Though, it appears that cars with 3 doors don't have a compact representation as they tend to spread all across the space. For the number of seats attribute, there is a transition from 2 seats to cars with more than 5 seats. This indicates that the model has found a progressive way to represent this attribute. For the type attribute, every car type is also represented around the region of its corresponding latent prototype. Some values are close to each other, for example coupe and convertible, which indicates that the model has captured the car shape similarities.

Binary representations. As we design our embedding model to be a probabilistic model, a binary representation can also be used for diversely-supervised search. In this scenario, the representation of every image corresponds to the one-hot predictions of the probabilistic model for the category label and every attribute label. As such, the composite query is represented by a binary representation for diversely-supervised search. Table 3.8 compares the binary rep-

(a) **Diverse–Cars** Our model can retrieve the multiple (third row) or only (second) matching car models. Yet, it can be fooled by cars of the same color (first).

(b) **Diverse–Clothes** Our model can retrieve rare (first row) or original (second and third) clothes items.

Figure 3.9: **Influence of the number of attributes.** We show examples of *unseen* composite queries with an increasing number of attribute values and their top-5 retrieved images (correct in *green*, incorrect in *red*).



(a) **Number of doors** (b) **Number of seats** (c) **Type**

Figure 3.10: **Attribute subspace visualization** with t-SNE on the test set of Diverse–Cars. Learned prototype representations for every value of every attribute are illustrated with a star. (a) The number of doors are clustered with prototypes being at extremities. Cars with 3 doors tend to spread all across the embedding space. (b) A transition from 2 seats to more than 5 seats is observed. Cars with more than 5 seats tend to spread all across the embedding space. (c) Car types are occupying the whole embedding space. Certain car types tend to be close to each other, *e.g.* *coupe* and *convertible*, or *pickup* and *van*.

resentations with real-valued representations. On Diverse–Cars, there is a large gap in performance between both representations. This difference resides in the fact that the performance for unseen queries drops by a factor two. On Diverse–Clothes, the performance is similar for both representations, which suggests that binary representations can be used if storage space becomes a challenge.

Text representations. An alternative to learned label subspaces is to rely on text representations. The idea is to process the diverse label through a language model to obtain a text representation. The model then learns to regress to the text embedding, which is considered as a prototype during learning. For example, for an image with a diverse label "a shirt with long sleeves and a stripe print", we feed this sentence to a language model and take the output embedding as a prototype to regress to. We rely on sentence-BERT [Reimers and Gurevych, 2019], a variant of BERT [Devlin et al., 2019] for sentences fine-tuned on natural language inference datasets [Bowman et al., 2015, Williams et al., 2018], to extract text representations. Table 3.9 shows that text representations underperform learned label subspaces. Recall the example above. While the image has a diverse label "a shirt with long sleeves and a stripe print", it should hit for composite queries such as "a shirt with long sleeves" or "a shirt with a stripe print". Having a sentence representation is too rigid as it imposes an order in the attributes and ties strongly attributes with the category. Instead, subspaces offer a more flexible representation, and allow composite queries with various numbers of attribute in an unordered manner.

(a) **Typical instances** (b) **Atypical instances** (c) **Eclectic instances**

Figure 3.11: **Discovering typical, atypical and eclectic products**. We explore product instances in the gallery set to discover design styles. Images in the same row share the same category label (underlined). The blue text box indicates the model prediction (*italics*). (a) Typical instances close to the category prototype depict the common appearance of sweaters, dresses and shirts. (b) Atypical instances far from the category prototype exhibit a global appearance that resembles other categories, which causes misclassification. (c) Eclectic instances with a high entropy display original attribute values for the category.

3.5.3 Discovering typical, atypical and eclectic products

In this experiment we aim to discover products with *typical*, *atypical* or *eclectic* styles in the gallery set. We rely on the *triple* grouping which integrates attribute, category and instance representations within the same embedding space. First, we aggregate visual representations per instance, *i.e.*images of the same instances are aggregated to the same visual representation. We refer to those as product representations. Second, we compute distances between product representations and all category prototypes $\mathbf{c}_y \in \mathbb{Z}_C$ in the embedding space. These distances provide three different indicators: (a) a small distance to the corresponding prototype indicates *typical* products, while (b) a large distance refers to *atypical* products. Additionally, the entropy can be computed over the probability distributions for each product representation, where (c) a high entropy refers to *eclectic* products on the edge of several categories. Probabilities are obtained by applying the softmax function over the distances.

We provide qualitative results based on the three indicators on Diverse–Clothes. Figure 3.11a illustrates the closest instances to category prototypes. These instances depict a common style, which makes them easily recognizable as they form typical instances [Rosch, 1978]. Distilling a typical design style in instances is particularly attractive for brands to enforce loyalty or attachment [van den Brink et al., 2006]. Yet, product design styles have a determined lifespan [Sproles, 1981] and other combinations of visual attributes defining the

style will emerge next due to cyclic [Al-Halah et al., 2017] or punctual [Mall et al., 2019] trends. Figure 3.11b illustrates the farthest instances to category prototypes. The global shape of these instances, either in size or fabric, makes them look like they are part of another category. For example, instances of "dresses" in row 2 look like "tees". Thus, the model can misclassify these instances. Figure 3.11c illustrates instances that confuse the embedding the most as they exhibit a high entropy. These instances depict an original visual appearance, especially for the *print* attribute. Searching for atypical and eclectic products reveals unexpected and intriguing trends in product design.

3.6 Conclusion

We have introduced the problem of diversely-supervised visual product search, where queries describe a specific set of diverse labels to search for. We have proposed a diversely-supervised embedding, where attribute, instance and attribute labels provide a diverse supervision to learn a representation for products. Evaluation relies on composite queries to describe the specific set of labels to search for. Composite query representations correspond to a per-label average of selected visual representations in the embedding space. Experiments on seen and unseen settings show that our diversely-supervised embedding better models a diverse set of labels than alternative baselines repurposed for diversely-supervised visual product search. The embedding also enables the discovery of the typicality effect in design styles, which reveals intriguing products. In the current form, labels describe physical properties of products but could also capture aesthetics, or cultural differences.

Chapter 4

Bias-Awareness for Zero-Shot Learning the Seen and Unseen

4.1 Introduction

Zero-shot recognition [Lampert et al., 2014, Palatucci et al., 2009] considers if models trained on a given set of seen classes $\mathcal{S}$ can extrapolate to a distinct set of unseen classes $\mathcal{U}$. In generalized zero-shot learning [Chao et al., 2016, Xian et al., 2018a], we also want to remember the seen classes and evaluate over the union of the two sets of classes $\mathcal{T} = \mathcal{S} \cup \mathcal{U}$. Nevertheless, when evaluating existing models in the generalized scenario, the seminal work of Chao et al. [2016] highlights that predictions tend to be biased towards the seen classes observed during training. In this chapter, we consider the challenge of mitigating this inherent bias present in classifiers by proposing a bias-aware model.

An effective remedy to remove the bias towards seen classes is to calibrate their predictions during inference. Chao et al. [2016] propose to reduce the scores for the seen classes, which in return improves the generalized zero-shot learning performance. Yet, the bias towards seen classes should also be tackled while training classifiers and not only during the evaluation phase to address the bias from the start. Towards this goal, seen and unseen classes can be addressed separately during training. Liu et al. [2018] define two separate training objectives to calibrate the confidence of seen classes and the uncertainty of unseen classes. Atzmon and Chechik [2019] break the classification into two separate experts, with one model for seen classes and another one for unseen classes. Their COSMO approach provides compelling results at the expense of a third additional expert to combine results. As generalized zero-shot learning considers both seen and

Published at *British Machine Vision Conference* [Thong and Snoek, 2020]

unseen classes simultaneously, learners should benefit from mitigating the bias in both directions by considering both sets jointly rather than separately.

The main objective of this chapter is to mitigate the bias towards seen classes by considering predictions of seen and unseen classes simultaneously during training. To achieve this, we propose a simple bias-aware learner that maps inputs to a semantic embedding space where class prototypes are formed by real-valued representations. We address the bias by introducing (*i*) a calibration for the learner with temperature scaling, and (*ii*) a margin-based bidirectional entropy term to regularize seen and unseen probabilities jointly. We show that the bias towards seen classes is also dataset-dependent, and every dataset does not suffer to the same extent. Finally, we illustrate the versatility of our approach. By relying on a real-valued embedding space, the model can (*i*) handle different types of prototype representation for both seen and unseen classes, and (*ii*) operate either on real features, akin to compatibility functions, or leverage generated unseen features. Comparisons on four datasets for generalized zero-shot learning show the effectiveness of bias-awareness. All source code and setups are released[1].

4.2 Related work

Generalized zero-shot learning has been introduced to provide a more realistic and practical setting than zero-shot learning, as models are evaluated on both seen and unseen classes [Chao et al., 2016]. This change in evaluation has a large impact as existing *compatibility functions* designed for zero-shot learning do not perform well in the generalized setting [Changpinyo et al., 2020, Chao et al., 2016, Xian et al., 2018a]. Indeed, whether they are based on a ranking loss [Akata et al., 2015, 2016, Frome et al., 2013, Romera-Paredes and Torr, 2015, Xian et al., 2016] or synthesis [Changpinyo et al., 2016, 2017, 2020], compatibility functions empirically exhibit a very low accuracy for unseen classes. As identified by Chao et al. [2016], this indicates a strong inherent bias in all classifiers towards the seen classes. To overcome the low accuracy for unseen classes, both Kumar Verma et al. [2018] and Xian et al. [2018b] learn a conditional *generative model* to generate image features. Once trained, image features of unseen classes are sampled by changing the conditioning. Classification then consists of training a one-hot softmax classifier on both real and sampled image features. Having access during training to generated unseen features leads to an increase in unseen class accuracy. Among the different generative models used in generalized zero-shot learning, are generative adversarial networks [Felix et al., 2018, Li et al., 2019, Xian et al., 2018b], variational autoencoders [Kumar Verma et al., 2018, Schönfeld et al., 2019] or a combination of both [Xian et al., 2019]. However, a classifier trained on generated features still suffers from a bias towards seen classes because generative models do not produce fully realistic features. In this

[1]Source code is available at https://github.com/twuilliam/bias-gzsl

chapter, we strive for a bias-aware classifier which can behave as a stand-alone model like *compatibility functions* and also leverage unseen features samples from a *generative model*.

Addressing the bias in classifiers remains an open challenge for generalized zero-shot learning. Although Chao et al. [2016] identify the critical bias towards seen classes, only a few works try to address it during training. Related works separate the seen and unseen classifications. Liu et al. [2018] map both features and semantic representations to a common embedding space. Probabilities are then calibrated separately in this common space to make seen class probabilities confident and reduce the uncertainty of unseen class probabilities. Atzmon and Chechik [2019] train expert models separately for seen and unseen class predictions. Their predictions are further combined in a soft manner with a third expert to produce the final decision. In this chapter, we strive to address the bias by considering seen and unseen class probabilities *jointly* rather than *separately*. Having access during training to the joint class probabilities lets the bias-aware model learn how to balance them from the start.

4.3 Method

During training, a generalized zero-shot learner $G : X \rightarrow \mathcal{T}$ is given a training set $\mathcal{D}^{\mathcal{S}} = \{(x_n, y_n), y_n \in \mathcal{S}\}_{n=1}^{N}$, where $x_n \in \mathbb{R}^D$ is an image feature of dimension D and y_n comes from the set $\mathcal{S}$ of seen classes, with $\mathcal{S} \subset \mathcal{T}$. For each $c \in \mathcal{S}$ there exists a corresponding semantic class representation $\phi(c) \in \mathbb{R}^A$ of dimension A. At testing time, G predicts for each sample in the testing set $\mathcal{D}^{\mathcal{T}} = \{x_n\}_{n=1}^{M}$ a label that belongs to $\mathcal{T}$ by exploiting the joint set of seen and unseen semantic class representations. This problem formulation can be extended with an auxiliary dataset $\widetilde{\mathcal{D}}^{\mathcal{U}} = \{(\widetilde{x}_n, y_n), y_n \in \mathcal{U}\}_{n=1}^{\widetilde{N}}$, where y_n comes from the set of unseen classes $\mathcal{U}$. $\widetilde{\mathcal{D}}^{\mathcal{U}}$ mimics image features from unseen classes, and is typically sampled from a generative model. The joint set $\{\mathcal{D}^{\mathcal{S}}, \widetilde{\mathcal{D}}^{\mathcal{U}}\}$ covers both seen and unseen classes.

In this chapter, we propose a bias-aware generalized zero-shot learner $f(\cdot)$, which can operate during training with (i) only $\mathcal{D}^{\mathcal{S}}$ similar to compatibility functions (Section 4.3.1) or (ii) the joint set $\{\mathcal{D}^{\mathcal{S}}, \widetilde{\mathcal{D}}^{\mathcal{U}}\}$ similar to classifiers in the generative approach (Section 4.3.2). In both scenarios, the learner includes mechanisms to mitigate the bias towards seen classes. Learning consists of mapping inputs x to their corresponding semantic class representations $\phi(c)$. In other words, the model regresses to a real-valued vector, which describes a class prototype. We denote the set of seen class prototypes as $\Phi^{\mathcal{S}} = \{\phi(c), c \in \mathcal{S}\}$, unseen class prototypes as $\Phi^{\mathcal{U}} = \{\phi(c), c \in \mathcal{U}\}$, and their union as $\Phi^{\mathcal{T}} = \Phi^{\mathcal{S}} \cup \Phi^{\mathcal{U}} = \{\phi(c), c \in \mathcal{T}\}$. Usually, the semantic knowledge used for class prototypes corresponds to semantic attributes [Farhadi et al., 2009, Lampert et al., 2014], word vectors of the class name [Frome et al., 2013, Palatucci et al., 2009], hierarchical representations [Akata et al., 2015, 2016, Xian et al., 2016], or sentence descriptions [Reed

et al., 2016, Xian et al., 2018b]. To exploit this diversity in semantic knowledge, we propose to swap the representation types for seen and unseen prototypes (Section 4.3.3).

4.3.1 Stand-alone classification with seen classes only

We design the bias-aware generalized zero-shot learner as a probabilistic model with two key principles. First, it is calibrated towards seen classes such that inputs from unseen classes yield a low confidence prediction at testing time. In return, this reduces the bias towards seen classes for unseen class inputs. Second, it maps inputs to class prototypes in the semantic embedding space. Following these two principles, we propose:

$$p(c|x, \mathcal{S}) = \exp\left(\frac{s\left(f(x), \phi(c)\right)}{T}\right) \Big/ \sum_{c' \in \mathcal{S}} \exp\left(\frac{s\left(f(x), \phi(c')\right)}{T}\right) \tag{4.1}$$

where $s(\cdot, \cdot)$ is the cosine similarity and $T \in \mathbb{R}_{>0}$ is the temperature scale. When $T = 1$, it acts as the normal softmax function. When $T > 1$, probabilities are spreading out. When $T < 1$, probabilities tend to concentrate similar to a Dirac delta function. Contrary to knowledge distillation [Hinton et al., 2014], we seek to concentrate the probabilities with a low temperature scale for discriminative purposes. Learning the probabilistic model is done via minimizing the cross-entropy loss function over the training set of seen examples $\mathcal{D}^{\mathcal{S}}$:

$$\mathcal{L}_{\text{s}} = -\frac{1}{N} \sum_{n=1}^{N} \log p(y_n|x_n, \mathcal{S}). \tag{4.2}$$

This probabilistic model behaves like a compatibility function, because it only sees samples from seen classes at training time. At testing time, the evaluation simply measure the similarity in the embedding space with respect to the union of seen and unseen prototypes $\Phi^{\mathcal{T}}$.

Variants of this prototype-based learner have been proposed in image retrieval [Liu et al., 2017b, Movshovitz-Attias et al., 2017, Wen et al., 2016, Zhai and Wu, 2019] or image classification [Liu et al., 2018, Snell et al., 2017, Wu et al., 2018]. We mainly differ by (*i*) fixing the prototypes to be semantic class representations rather than learning them; (*ii*) learning a mapping from the inputs to the class representations rather than learning a common embedding space; (*iii*) applying a softmax function to provide a probabilistic interpretation of cosine similarities; and (*iv*) calibrating the model with the same temperature scaling for both training and testing.

4.3.2 Classification with both seen and unseen classes

In the generative approach for generalized zero-shot learning, samples from unseen classes are generated. We can then use the generated data $\widetilde{\mathcal{D}}^{\mathcal{U}}$ as an auxiliary

dataset for *calibration* and for *entropy regularization*. In this context, given an input x the probabilistic model learns to predict a class from the union of both seen and unseen classes:

$$p(c|x,\mathcal{T}) = \exp\left(\frac{s\left(f(x),\phi(c)\right)}{T}\right) \Big/ \sum_{c'\in\mathcal{T}} \exp\left(\frac{s\left(f(x),\phi(c')\right)}{T}\right). \tag{4.3}$$

The only and major difference with eq. 4.1 resides in the class prototypes that are considered to produce the prediction while $f(\cdot)$ remains the same model. $p(c|x,\mathcal{S})$ only evaluates over the set of seen class prototypes $\Phi^{\mathcal{S}}$ while $p(c|x,\mathcal{T})$ evaluates over the union of seen and unseen class prototypes $\Phi^{\mathcal{T}}$. In this case, the temperature scaling ensures that the model is confident for both seen and unseen classes. This difference also makes the learning distinctive from related works (*i.e.*, DCN [Liu et al., 2018] or COSMO [Atzmon and Chechik, 2019]) as they consider seen and unseen classifications separately rather than jointly. Akin to eq. 4.2, we minimize the cross-entropy loss function on the joint set $\{\mathcal{D}^{\mathcal{S}}, \widetilde{\mathcal{D}}^{\mathcal{U}}\}$ of seen and unseen classes:

$$\mathcal{L}_{\text{s+u}} = -\frac{1}{N}\sum_{n=1}^{N} \log p(y_n|x_n,\mathcal{T}) - \frac{1}{\widetilde{N}}\sum_{n=1}^{\widetilde{N}} \log p(y_n|\widetilde{x}_n,\mathcal{T}). \tag{4.4}$$

This probabilistic model behaves like a classifier used in generative approaches, because it sees samples from both seen and unseen classes at both training and testing times, and the partition function normalizes over the union of seen and unseen sets of classes. Having a classification over the union enables regularization in both seen and unseen directions.

Bidirectional entropy regularization. Intuitively, when an image from an *unseen* class is fed to the classifier, probabilities for *seen* classes should yield a high entropy while probabilities for *unseen* classes should result in a low entropy. In other words, the evaluation over *seen* classes of an *unseen* class input should be uncertain because the image comes from a class the classifier has never encountered during training. Conversely, when an image from a *seen* class is fed to the classifier, the entropy of the probabilities for *unseen* classes should be high while the entropy for *seen* classes should be low. To encourage this effect, given an image x we compute the normalized Shannon entropy [Shannon, 1948] of the probabilistic model $p(c|x,\mathcal{T})$ for both seen and unseen class directions:

$$\mathcal{H}_{\text{s}}(x) = \frac{-1}{|\mathcal{S}|}\sum_{c\in\mathcal{S}} p(c|x,\mathcal{T}) \log p(c|x,\mathcal{T}), \tag{4.5}$$

$$\mathcal{H}_{\text{u}}(x) = \frac{-1}{|\mathcal{U}|}\sum_{c\in\mathcal{U}} p(c|x,\mathcal{T}) \log p(c|x,\mathcal{T}), \tag{4.6}$$

where $\mathcal{H}_{\text{s}}$ and $\mathcal{H}_{\text{u}}$ are the average entropy for seen and unseen classes, respectively. For training, we derive a margin-based regularization for both seen and unseen class directions:

$$R_{\text{s}} = \left[m + \frac{1}{N} \sum_{n=1}^{N} \mathcal{H}_{\text{s}}(x_n) - \frac{1}{\widetilde{N}} \sum_{n=1}^{\widetilde{N}} \mathcal{H}_{\text{s}}(\widetilde{x}_n) \right]_+, \tag{4.7}$$

$$R_{\text{u}} = \left[m + \frac{1}{\widetilde{N}} \sum_{n=1}^{\widetilde{N}} \mathcal{H}_{\text{u}}(\widetilde{x}_n) - \frac{1}{N} \sum_{n=1}^{N} \mathcal{H}_{\text{u}}(x_n) \right]_+, \tag{4.8}$$

where $[\cdot]_+ = \max(0, \cdot)$. R_{s} ensures a margin of at least m between the average seen class entropy of seen inputs x_n and generated unseen inputs $\widetilde{x}_n$. In other words, this formulation seeks to minimize $\mathcal{H}_{\text{s}}(x_n)$ and maximize $\mathcal{H}_{\text{s}}(\widetilde{x}_n)$. R_{u} has a corresponding effect on the unseen class entropy. The final loss function for training then becomes:

$$\mathcal{L}_{\text{f}} = \mathcal{L}_{\text{s+u}} + \lambda_{\text{Ent}}(R_{\text{s}} + R_{\text{u}}). \tag{4.9}$$

where $\lambda_{\text{Ent}} \in \mathbb{R}_{\geq 0}$ is a hyper-parameter to control the contribution of the bidirectional entropy.

4.3.3 Swapping seen and unseen class representations

As presented above, relying on a real-valued embedding space allows mechanisms to mitigate the bias in two scenarios. It also enables to swap class representations to less biased representations. Consider now the case where there exist multiple types of semantic information which differ by their type of representation and by how expensive it is to collect them. For example, attribute descriptions require expert knowledge while sentence descriptions can be crowd-sourced to non-expert workers. Practically, sentences tend to be less biased than attributes and perform better [Xian et al., 2018b], but do not offer a comprehensive expert-based explanation [Reed et al., 2016]. One could then train a model for seen classes on attributes as they rely on expert-based explanations and rely for unseen classes on sentences as they are easier to collect. This results in different representation types for seen and unseen classes.

Formally, we assume that we have access to seen prototypes $\{\Phi_A^{\mathcal{S}}, \Phi_B^{\mathcal{S}}\}$ with representations from domain A and B, respectively. For evaluation, we have access to unseen prototypes $\Phi_A^{\mathcal{U}}$ of domain A but $\Phi_B^{\mathcal{U}}$ of domain B is absent. The objective is then to learn a mapping β from $\Phi_A^{\mathcal{S}}$ to $\Phi_B^{\mathcal{S}}$, in order to regress $\hat{\Phi}_B^{\mathcal{U}}$ from $\Phi_A^{\mathcal{U}}$ at testing time. We define the mapping as a linear least squares regression problem with Tikhonov regularization, which corresponds to:

$$\min_{\beta} \|\Phi_B^{\mathcal{S}} - \beta \Phi_A^{\mathcal{S}}\|_2 + \lambda_\beta \|\beta\|_2. \tag{4.10}$$

where λ_β controls the amount of regularization. Relying on a linear transformation prevents overfitting, as the mapping involves a limited set of class prototypes. During evaluation, we apply β to unseen prototypes of domain A to regress their values in domain B: $\hat{\Phi}_B^{\mathcal{U}} = \beta \Phi_A^{\mathcal{U}}$. Swapping representations then corresponds to regressing from one domain to another.

4.4 Experimental details

Datasets. We report experiments on four datasets commonly used in generalized zero-shot learning, *e.g.*, [Changpinyo et al., 2020, Chao et al., 2016, Reed et al., 2016, Xian et al., 2018a]. For all datasets, we rely on the train and test splits proposed by Xian et al. [2018a]:

- *Caltech-UCSD-Birds 200-2011 (CUB)* [Wah et al., 2011] contains 11,788 images from 200 bird species. Every species is described by a unique combination of 312 semantic attributes to characterize the color, pattern and shape of their specific parts. Moreover, every bird image comes along with 10 sentences describing the most prominent characteristics [Reed et al., 2016]. 150 species are used as seen classes during training, and 50 distinct species are left out as unseen classes during testing.

- *SUN Attribute (SUN)* [Patterson and Hays, 2012] contains 14,340 images from 717 scene types. Every scene is also described by a unique combination of 102 semantic attributes to characterize material and surface properties. 645 scene types are used as seen classes during training, and 72 distinct scene types are left as unseen classes during testing.

- *Animals with Attributes (AWA)* [Lampert et al., 2014] contains 30,475 images from 50 animals. Every animal comes with a unique combination of 85 semantic attributes to describe their color, shape, state or function. 40 animals are used as seen classes during training, and 10 distinct animals are left out as unseen classes during testing.

- *Oxford Flowers (FLO)* [Nilsback and Zisserman, 2008] contains 8,189 images from 102 flower plants. Every flower plant image is described by 10 different sentences describing the shape and appearance [Reed et al., 2016]. 82 flowers are used as seen classes during training, and 20 distinct flowers are left as unseen classes during testing.

Features extraction. For all datasets, we rely on the features extracted by Xian et al. [2018a]. Image features x come from ResNet101 [He et al., 2016] trained on ImageNet [Russakovsky et al., 2015] and sentences representations are extracted from a 1024-dimensional CNN-RNN [Reed et al., 2016]. As established by Xian et al. [2018a], parameters of ResNet101 and the CNN-RNN are frozen and are not fine-tuned during the training phase. No data augmentation is performed either.

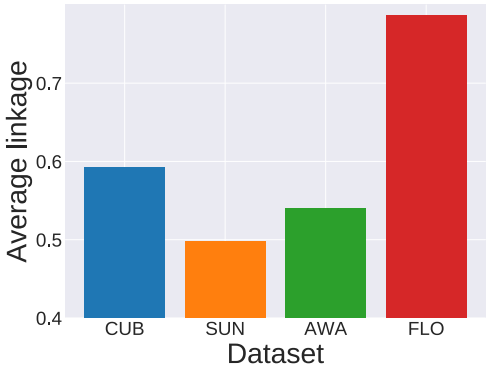


Figure 4.1: **Bias variation** across datasets. When measuring the average linkage between seen and unseen representations, FLO is the most affected while SUN is the least. Thus, the bias towards seen classes differs across datasets.

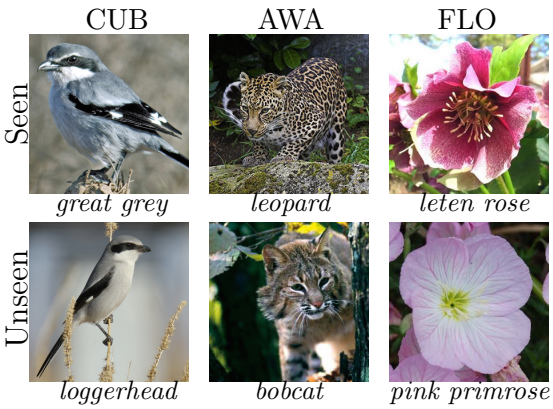


Figure 4.2: **Seen and unseen class samples**. Visual differences arise from the global shape (CUB, AWA) or colors (FLO). Yet, their semantic class representation yields a very high pairwise similarity, which creates a high bias.

Evaluation. We evaluate experiments with calibration stacking as proposed by Chao et al. [2016], which penalizes the seen class probabilities to reduce the bias during evaluation. Following Xian et al. [2018a], we compute the average per-class top-1 accuracy of seen (denoted as $\mathbf{s}$) and unseen (denoted as $\mathbf{u}$) classes, and their harmonic mean $\mathbf{H}=(2 \times \mathbf{s} \times \mathbf{u})/(\mathbf{s}+\mathbf{u})$. We report the 3-run average.

Implementation details. In our model, $f(\cdot)$ corresponds to a multilayer perceptron with 2 hidden layers of size 2048 and 1024 to map the features x to the joint visual-semantic embedding space of size A. The output layer has a linear activation, while hidden layers have a ReLU activation [Nair and Hinton, 2010] followed by a Dropout regularization ($p = 0.5$) [Srivastava et al., 2014]. We train $f(\cdot)$ using stochastic gradient descent with Nesterov momentum [Sutskever et al., 2013]. We set the following hyper-parameters for all datasets: learning rate of 0.01 with cosine annealing [Loshchilov and Hutter, 2017], initial momentum of 0.9, batch size of 64, temperature of 0.05, and an entropy regularization term of 0.1 with a margin of 0.2. For AWA, we reduce the learning rate to 0.0001 and increase the entropy regularization to 0.5 while keeping the same margin. When relying on sentence representations, we double the capacity of $f(\cdot)$ with twice the number of hidden units in each layer. We set hyper-parameters on a hold-out validation set and re-train on the joint training and validation sets. The source code uses the Pytorch framework [Paszke et al., 2019].

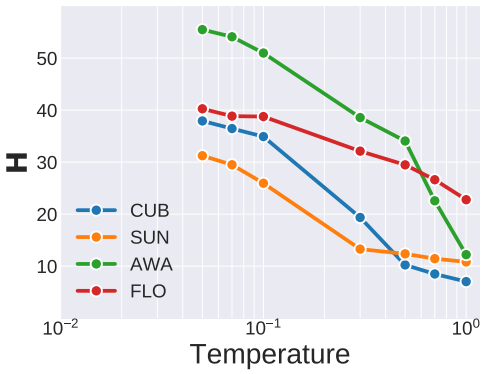


Figure 4.3: **Temperature scaling** ablation from $T = 0.05$ to $T = 1$. Temperature values over 0.1 degrade the performance because probabilities start to spread which makes the model less confident.

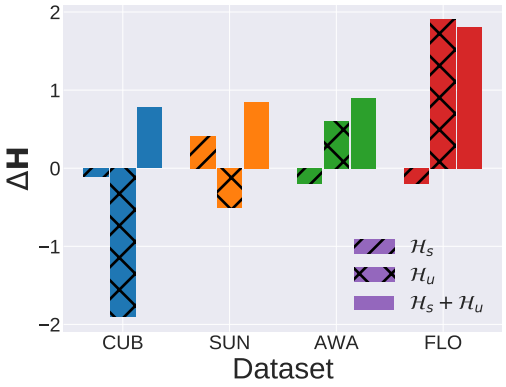


Figure 4.4: **Entropy regularization** in one ($\mathcal{H}_s$ or $\mathcal{H}_u$, hatched) and two ($\mathcal{H}_s + \mathcal{H}_u$, not hatched) directions compared with models without. Regularizing in only one direction can result in a negatively effect. Including both directions consistently improves results by creating a better bias trade-off.

4.5 Results

Bias variation. To understand whether the bias towards seen classes is dataset-dependent, we measure the average linkage between seen and unseen representations. Concretely, we compute the average of the pairwise cosine similarity between $\Phi^{\mathcal{S}}$ and $\Phi^{\mathcal{U}}$. A high average linkage then refers to a high similarity between seen and unseen representations. Intuitively, a high average linkage is not desirable as unseen representations can easily be confused with seen ones, which makes the generalized zero-shot learning problem harder. Figure 4.1 depicts the average linkage per dataset. FLO exhibits the highest average linkage while SUN the lowest, with a 1.6 times difference. In other words, classifiers trained on FLO are highly affected by the bias towards seen classes. Figure 4.2 illustrates seen and unseen class samples with a very high pairwise similarity on CUB, AWA and FLO. Visually, these classes can be differentiated by their color or shape. Though, their semantic representations are very similar, which creates a high bias. The bias towards seen classes then differs across datasets. Addressing the bias within generalized zero-shot learners should then result in varying extents.

Temperature scaling. Figure 4.3 varies the scale of temperature in eq. 4.1. Following related metric learning works (*e.g.*, [Wu et al., 2018, Zhai and Wu, 2019]), we consider the temperature as a hyper-parameter. When treated as a latent parameter, the optimization diverges as its value goes down to zero to satisfy the loss function. The highest **H** score occurs when $T = 0.05$ on the validation

Seen	Unseen	**H**
`Att`	`Att`	48.5
`Sen`	`Att`	47.4
`Att`	`Sen`	49.7
`Sen`	`Sen`	50.3

Table 4.1: **Swapping attribute (`Att`) and sentence (`Sen`) representations**. While `Att-Att` and `Sen-Sen` are the usual non-swapped evaluation settings, our method can also swap them. When using sentences for unseen classes, it always improves upon attributes in swapped and non-swapped evaluations as they are less biased and more discriminative.

set of all datasets. Performance starts to degrade substantially after $T > 0.1$. A temperature lower than $T < 0.05$ can yield even higher scores but is usually prone to numerical errors. As such, we set $T = 0.05$ in all our experiments when training the model with only seen samples (eq. 4.2) or in combination with generated unseen samples (eq. 4.4). We also evaluate modifying T between training and testing phases. Setting it to 1 during training and testing as in a normal softmax drops **H** by 43.3% on AWA. Changing it to 0.05 when testing drops the score by 25.6%. A fixed temperature value is preferred to ensure $f(\cdot)$ maps inputs to prototypes similarly in training and testing. Having a low temperature yields narrow probabilities, which translates into a more confident and discriminative model. Hence, the model reduces the bias by having a lower likelihood to classify an unseen class input as part of a seen class.

Entropy regularization. Figure 4.4 ablates the direction of the margin-based entropy term in eq. 4.9. For this experiment, we rely on unseen class features generated from Cycle-CLSWGAN [Felix et al., 2018]. When using a unidirectional entropy regularization, the improvement is either very low or even negative over a model without any regularization. Interestingly, this negative effect does not depend on the direction as both $\mathcal{H}_s$ and $\mathcal{H}_u$ are affected when considered individually. Regularizing in only one direction forces the model to compensate for the other direction. Only the bidirectional regularization provides a benefit for all datasets consistently. This positive effect indicates the importance of balancing out both seen and unseen probabilities when mitigating the bias. Regularizing in both directions jointly helps the model learn a correct bias trade-off.

Swapping representations. Table 4.1 presents the different combinations of attribute (`Att`) and sentence (`Sen`) representations for training and evaluation. `Att-Att` and `Sen-Sen` are the common non-swapped settings. `Sen-Sen` forms an upper-bound as sentences provide better class representations over attributes. Indeed, sentence descriptions exhibit a lower average linkage than attribute descriptions. In a swapped setting, the unseen representations are regressed from representations in another domain based on eq. 4.10. A model trained on `Att` can be improved by 1.2 points at testing time when using `Sen` to regress the unseen representations. On the other hand, a model trained on `Sen` degrades when using `Att` to regress unseen representations. Indeed, `Sen-Att` requires to map

low-dimensional attribute representations of unseen classes to a high-dimensional space of sentence representations on which the classifier has been trained. `Sen-Att` then involves dimensionality expansion, which is a harder problem than dimensionality compression in `Att-Sen`. In the scenario where a model is trained on attributes for seen class derived from experts, it is possible to leverage sentences for unseen classes derived from crowd-sourcing to improve the results.

Comparison with the state of the art. We compare the bias-aware prototype learner with eight other classifiers. Scores from other classifiers correspond to the performance as reported by the authors in their original paper. First, we consider stand-alone classifiers which only observe the seen class inputs during training, *i.e.*, without using any generated features. Compared with the one-hot softmax [Xian et al., 2018b] and COSMO [Atzmon and Chechik, 2019], our proposal can operate as a stand-alone classifier akin to a compatibility functions [Akata et al., 2015, 2016, Frome et al., 2013, Liu et al., 2018, Romera-Paredes and Torr, 2015, Xian et al., 2016]. Indeed, our formulation relies on a real-valued embedding space rather than a discrete label space for classification. We outperform all other stand-alone classifiers on all datasets. Second, our approach is easily extended with existing generative models to include an auxiliary dataset $\widetilde{\mathcal{D}}^{\mathcal{U}}$ for unseen classes. We select f-CLSWGAN [Xian et al., 2018b] and Cycle-CLSWGAN [Felix et al., 2018] as the authors provide source code to evaluate on all four datasets. Reproducing their experiments yields results within a reasonable range, *i.e.*, less than a 2-point difference in the **H** metric. We obtain better results with Cycle-CLSWGAN [Felix et al., 2018] than f-CLSWGAN [Xian et al., 2018b], which highlights the importance of the quality of the generated unseen class features. Moreover, our method profits more when generate samples better reflect the true distribution. When switching from f-CLSWGAN [Xian et al., 2018b] to cycle-CLSWGAN [Felix et al., 2018] on CUB, a one-hot softmax classifier leads to a 2.6% increase while our bias-aware classifier with a joint entropy regularization yields a 7.5% increase. We achieve state-of-the-art results on CUB, AWA and FLO. Only on the SUN dataset the one-hot softmax [Xian et al., 2018b] and COSMO [Atzmon and Chechik, 2019] provide higher scores. This originates from a lower bias towards seen classes in the SUN dataset (see Figure 4.1), which makes a bias-aware model less beneficial. When a dataset exhibits a low bias, separating the model for seen and unseen classes is preferred to treat them equally. Conversely, when a dataset exhibits a high bias, the training of the model should consider seen and unseen classes jointly to balance out their probabilities from the start. Overall, we produce competitive results in both scenarios, especially compared with classifiers without any bias-awareness.

Method	CUB			SUN			AWA			FLO		
	u	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**
DeViSE [Frome et al., 2013]	23.8	53.0	32.8	16.9	27.4	20.9	13.4	68.7	22.4	9.9	44.2	16.2
`w/ f-CLSWGAN` [Xian et al., 2018b]	52.2	42.4	46.7	38.4	25.4	30.6	35.0	62.8	45.0	45.0	38.6	41.6
SJE [Akata et al., 2015]	23.5	59.2	33.6	14.7	30.5	19.8	11.3	74.6	19.6	13.9	47.6	21.5
`w/ f-CLSWGAN` [Xian et al., 2018b]	48.1	37.4	42.1	36.7	25.0	29.7	37.9	70.1	49.2	52.1	56.2	54.1
LATEM [Xian et al., 2016]	15.2	57.3	24.0	14.7	28.8	19.5	7.3	71.7	13.3	6.6	47.6	11.5
`w/ f-CLSWGAN` [Xian et al., 2018b]	53.6	39.2	45.3	42.4	23.1	29.9	33.0	61.5	43.0	47.2	37.7	41.9
ESZSL [Romera-Paredes and Torr, 2015]	12.6	63.8	21.0	11.0	27.9	15.8	6.6	75.6	12.1	11.4	56.8	19.0
`w/ f-CLSWGAN` [Xian et al., 2018b]	36.8	50.9	43.2	27.8	20.4	23.5	31.1	72.8	43.6	25.3	69.2	37.1
ALE [Akata et al., 2016]	23.7	62.8	34.4	21.8	33.1	26.3	16.8	76.1	27.5	13.3	61.6	21.9
`w/ f-CLSWGAN` [Xian et al., 2018b]	40.2	59.3	47.9	41.3	31.1	35.5	47.6	57.2	52.0	54.3	60.3	57.1
DCN [Liu et al., 2018]	28.4	60.7	38.7	25.5	37.0	30.2	25.5	84.2	39.1	–	–	–

Table 4.2: **Comparison with the state of the art** (part 1)

Method	CUB			SUN			AWA			FLO		
	u	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**	**u**	**s**	**H**
One-hot softmax [Xian et al., 2018b]	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
`w/ f-CLSWGAN` [Xian et al., 2018b]	43.7	57.7	49.7	42.6	36.6	39.4	57.9	61.4	59.6	59.0	73.8	65.6
`w/ Cycle-CLSWGAN` [Felix et al., 2018]†	45.7	61.0	52.3	49.4	33.6	40.0	56.9	64.0	60.2	72.5	59.2	65.1
`w/ CADA-VAE` [Schönfeld et al., 2019]	51.6	53.5	52.4	47.2	35.7	40.6	57.3	72.8	64.1	–	–	–
`w/ f-VAEGAN-D2` [Xian et al., 2019]†	48.4	60.1	53.6	45.1	38.0	**41.3**	57.6	70.6	63.5	56.8	74.9	64.6
`w/ LisGAN` [Li et al., 2019]	46.5	57.9	51.6	42.9	37.8	40.2	52.6	76.3	62.3	57.7	83.9	68.3
COSMO [Atzmon and Chechik, 2019]	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a	n/a
`w/ f-CLSWGAN` [Xian et al., 2018b]	60.5	41.0	48.9	35.3	40.2	37.6	64.8	51.7	57.5	59.6	81.4	68.8
`w/ LAGO` [Atzmon and Chechik, 2018]	44.4	57.8	50.2	44.9	37.7	41.0	52.8	80.0	63.6	n/a	n/a	n/a
This work	45.1	52.5	48.5	41.0	30.1	34.7	55.2	70.5	61.9	42.6	66.6	52.0
`w/ f-CLSWGAN` [Xian et al., 2018b]	50.7	49.9	50.3	41.1	31.6	35.7	57.7	68.4	62.5	53.8	76.0	63.0
`w/ Cycle-CLSWGAN` [Felix et al., 2018]†	57.4	58.2	**57.8**	44.8	32.7	37.8	61.3	69.2	**65.0**	69.3	79.9	**74.2**

† Method relies on sentence representations instead of attribute representations for CUB.

Table 4.2: **Comparison with the state of the art** (part 2, continued), where classifiers are delimited by a horizontal rule and their combination with a generative model is in `teletype` font. "n/a" denotes a non-applicable setting to the method while "–" refers to non-reported results in the original paper. Compared with one-hot softmax and COSMO, our proposal is a stand-alone method that can also operate with seen class samples only. Compared with the other compatibility functions that also operate in this setting, it achieves the best results in this setting (underlined). When extended with generated unseen class samples, we also improve over other classifiers (**bold**), leading to state-of-the-art results on the three most biased datasets out of four (see Figure 4.1).

4.6 Conclusion

The classification of seen and unseen classes in generalized zero-shot learning requires models to be aware of the bias towards seen classes. In this chapter, we present such a model which calibrates the probabilities of seen and unseen classes jointly during training, and ensures a margin between the average entropy of both seen and unseen class probabilities. Learning consists of regressing inputs to real-valued representations. Relying on a mapping to a real-valued embedding space enables to swap seen and unseen representations, and to evaluate the model in a stand-alone scenario or in combination with generated unseen features. Overall, our proposed bias-aware learner provides an effective alternative to separate classification approaches or classifiers without bias-awareness.

Chapter 5

Feature and Label Embedding Spaces Matter in Addressing Image Classifier Bias

5.1 Introduction

This chapter strives to identify and mitigate biases present in image classifiers, with a focus on their feature and label embedding spaces. Adverse decisions from image classifiers can create discrimination against members of a certain class of protected attribute, such as age, gender, or skin tone. Buolamwini and Gebru [2018] importantly show that face recognition systems misclassify subgroups with darker skin tones. This also applies to object recognition, where performance is higher for high-income communities [de Vries et al., 2019] mainly located in Western countries [Shankar et al., 2017]. Similarly problematic, current classifiers perpetuate and amplify current discrimination present in society [Caliskan et al., 2017, Garg et al., 2018]. For example, Kay et al. [2015] highlight the exaggeration of gender bias in occupations by image search systems. These adverse decisions notably arise because image classifiers are prone to biases present in the dataset [Geirhos et al., 2020]. It is therefore essential to identify harmful biases in image representations and assess their effects on the classification predictions, as we do in this chapter.

Addressing dataset biases is not enough, and classifier biases should also be addressed. Zhao et al. [2017b] importantly show that biases can actually be amplified during the image classifier training. Even when balancing a dataset for the protected attribute gender, image classifiers can still surprisingly amplify biases when making a prediction [Wang et al., 2019b]. This outcome emphasizes the importance of considering protected attributes during the training to avoid biased and adverse decisions. A first approach is to perform *fairness through*

Published at *British Machine Vision Conference* [Thong and Snoek, 2021]

blindness, where the objective is to make the feature space blind to the protected attribute [Alvi et al., 2018, Hendricks et al., 2018, Zhang et al., 2018a]. An alternative is to perform *fairness through awareness*, where the classifier label space is explicitly aware of the protected attribute label [Dwork et al., 2012]. To better understand the effectiveness of these methods, Wang et al. [2020b] propose crucial benchmarks in biased image classification. They notably expose the shortcomings of these methods and show that a simple method with separate classifiers is more effective at mitigating biases. Building on this line of work, this paper first identifies a bias direction in the feature space, and secondly address bias mitigation in both label and feature spaces. Another important aspect concerns how to measure the fairness of image classifiers. We borrow from the general fairness literature [Beutel et al., 2017, Dwork et al., 2012, Hardt et al., 2016] to ensure that predictions are similar for all members of a protected attribute, which complements the benchmarks introduced by Wang et al. [2020b] on image classification bias.

Contributions. Our main contribution is to demonstrate the importance of feature and label spaces for addressing image classifier bias. First, we identify a bias direction in the feature space of common classifiers. We aggregate class prototypes to represent every class of each protected attribute value, and show a main direction to explain the maximum variance of the bias. Second, we mitigate biases at both classification and feature levels. We introduce protected classification heads, where each head projects the features to a label embedding space specific to each protected attribute value. This differs from common classification, which usually considers a one-hot encoding for the label space [Luo et al., 2019, Saito et al., 2018, Wang et al., 2020b]. For training, we derive a cosine softmax cross-entropy loss for both multi-class, multi-label and binary classifications. Once trained, we apply in the feature space a bias removal operation to further reduce the bias effect. Experiments on the two benchmarks introduced by Wang et al. [2020b] show the benefits on addressing classifier bias in both feature and label embedding spaces to improve the fairness of the predictions, while preserving the classification performance. The source code is available at: https://github.com/twuilliam/bias-classifiers.

5.2 Related work

Biases in word embeddings. Assessing the presence of biases in word embeddings, especially the gender bias, has received a large attention given their wide range of applications within and beyond natural language processing. The seminal and important work of Bolukbasi et al. [2016] reveals that the difference between female and male entities in word2vec [Mikolov et al., 2013] contains a gender bias direction. This shows that word2vec implicitly captures gender biases, which in return creates sexism in professional activities. Caliskan et al. [2017]

further reveal that multiple human-like biases are actually present in word embeddings. Even contextualized word embeddings [Peters et al., 2018] are affected by a gender bias direction [Zhao et al., 2019], which creates harmful risks [Bender et al., 2021]. To mitigate such gender bias, Bolukbasi et al. [2016] propose a post-processing removal operation while Zhao et al. [2018a] derive regularizers to control the distance between relevant words during training. It is important to note that biases cannot be removed entirely as they can still be recovered to some extent [Gonen and Goldberg, 2019]. As such, methods mainly mitigate biases in models rather than producing debiased models. Inspired by the literature on gender bias identification and mitigation in word embeddings, we pursue an analogous reasoning to show that biases are implicitly encoded in image classification models as well.

Biases in image datasets. As computer vision research relies heavily on datasets, they constitute a main source of biases. Torralba and Efros [2011] notably identify that datasets have a strong built-in bias as they only represent a narrow view of the visual world. Models trained on this narrow view can then rely on spurious correlations and produce detrimental predictions. For fairness and transparency purposes, it becomes necessary to document the dataset creation [Gebru et al., 2018, Hutchinson et al., 2021], as well as detecting the presence of potential biases and harms due to an unfair and unequal label sampling [Birhane and Prabhu, 2021, Dixon et al., 2018, Shankar et al., 2017, Yang et al., 2020]. Towards this end, Bellamy et al. [2018] and Wang et al. [2020a] propose metrics to measure biases, and actionable insights to mitigate them in a dataset. Even though addressing biases from the start of the dataset creation is highly recommended, models can still be affected by spurious correlations and produce unfair decisions [Wang et al., 2019b]. In this chapter, we focus on addressing image classifier bias.

Biases in image classifiers. Searching for a representative subset of image examples provides visual explanations of biases [Kim et al., 2016, Stock and Cisse, 2018]. In this chapter, we rather identify that such bias exists in the feature space in image classifiers. To mitigate image classification bias, training with adversarial learning [Goodfellow et al., 2014] makes the classifier blind to the protected attribute. For example, reducing the gender bias can be achieved by forcing a model to avoid looking at people to produce a prediction [Hendricks et al., 2018, Wang et al., 2019b]. Blindness can also be achieved in the feature space by removing the variation of the protected attribute with a confusion loss [Alvi et al., 2018, Zhang et al., 2018a]. Though, when benchmarking these methods, Wang et al. [2020b] illustrate that adversarial approaches tend to be detrimental as they decrease the performance by making image classifiers less discriminative. At the same time, non-adversarial approaches tend to amplify biases less, while performing well on image classification. Wang et al. [2020b] notably show that encoding the protected attribute into separate heads better mitigates bias. We build on this literature and propose to mitigate biases at classification and feature levels.

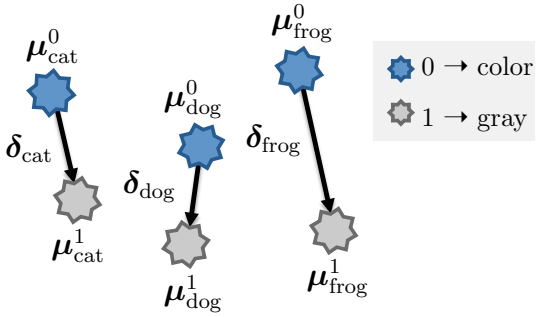
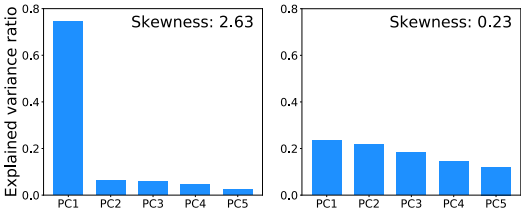


Figure 5.1: **2D toy visualization** of the feature space, where class prototypes $\boldsymbol{\mu}$ represent three categories with a color bias (✸ *vs.* ✸). A bias vector $\boldsymbol{\delta}$ is computed for every class.



(a) $\boldsymbol{\Delta}$. (b) Random $\boldsymbol{\Delta}$.

Figure 5.2: **Bias direction** in the feature space. (a) The PCA of $\boldsymbol{\Delta}$ shows the maximum variance as the bias direction. (b) On a random $\boldsymbol{\Delta}$, the direction disappears and the explained variance is no longer skewed.

Biases benchmarking. There is no consensus (yet) in benchmarking image classifier bias, which makes apple-to-apple comparisons complicated: (a) benchmarks become no longer valid because datasets are taken down for ethical reasons [Peng et al., 2021] (e.g., Racial faces in-the-wild [Wang et al., 2019a] derives from the problematic MS-Celeb-1M [Guo et al., 2016]); (b) datasets are introduced without benchmarks of debiasing methods (e.g., FairFace [Karkkainen and Joo, 2021] only evaluates commercial facial classification systems, and Diversity in Faces [Merler et al., 2019] only provides statistics about craniofacial measures); (c) related works come with differing evaluation settings (e.g., Wang et al. [2019b] train MLP probes to measure model leakage). While addressing algorithm bias in face verification [Gong et al., 2020, Singh et al., 2020, Yin et al., 2019] is crucial, we focus on image classification [Hwang et al., 2020, Kim et al., 2019, Wang et al., 2019b, 2020b]. Therefore, we adopt in this chapter the benchmarks introduced by Wang et al. [2020b] and Kim et al. [2019] in multi-class, multi-label and binary classifications for their comprehensiveness and reproducibility.

5.3 Identifying a bias direction

Problem formulation. We consider the task of image classification where every image $\boldsymbol{x}$ is assigned a label $y \in \mathcal{Y}$. For every image, there also exists a protected attribute value $v \in \mathcal{V}$, on which the classifier should not base its decision. In other words, classifiers should not discriminate against specific members of a protected attribute. In this paper, we consider discrete variables for protected attribute values, and limit the problem to binary values with $\mathcal{V}=\{0,1\}$. For example, we only consider the values "female" and "male" to describe the protected attribute

gender. It is important to note that this formulation is a simplification of the real world where protected attributes go beyond binary values, and are non-discrete.

Image classifiers are typically composed of a base encoder and a projection head. First, a base encoder $f(\cdot)$ extracts the feature representations of images $\boldsymbol{x}$. In our case, this corresponds to a convolutional network and results in $\boldsymbol{h}=f(\boldsymbol{x})$. Second, a projection head $g(\cdot)$ maps the features $\boldsymbol{h}$ to a discriminative space where a class is assigned. In our case this corresponds to a linear projection, or a multilayer perceptron, and results in $\boldsymbol{z}=g(\boldsymbol{h})$ with $\boldsymbol{z} \in \mathbb{R}^M$. For example, in a one-hot encoding, M equals the number of classes.

During training, we are given access to the protected attribute labels and can incorporate it in model formulations. We denote the triplet $(\boldsymbol{x}_i, y_i, v_i)$ as the i-th sample in the training set. During the evaluation, models only have access to the images. In this section, we show that common image classifiers – that do not leverage protected attribute labels during training – still implicitly encode their information in the feature space.

Protected class prototypes. Once a model has been trained, we extract the features $\boldsymbol{h}$ from the training set. We then aggregate prototypes $\boldsymbol{\mu}_y^v$ for every class y and specific to each protected attribute value v, coined as protected class prototypes. For example in Figure 5.1, the class y=cat has two prototypes in the feature space, one for v=*color* images and one for v=*gray*. For any class y with any protected attribute value v, we compute the protected class prototypes as their average representation in the feature space from the training set:

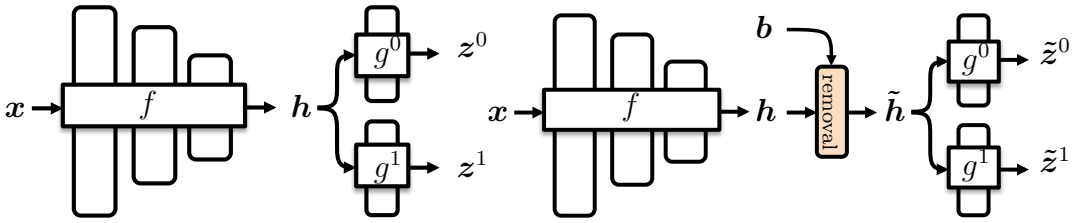
$$\boldsymbol{\mu}_y^v = \frac{1}{N_y^v} \sum_i \mathbb{I}[y_i = y \cap v_i = v] f(\boldsymbol{x}_i), \tag{5.1}$$

where N_y^v is the number of training images of class y with protected attribute v, and $\mathbb{I}[\cdot]$ is the indicator function. Once all protected class prototypes are computed, we extract a subspace that captures the variance of the bias related to the protected attribute.

Bias direction. To identify a bias direction, we experiment with a standard convolutional network trained with a softmax cross-entropy loss on CIFAR-10S [Wang et al., 2020b]. This dataset provides a simple testbed to measure biases in images, as certain classes are skewed towards *gray* images, while others are skewed towards *color* images. Once trained, we aggregate the difference between class prototypes of each protected attribute value for every class:

$$\boldsymbol{\Delta} = \{\boldsymbol{\delta}_y | y \in \mathcal{Y}\} = \{\boldsymbol{\mu}_y^1 - \boldsymbol{\mu}_y^0 | y \in \mathcal{Y}\}. \tag{5.2}$$

Note that for multi-label classification, we consider all binary labels to define $\mathcal{Y}$. Figure 5.2a shows the principal component analysis (PCA) of $\boldsymbol{\Delta}$. When computing the ratio of explained variance of every principal component (PC), a main direction of variance appears. The first PC is more important than the



(a) Protected embeddings for classification.

(b) Bias removal in the feature space.

Figure 5.3: **Mitigating biases in classification predictions.** (a) For classification, we mitigate biases with protected label embeddings where each protected attribute value has its own space. (b) In the feature space, we include a removal operation of the bias direction $\boldsymbol{b}$ in where $\boldsymbol{b}$ is computed from the training set, which is applied once the model has been trained.

others, which yields a high skewness. Figure 5.2b depicts the same analysis on a random $\boldsymbol{\Delta}$, where no main direction appears. Hence, there exists a subspace in the feature space where the bias information is maximized.

5.4 Mitigating classifier bias

Figure 5.3 illustrates our approach to mitigate biases class predictions at both classification and feature levels. For the classification level, we create two protected label embedding spaces, one for each value of the binary protected attribute. For the feature level, we propose a bias removal operation once the model has been trained. The proposed method works for both multi-class, multi-label abd binary settings.

Protected label embeddings. We project features $\boldsymbol{h}$ into embedding spaces, one for each protected attribute value. This results in the embedding representation $\boldsymbol{z}^v = g^v(\boldsymbol{h}) \in \mathbb{R}^M$, where classification occurs. During training, each projection head $g^v(\cdot)$ only sees samples from its assigned attribute value, which creates a protected embedding. By only seeing samples of one protected value, class boundaries are better separated [Saito et al., 2018].

We further push these properties by relying on a cosine softmax cross-entropy loss for classification. $\boldsymbol{z}$ constitutes a discriminative embedding representation with semantic information about classes. This differs from related approaches in domain adaptation [Luo et al., 2019, Saito et al., 2018] or bias mitigation [Wang et al., 2020b], which also show the benefits of separate projection heads with a standard softmax but with a one-hot encoding label space. Below we derive a cosine softmax with protected embeddings for both multi-class, multi-label and binary classifications.

Multi-class classification assigns a label $y \in \mathcal{Y}$ to an image $\boldsymbol{x}$. We introduce a protected weight matrix $\boldsymbol{W}^v \in \mathbb{R}^{|\mathcal{Y}| \times M}$, where M is the size of the embedding space and $v \in \mathcal{V}$ is the protected attribute value. Every row $\boldsymbol{W}^v_{y,:}$ acts as a latent real-valued semantic representation for every class y of each protected attribute v. The objective is then to maximize the cosine similarity, denoted as "sim", between an embedding representation $\boldsymbol{z}^v$ and its corresponding weight representation. This results in the probabilistic model:

$$p(y|\boldsymbol{z}^v, v) = \frac{\exp\Big(\text{sim}(\boldsymbol{W}^v_{y,:}, \boldsymbol{z}^v)/\tau\Big)}{\sum_{y' \in \mathcal{Y}} \exp\Big(\text{sim}(\boldsymbol{W}^v_{y',:}, \boldsymbol{z}^v)/\tau\Big)}, \tag{5.3}$$

where τ is a temperature scaling hyper-parameter. For training, we minimize the cross-entropy loss over the training set of size N:

$$\mathcal{L} = -\frac{1}{N} \sum_{i}^{N} \sum_{v' \in \mathcal{V}} \mathbb{I}[v_i = v'] \log p(y_i|\boldsymbol{x}_i, v_i), \tag{5.4}$$

During inference, the attribute value label is not present. Thus, we perform an ensemble prediction over both heads to predict $\hat{y} = \arg\max_y \sum_{v' \in \mathcal{V}} p(y|\boldsymbol{x}, v')$. Binary classification is a special case where C=1.

Multi-label classification assigns multiple binary labels $\boldsymbol{y}$ to an image $\boldsymbol{x}$. This typically occurs when we want to predict the presence of multiple binary attributes in an image. We denote as $y^{(c)} \in \{0, 1\}$ the label of attribute c. Similar to multi-class classification, we introduce a protected weight matrix $\boldsymbol{W}^{v,c} \in \mathbb{R}^{2 \times M}$ where the two rows correspond to the absence and presence of attribute c for protected attribute v. The resulting probabilistic model is:

$$p(y^{(c)}|\boldsymbol{z}^v, v) = \frac{\exp\Big(\text{sim}\left(\boldsymbol{W}^{v,c}_{y,:}, \boldsymbol{z}^v\right)/\tau\Big)}{\sum_{y' \in \{0,1\}} \exp\Big(\text{sim}(\boldsymbol{W}^{v,c}_{y',:}, \boldsymbol{z}^v)/\tau\Big)}, \tag{5.5}$$

which corresponds to a classifier for two classes. Compared to a binary classifier with a sigmoid function, the softmax function offers more flexibility for the model to represent the negatives. We minimize the cross-entropy loss over all C attributes of the training set of size N:

$$\mathcal{L} = -\frac{1}{N \cdot C} \sum_{i=1}^{N} \sum_{c=1}^{C} \sum_{v' \in \mathcal{V}} \mathbb{I}[v_i = v'] \log p(y_i^c|\boldsymbol{x}_i, v_i). \tag{5.6}$$

During inference, we also perform an ensemble prediction to compute the probability score for the presence of every attribute $\hat{y}^c = \sum_{v' \in \mathcal{V}} p(y^{(c)} = 1|\boldsymbol{x}, v')$.

Bias removal in the feature space. Once trained, we perform the same analysis as in Section 5.3 where we collect protected class prototypes in the feature space from the training set and also apply a principal component analysis on their differences $\boldsymbol{\Delta}$. We refer to the direction of the first principal component of $\boldsymbol{\Delta}$ as $\boldsymbol{b}$. Following Bolukbasi et al. [2016], we first project features $\boldsymbol{h}$ on the bias direction $\boldsymbol{b}$ to obtain $\boldsymbol{h_b}$. Then, we neutralize the bias effect by removing $\boldsymbol{h_b}$ from the features $\boldsymbol{h}$, resulting in the mitigated features $\tilde{\boldsymbol{h}}$ Mathematically, this bias removal operation corresponds to:

$$\tilde{\boldsymbol{h}} = \boldsymbol{h} - \boldsymbol{h_b} = \boldsymbol{h} - \frac{\boldsymbol{h} \cdot \boldsymbol{b}}{\|\boldsymbol{b}\|} \frac{\boldsymbol{b}}{\|\boldsymbol{b}\|}. \tag{5.7}$$

Once $\tilde{\boldsymbol{h}}$ is computed, we can further feed it to each head to get the mitigated protected embeddings $\tilde{\boldsymbol{z}}^v{=}g^v(\tilde{\boldsymbol{h}})$.

Relation with *Domain Independent* [Wang et al., 2020b]. Our proposed method builds on the observation from Wang et al. [2020b] that separate classification heads improve the fairness of the predictions. We differ by demonstrating how feature and label spaces also matter for addressing biases. We find the feature space implicitly encodes a bias direction (Section 5.3) and we derive a bias removal operation to reduce its influence. As distances matter in the feature space, this motivates us to switch from a one-hot encoding to a real-valued vector representation for the label space, where classification now occurs through a cosine embedding softmax.

5.5 Experiments

5.5.1 Fairness metrics

Bias amplification measures whether spurious correlations in the dataset have been amplified by the model during training [Zhao et al., 2017b]. The idea is to compare the number of positive hits of the model for every class and for each value of the protected attribute with the training set statistics. Following Zhao et al. [2017b], the bias amplification score corresponds to:

$$\frac{1}{|\mathcal{Y}|} \sum_{v \in \mathcal{V}} \sum_{y \in \mathcal{Y}} \mathbb{I}_{s(y,v) > \frac{1}{|\mathcal{V}|}} \frac{P_y^v}{P_y^0 + P_y^1} - s(y, v), \tag{5.8}$$

where P_y^v is the number of images positive for class y with a protected attribute v predicted by the model, and $s(y, v){=}N_y^v/(N_y^0 + N_y^1)$ is the ratio of training images N_y^v of class y with a protected attribute v. Intuitively, the score should be as low as possible: a positive value indicates a bias amplification while a negative value indicates a bias reduction. When training and testing sets are not *i.i.d.*, we follow

Wang et al. [2020b] and compute:

$$\frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}\frac{\max(P_y^0,P_y^1)}{P_y^0+P_y^1}-0.5. \tag{5.9}$$

Demographic parity assesses the independence between a prediction $\hat{y}$ and a protected attribute v such that $p(\hat{y}{=}y'|v{=}0){=}p(\hat{y}{=}y'|v{=}1)$ [Dwork et al., 2012, Hardt et al., 2016]. The idea is to compare whether model predictions for a particular class y' are similar for both values of the protected attribute d. Following Beutel et al. [2017], a statistical parity difference score is derived:

$$\frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}\left|\frac{\mathrm{TP}_y^1+\mathrm{FP}_y^1}{N^1}-\frac{\mathrm{TP}_y^0+\mathrm{FP}_y^0}{N^0}\right|, \tag{5.10}$$

where TP_y^v and FP_y^v are the number of true positives and false positives of class y with protected attribute v, and N^v is the number of images with protected attribute v in the evaluation set. When the score tends to zero, the model makes the same rate of predictions for class y' regardless of the protected attribute value.

Equality of opportunity assesses the conditional independence on a particular class y' between a prediction $\hat{y}$ and a protected attribute v such that $p(\hat{y}=y'|y=y',v=0)=p(\hat{y}=y'|y=y',v=1)$ [Hardt et al., 2016]. The idea is to compare whether a model produces a true positive rate (*a.k.a.* recall) for a particular class y' that is the same for both values of the protected attribute d. Following Beutel et al. [2017], an equality of opportunity difference score is derived:

$$\frac{1}{|\mathcal{Y}|}\sum_{y\in\mathcal{Y}}\left|\frac{\mathrm{TP}_y^1}{\mathrm{TP}_y^1+\mathrm{FN}_y^1}-\frac{\mathrm{TP}_y^0}{\mathrm{TP}_y^0+\mathrm{FN}_y^0}\right|, \tag{5.11}$$

where FN_y^v is the number of false negatives of class y with protected attribute v. When the score tends to zero, the model classifies images as class y' correctly regardless of the protected attribute value.

Equality of odds assesses the conditional independence on any class y' between a prediction $\hat{y}$ and a protected attribute v such that $p(\hat{y}=y'|y=y,v=0)=p(\hat{y}=y'|y=y,v=1)$ [Hardt et al., 2016]. Following Bellamy et al. [2018], an equality of odds difference score is derived:

$$0.5\cdot\left(|FPR_y^1-FPR_y^0|+|TPR_y^1-TPR_y^0|\right), \tag{5.12}$$

where FPR_y^v is the false positive rate of class y with protected attribute v and TPR_y^v is the true positive rate. When the score tends to zero, the model exhibits similar true positive and false positive rates for both protected attribute values.

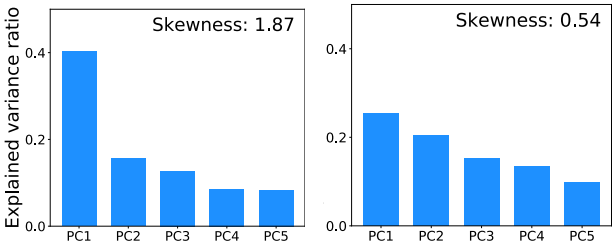
Figure 5.4: **CIFAR-10S** samples, where five classes are skewed towards color images and five other classes are skewed towards gray images in the training set.

5.5.2 Multi-class classification

Setup. We evaluate multi-class classification on the CIFAR-10S dataset [Wang et al., 2020b], which is a biased version of the original CIFAR-10 [Krizhevsky and Hinton, 2009]. A color bias is introduced in the training set, where 5 classes contain 95% gray images and 5% color images, and conversely for the 5 other classes. Figure 5.4 shows examples for every class in their dominant color bias. This creates simple spurious correlations that still affect common classifiers. Two versions of the testing set are considered: one with only gray images and another one with only color images. Although this breaks the *i.i.d.* assumption between training and testing sets, it allows the assessment of the color bias in a controlled manner. We report the per-class accuracy. We rely on ResNet18 [He et al., 2016] as the encoding function f and set each projection function g^v as a fully-connected layer of size M=128 followed by a linear activation. Training is done from scratch with stochastic gradient descent with momentum [Sutskever et al., 2013] for 200 epochs, and the following hyper-parameters: learning rate of 0.1 with a momentum of 0.9, batch size of 128, weight decay of 5e-4, and temperature of 0.1. The learning rate is reduced by a factor 10 every 50 epochs. Note that this setup is identical for all models we compare with, as benchmarked by Wang et al. [2020b]. We report the average over 5 runs.

Bias removal. Once the proposed model has been trained, we compute $\mathbf{\Delta}$ in Eq. 5.2 from the training set of CIFAR-10S. When performing a principal component analysis on $\mathbf{\Delta}$, we observe that there remains a main direction explaining the variance (Figure 5.5a). Though, compared to the baseline model (Figure 5.2a), our model with protected embeddings reduces the skewness from 2.63 to 1.87. This effect is even more noticeable after the bias removal (Figure 5.5b). Indeed, the skewness drops to 0.54 and there is no longer a main direction of variance. The bias removal operation reduces the presence of the bias in the feature space.

Results. Table 5.1 compares our method with four other approaches. Baseline is a standard model trained with an N-way softmax while Oversampling balances out the training by sampling more often underrepresented values of the protected attribute. Adversarial blinds the feature space to the protected attribute. This is achieved either with a uniform confusion loss [Alvi et al., 2018, Tzeng et al., 2015] or a gradient reversal layer [Ganin et al., 2016]. Domain



(a) Before. (b) After.

Figure 5.5: **Bias removal in the feature space effect** on the explained variance of the principal components of $\mathbf{\Delta}$ in CIFAR-10S. After the removal of the bias direction, there is no longer a main direction of variance as illustrated by a reduced skewness.

Model	Loss	Acc. (%,↑)	Bias (↓)	Par. (%,↓)	Opp. (%,↓)	Odds (%,↓)
BASELINE	N-way softmax	88.5±0.3	0.074±0.003	2.90±0.11	12.72±0.51	7.19±0.21
OVERSAMP.	N-way softmax	89.1±0.4	0.066±0.002	2.77±0.67	11.64±0.33	6.91±0.11
ADVERS.	w/ confusion Alvi et al. [2018]	83.8±1.1	0.101±0.007	4.14±0.28	17.55±1.05	9.28±0.73
	w/ ∇ rev. proj. Ganin et al. [2016]	84.1±1.0	0.094±0.011	3.60±0.46	15.60±2.05	7.89±0.81
DOM. DIS.	joint ND-way softmax	90.3±0.5	0.040±0.002	1.65±0.06	7.24±0.31	4.02±0.17
DOM. IND.	N-way softmax×D	**92.0**±0.1	**0.004**±0.001	0.20±0.04	1.02±0.17	0.59±0.12
This paper	N-way cos softmax×D	91.5±0.2	**0.004**±0.000	**0.15**±0.01	**0.88**±0.17	**0.46**±0.07

Table 5.1: **Multi-class classification comparison** on N=10 classes of CIFAR-10S. Despite a small loss in the accuracy score, our proposed approach with a cosine softmax, rather than a common softmax as in DOMAIN INDEPENDENT, improves the fairness of the model in multi-class classification.

DISCRIMINATIVE makes the classification aware of the protected attribute label by assigning a class for every category and protected attribute pair [Dwork et al., 2012]. DOMAIN INDEPENDENT creates two classification heads, one head for each value of the protected attribute [Wang et al., 2020b]. Reported accuracy and bias amplification scores correspond to Wang et al. [2020b], while we reproduce their experiments from the original source code to report the demographic parity and equality of opportunity and odds scores.

Our proposed approach improves upon the other alternatives in the fairness scores. Only in the accuracy metric our model yields slightly lower results compared to DOMAIN INDEPENDENT. This shows that there might exist a trade-off between the downstream task and the fairness of the classifier, as improving both remains challenging. It is interesting that ADVERSARIAL produces worse results than simple methods such as BASELINE or OVERSAMPLING. As ADVERSARIAL

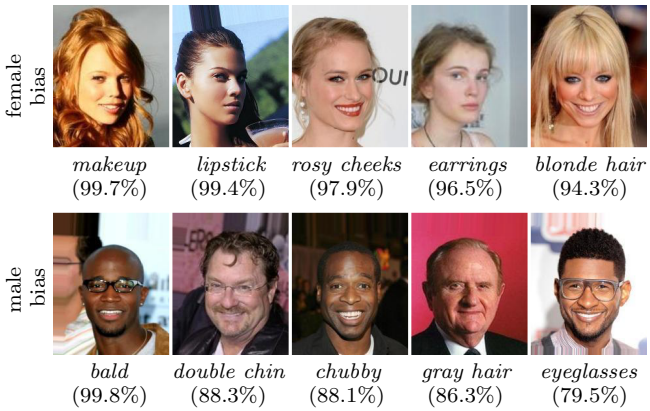


Figure 5.6: **CelebA** samples of the top-5 attributes skewed towards "female" and "male" genders in the training set.

blurs the distinction between both protected attribute values, it also alters the class boundaries, which makes the model less discriminative. DOMAIN DISCRIMINATIVE achieves a lower performance than our model and DOMAIN INDEPENDENT. This highlights the importance of separating the classification heads for each protected attribute value. Overall, our proposed approach with a cosine softmax, rather than a common softmax as in DOMAIN INDEPENDENT, reduces the bias direction in the feature space and improves the fairness in multi-class classification.

5.5.3 Multi-label classification

Setup. We evaluate multi-label classification on the "Align and Cropped" split of the CelebA dataset [Liu et al., 2015], which contains 202,599 face images labeled with 40 binary attributes. Following Wang et al. [2020b], we consider the gender as the protected attribute and train models to predict the other 39 attributes. Figure 5.6 shows examples for the top-5 attributes skewed towards each gender value. During the testing phase, only 34 attributes are considered as the other 5 don't contain both genders. We report the weighted mean average (mAP) precision across the selected attributes. Every positive man image is weighted by $(N_m + N_w)/(2N_m)$ while every positive woman image by $(N_m + N_w)/(2N_w)$, where N_m and N_w are the man and woman image counts in the test set. This weighting ensures a balanced representation of both genders in the evaluation of every attribute.

We rely on ResNet50 [He et al., 2016] pre-trained on ImageNet [Russakovsky et al., 2015] as the encoding function f. We remove the final classification layer and replace it with two fully-connected layers (one for each protected attribute v) of size M=128 followed by a linear activation as the projection function g^v. Training is done with stochastic gradient descent with momentum [Sutskever et al., 2013], and the following hyper-parameters: learning rate of 0.1 with a momentum of 0.9, batch size of 32, and temperature of 0.05. The best model is selected according to the weighted mAP score on the validation set. Compared to

Loss	mAP	Bias	Parity	Opp.	Odds
N sigmoids × D	75.4	-0.039	17.74	14.87	9.19
N cos sigmoids × D	75.5	0.001	11.63	10.29	5.79
+ bias removal	74.7	-0.020	7.43	7.00	**4.00**
N cos softmax × D	**76.3**	-0.006	11.97	10.18	6.06
+ bias removal	75.3	**-0.041**	**6.71**	**6.73**	4.10

Table 5.2: **Label space** comparison on CelebA. An embedding learned with a cosine similarity improves the fairness upon common sigmoids. A softmax with bias removal in the feature space further improves fairness.

Embedding	Cos softmax	mAP	Bias	Parity	Opp.	Odds
Single	N	74.5	-0.039	10.65	14.02	7.77
Single	N × D	67.7	**-0.070**	19.26	21.02	13.54
Protected	N × D	**75.3**	-0.041	**6.71**	**6.73**	**4.10**

Table 5.3: **Single *vs.*protected embedding** comparison on CelebA. Separating the gender information into protected heads results in an increased classification and fairness performance over a single head.

Model	Loss	mAP (%,↑)	Bias (↓)	Par. (%,↓)	Opp. (%,↓)	Odds (%,↓)
Baseline	N sigmoids	74.7	0.010	23.32	24.34	14.28
Adversarial	w/ confusion [Alvi et al., 2018]	71.9	0.019	23.73	28.66	16.69
Dom. Dis.	ND sigmoids	73.8	0.007	22.34	25.35	14.69
Dom. Ind.	N sigmoids × D	**75.4**	-0.039	17.74	14.87	9.19
This paper	N cos softmax × D	75.3	**-0.041**	**6.71**	**6.73**	**4.10**

Table 5.4: **Multi-label classification comparison** of N=34 attributes in CelebA. Despite a small loss in the mAP score, our proposed embedding learned with a cosine softmax, rather than a common softmax with one-hot encoding as in Domain independent, improves the fairness of the model in multi-label classification.

the benchmarks introduced by Wang et al. [2020b], our model training only differs by the optimizer, as we notice some overfitting issues when using Adam [Kingma and Ba, 2015]. The backbone and the rest of the hyper-parameters are similar.

Label space. Table 5.2 compares the different formulations of the label embedding space. Relying a real-valued embedding space learned with a cosine similarity function improves the fairness of the predictions compared to the common one-hot representation. Labels now correspond to a real-valued vector instead of a binary value, which enables a distributed class representation. Switching to a softmax function instead of a sigmoid provides a weight representation for negatives, which in return helps the classification performance. The benefit of negative representations is further highlighted when applying the bias removal operation in the feature space, even though a small drop in the classification score occurs. Overall, the embedding formulation with a softmax cross-entropy in combination

Method	Trained on *EB1*		Trained on *EB2*	
	EB2	*Test*	*EB1*	*Test*
BASELINE	59.86	84.42	57.84	69.75
Alvi et al. [2018]	63.74	85.56	57.33	69.90
Kim et al. [2019]	68.00	86.66	64.18	74.50
This paper	**70.85**	**88.73**	**80.59**	**83.65**

(a) Gender prediction (age protected)

	Trained on *EB1*		Trained on *EB2*	
	EB2	*Test*	*EB1*	*Test*
	54.30	77.17	48.91	61.97
	66.80	75.13	64.16	62.40
	54.27	77.43	62.18	63.04
	35.93	**77.67**	**65.90**	**73.08**

(b) Age prediction (gender protected)

Table 5.5: **Binary classification comparison** on IMDB face dataset. Our formulation of the label embedding space improves the binary classification accuracy (%) with an extreme bias over methods that impose an invariance to the protected attribute in the feature space.

with the bias removal preserves the performance of the downstream task while improving the fairness of the predictions.

Single *vs.* protected embeddings. Table 5.3 assesses the importance of having protected embeddings, with one projection function g^v for each value v of the protected attribute gender. We evaluate the single head setting with and without the protected attribute label in the loss function. When the protected attribute information is available, we basically have two cosine softmax losses, one for each value. Mixing the two losses in one single head is detrimental to the performance as the model gets confused on where to project the inputs in the embedding space. Protected embeddings better separate the gender information for the classification of every attribute given the improved performance and fairness scores.

Results. Table 5.4 compares our model with four other approaches, similarly to the comparison in Table 5.1. Reported mAP and bias amplification scores correspond to Wang et al. [2020b], while we reproduce their experiments to measure demographic parity and equality of opportunity and odds scores. Our proposed approach yields the fairer scores across all evaluated models. And similar to multi-class classification, we also notice a small drop in the downstream task when measuring the mAP. The ADVERSARIAL produces again the worse results across all metrics. This indicates that current methods applying an adversarial training remove more information than the bias, which is detrimental for both the downstream task and the fairness of the model. DOMAIN DISCRIMINATIVE and BASELINE result in a similar performance. Interestingly, a trade-off between the mAP and fairness scores is also present in DOMAIN INDEPENDENT. Our proposed approach improves over DOMAIN INDEPENDENT in the fairness scores by a large margin. Mitigating the bias in both feature and label embedding spaces is then preferred over methods that only address one of the two as it improves the fairness of the model predictions.

Binary classification. We further evaluate the binary classification task on the "cropped" split of the IMDB face dataset [Rothe et al., 2018]. Following Kim et al. [2019], we create three sets with an extreme bias: *EB1* comprises women $\leq$ 29 years old (yo) and men $\geq$ 40 yo; *EB2* has women $\geq$ 40 yo and men $\leq$ 29 yo; and *Test* has women and men $\leq$ 29 yo and $\geq$ 40 yo. They contain 36,004, 16,800 and 13,129 face images of celebrities, respectively. Similar to Kim et al. [2019], we learn to predict the gender with age as a protected attribute (and conversely), and rely on ResNet18 [He et al., 2016] pre-trained on ImageNet [Russakovsky et al., 2015] as the encoding function f. We add a fully-connected layer of size M=128 with linear activation for each projection function g^v. Training is done with stochastic gradient descent with momentum [Sutskever et al., 2013] for 5 epochs, and the hyper-parameters: learning rate of 0.1 with momentum of 0.9 and an exponential decay of 0.999, batch size of 128, and temperature of 0.1.

Table 5.5 compares our model with three other approaches. Baseline is also a standard model trained with binary cross-entropy. Both Alvi et al. [2018] and Kim et al. [2019] mitigate the extreme bias by making the feature space invariant to the protected attribute. Kim et al. [2019] rely on an adversarial formulation [Chen et al., 2016, Ganin et al., 2016], improving over Alvi et al. [2018]. Given the binary classification setting, we did not apply a bias removal operation, as a PCA on two samples is not pertinent. Still, our formulation of the label space improves the performance in both the gender and age settings. Only when predicting age and training on *EB1*, our model struggles a bit as it tends to overfit quickly. This binary classification comparison further confirms that simpler alternatives to adversarial losses can better mitigate biases present during training.

5.6 Conclusion

Reducing the effect of adverse decisions involves the identification and mitigation of biases within model representations. In this paper, we focus on biases coming from binary protected attributes. First, we identify a bias direction in the feature space of common image classifiers, where the first principal component of the difference of protected class prototypes captures bias variation. Second, building on this observation, we mitigate classification bias with protected projection heads that learn a label embedding space for each protected attribute value. This formulation trained with a cosine softmax cross-entropy loss improves upon the common one-hot encoding in terms of fairness for both multi-class, multi-label and binary classifications. Furthermore, removing the bias direction in the feature space reduces even further the bias effect on the classifier predictions. Overall, addressing image classifier bias on both feature and label embedding levels improves the fairness of predictions, while preserving the classification performance.

Chapter 6

Conclusions

6.1 Thesis summary

This thesis investigates how visual similarities help to learn models robust to bias in computer vision tasks. Models should be able to adapt constantly to new and changing environments without being biased to what they have seen during training. Throughout this thesis, we focus on the research question: *how to learn visual similarities robust to bias?* We explore this question through the multiple facets of biases with a common theme on visual similarities to address them. We start with categorization across multiple domains, then investigate the ability to retrieve seen and unseen attribute combinations, followed by the study of the confidence of image classifiers towards seen classes, and finally the identification and mitigation of adverse predictions in image classifiers.

Chapter 2 addresses cross-domain visual search, where visual queries retrieve category samples from a different domain. For example, we may want to sketch an airplane and retrieve photographs of airplanes. Despite considerable progress, the search occurs in a closed setting between two pre-defined domains. In this chapter, we make the step towards an open setting where multiple visual domains are available. This notably translates into a search between any pair of domains, from a combination of domains or within multiple domains. We introduce a simple –yet effective– approach. We formulate the search as a mapping from every visual domain to a common semantic space, where categories are represented by hyperspherical prototypes. Open cross-domain visual search is then performed by searching in the common semantic space, regardless of which domains are used as source or target. Domains are combined in the common space to search from or within multiple domains simultaneously. A separate training of every domain-specific mapping function enables an efficient scaling to any number of domains without affecting the search performance. We empirically illustrate our capability to perform open cross-domain visual search in three different scenarios. Our approach is also competitive with respect to existing closed settings, where

we obtain state-of-the-art results on several benchmarks for three sketch-based search tasks.

Chapter 3 introduces a diversely supervised visual product search, where queries specify a diverse set of labels to search for. Where previous works have focused on representing attribute, instance, or category labels individually, we consider them together to create a diverse set of labels for visually describing products. We learn an embedding from the supervisory signal provided by every label to encode their interrelationships. Once trained, every label has a corresponding visual representation in the embedding space, which is an aggregation of selected items from the training set. At search time, composite query representations retrieve images that match a specific set of diverse labels. We form composite query representations by averaging over the aggregated representations of each diverse label in the specific set. For evaluation, we extend existing product datasets of cars and clothes with a diverse set of labels. Experiments show the benefits of our embedding for diversely supervised visual product search in seen and unseen product combinations and for discovering product design styles.

Chapter 4 looks at the problem of generalized zero-shot learning, which aims to recognize inputs from both seen and unseen classes. Yet, existing methods tend to be biased towards the classes seen during training. In this chapter, we strive to mitigate this bias. We propose a bias-aware learner to map inputs to a semantic embedding space for generalized zero-shot learning. During training, the model learns to regress to real-valued class prototypes in the embedding space with temperature scaling while a margin-based bidirectional entropy term regularizes seen and unseen probabilities. Relying on a real-valued semantic embedding space provides a versatile approach, as the model can operate on different types of semantic information for both seen and unseen classes. Experiments are carried out on four benchmarks for generalized zero-shot learning and demonstrate the benefits of the proposed bias-aware classifier, both as a stand-alone method or in combination with generated features.

Chapter 5 addresses image classifier bias, with a focus on both feature and label embedding spaces. Previous works have shown that spurious correlations from protected attributes, such as age, gender, or skin tone, can cause adverse decisions. To balance potential harms, there is a growing need to identify and mitigate image classifier bias. First, we identify in the feature space a bias direction. We compute class prototypes of each protected attribute value for every class, and reveal an existing subspace that captures the maximum variance of the bias. Second, we mitigate biases by mapping image inputs to label embedding spaces. Each value of the protected attribute has its projection head where classes are embedded through a latent vector representation rather than a common one-hot encoding. Once trained, we further reduce in the feature space the bias effect by removing its direction. Evaluation on biased image datasets, for multi-class, multi-label and binary classifications, shows the effectiveness of tackling both feature and label embedding spaces in improving the fairness of the

classifier predictions, while preserving classification performance.

6.2 Closing Remarks

In this thesis, we have unveiled the potential of visual similarities to address several biases arising in computer vision tasks. Throughout the thesis, the approach has been to derive a loss function to learn an embedding space robust to various biases. By learning visual associations, models are able to generalize better and cope with a constantly changing environment.

There remains more to be explored to identify and address adverse biases in computer vision. With the growing ubiquity of computer vision, current models are now at risk of producing potential harms. As such, we need to ensure that computer vision models don't reproduce biases present in society or amplify them even more. Identifying the source of biases then becomes critical to mitigate their effects on the predictions. This leads towards fundamental changes in the data collection and the training of the models to improve their generalizability, robustness, and fairness.

Bibliography

Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. Diversifying search results. In *Web Search and Data Mining (WSDM)*, 2009.

Kenan E. Ak, Ashraf A. Kassim, Joo Hwee Lim, and Jo Yew Tham. Learning attribute representations with localization for flexible fashion search. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Kenan E. Ak, Joo Hwee Lim, Jo Yew Tham, and Ashraf A. Kassim. Attribute manipulation generative adversarial networks for fashion images. In *International Conference on Computer Vision (ICCV)*, 2019.

Zeynep Akata, Scott Reed, Daniel Walter, Honglak Lee, and Bernt Schiele. Evaluation of output embeddings for fine-grained image classification. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Zeynep Akata, Florent Perronnin, Zaid Harchaoui, and Cordelia Schmid. Label-embedding for image classification. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 38(7), 2016.

Ziad Al-Halah, Rainer Stiefelhagen, and Kristen Grauman. Fashion forward: Forecasting visual style in fashion. In *International Conference on Computer Vision (ICCV)*, 2017.

Mohsan Alvi, Andrew Zisserman, and Christoffer Nellåker. Turning a blind eye: Explicit removal of biases and variation from deep neural network embeddings. In *European Conference in Computer Vision Workshops (ECCVw)*, 2018.

Yuval Atzmon and Gal Chechik. Probabilistic and-or attribute grouping for zero-shot learning. In *Uncertainty in Artificial Intelligence (UAI)*, 2018.

Yuval Atzmon and Gal Chechik. Adaptive confidence smoothing for generalized zero-shot learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Moshe Bar. The proactive brain: memory for predictions. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 364(1521), 2009.

Sean Bell and Kavita Bala. Learning visual similarity for product design with convo-

lutional neural networks. *ACM Transactions on Graphics (TOG)*, 34(4), 2015.

Rachel KE Bellamy, Kuntal Dey, Michael Hind, Samuel C Hoffman, Stephanie Houde, Kalapriya Kannan, Pranay Lohia, Jacquelyn Martino, Sameep Mehta, Aleksandra Mojsilovic, Seema Nagar, Karthikeyan Natesan Ramamurthy, John Richards, Diptikalyan Saha, Prasanna Sattigeri, Moninder Singh, Kush R. Varshney, and Yunfeng Zhang. Ai fairness 360: An extensible toolkit for detecting, understanding, and mitigating unwanted algorithmic bias. In *arXiv:1810.01943*, 2018.

Emily M Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. On the dangers of stochastic parrots: Can language models be too big? In *Fairness, Accountability, and Transparency (FAccT)*, 2021.

Alessandro Bergamo, Lorenzo Torresani, and Andrew W. Fitzgibbon. Picodes: Learning a compact code for novel-category recognition. In *Neural Information Processing Systems (NeurIPS)*, 2011.

Alex Beutel, Jilin Chen, Zhe Zhao, and Ed H Chi. Data decisions and theoretical implications when adversarially learning fair representations. In *Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*, 2017.

Abeba Birhane and Vinay Uday Prabhu. Large image datasets: A pyrrhic win for computer vision? In *Winter Conference on Applications of Computer Vision (WACV)*, 2021.

Gilles Blanchard, Gyemin Lee, and Clayton Scott. Generalizing from several related classification tasks to a new unlabeled sample. In *Neural Information Processing Systems (NeurIPS)*, 2011.

Tolga Bolukbasi, Kai-Wei Chang, James Y Zou, Venkatesh Saligrama, and Adam T Kalai. Man is to computer programmer as woman is to homemaker? debiasing word embeddings. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2015.

Tu Bui, Leonardo Ribeiro, Moacir Ponti, and John Collomosse. Compact descriptors for sketch-based image retrieval using a triplet loss convolutional neural network. *Computer Vision and Image Understanding (CVIU)*, 164, 2017.

Tu Bui, Leonardo Ribeiro, Moacir Ponti, and John Collomosse. Sketching out the details: Sketch-based image retrieval using convolutional neural networks with multi-stage regression. *Computers & Graphics*, 71, 2018.

Joy Buolamwini and Timnit Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. In *Fairness, Accountability, and Transparency (FAccT)*, 2018.

Aylin Caliskan, Joanna J Bryson, and Arvind Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 356(6334), 2017.

Fabio M. Carlucci, Antonio D'Innocente, Silvia Bucci, Barbara Caputo, and Tatiana

Tommasi. Domain generalization by solving jigsaw puzzles. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Synthesized classifiers for zero-shot learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Soravit Changpinyo, Wei-Lun Chao, and Fei Sha. Predicting visual exemplars of unseen classes for zero-shot learning. In *International Conference on Computer Vision (ICCV)*, 2017.

Soravit Changpinyo, Wei-Lun Chao, Boqing Gong, and Fei Sha. Classifier and exemplar synthesis for zero-shot learning. *International Journal of Computer Vision (IJCV)*, 128(1), 2020.

Wei-Lun Chao, Soravit Changpinyo, Boqing Gong, and Fei Sha. An empirical study and analysis of generalized zero-shot learning for object recognition in the wild. In *European Conference on Computer Vision (ECCV)*, 2016.

Chandramani Chaudhary, Poonam Goyal, Navneet Goyal, and Yi-Ping Phoebe Chen. Image retrieval for complex queries using knowledge embedding. *Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 16(1), 2020.

Gal Chechik, Uri Shalit, Varun Sharma, and Samy Bengio. An online algorithm for large scale image similarity learning. In *Neural Information Processing Systems (NeurIPS)*, 2009.

Jiaxin Chen and Yi Fang. Deep cross-modality adaptation via semantics preserving adversarial learning for sketch-based 3d shape retrieval. In *European Conference on Computer Vision (ECCV)*, 2018.

Jiaxin Chen, Jie Qin, Li Liu, Fan Zhu, Fumin Shen, Jin Xie, and Ling Shao. Deep sketch-shape hashing with segmented 3d stochastic viewing. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Xi Chen, Yan Duan, Rein Houthooft, John Schulman, Ilya Sutskever, and Pieter Abbeel. Infogan: Interpretable representation learning by information maximizing generative adversarial nets. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. In *arXiv:2003.04297*, 2020.

Soumith Chintala, Marc'Aurelio Ranzato, Arthur Szlam, Yuandong Tian, Mark Tygert, and Wojciech Zaremba. Scale-invariant learning and convolutional networks. *Applied and Computational Harmonic Analysis*, 42(1), 2017.

Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *Computer Vision and Pattern Recognition (CVPR)*, 2005.

Gabriela Csurka. *Domain adaptation in computer vision applications*. Springer, 2017.

Guoxian Dai, Jin Xie, Fan Zhu, and Yi Fang. Deep correlated metric learning for sketch-based 3d shape retrieval. In *AAAI Conference on Artificial Intelligence (AAAI)*,

2017.

Terrance de Vries, Ishan Misra, Changhan Wang, and Laurens van der Maaten. Does object recognition work for everyone? In *Computer Vision and Pattern Recognition Workshops (CVPRw)*, 2019.

Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. Arcface: Additive angular margin loss for deep face recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Thomas Deselaers and Vittorio Ferrari. Visual and semantic similarity in imagenet. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

Sounak Dey, Pau Riba, Anjan Dutta, Josep Llados, and Yi-Zhe Song. Doodle to search: Practical zero-shot sketch-based image retrieval. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and mitigating unintended bias in text classification. In *Artificial Intelligence, Ethics, and Society (AIES)*, 2018.

Qi Dou, Daniel C Castro, Konstantinos Kamnitsas, and Ben Glocker. Domain generalization via model-agnostic learning of semantic features. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Anjan Dutta and Zeynep Akata. Semantically tied paired cycle consistency for zero-shot sketch-based image retrieval. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Titir Dutta and Soma Biswas. Style-guided zero-shot sketch-based image retrieval. In *British Machine Vision Conference (BMVC)*, 2019.

Cynthia Dwork, Moritz Hardt, Toniann Pitassi, Omer Reingold, and Richard Zemel. Fairness through awareness. In *Innovations in Theoretical Computer Science Conference*, 2012.

Mathias Eitz, Kristian Hildebrand, Tamy Boubekeur, and Marc Alexa. Sketch-based image retrieval: Benchmark and bag-of-features descriptors. *Transactions on Visualization and Computer Graphics (TVCG)*, 17(11), 2010.

Mathias Eitz, James Hays, and Marc Alexa. How do humans sketch objects? *ACM Transactions on Graphics (TOG)*, 31(4), 2012.

Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *Computer Vision and Pattern Recognition (CVPR)*, 2009.

Rafael Felix, Vijay BG Kumar, Ian Reid, and Gustavo Carneiro. Multi-modal cycle-consistent generalized zero-shot learning. In *European Conference on Computer Vision (ECCV)*, 2018.

Vittorio Ferrari and Andrew Zisserman. Learning visual attributes. In *Neural Information Processing Systems (NeurIPS)*, 2008.

Andrea Frome, Yoram Singer, and Jitendra Malik. Image retrieval and classification using local distance functions. In *Neural Information Processing Systems (NeurIPS)*, 2007.

Andrea Frome, Greg S Corrado, Jon Shlens, Samy Bengio, Jeff Dean, Marc'Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *Neural Information Processing Systems (NeurIPS)*, 2013.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research (JMLR)*, 17(1), 2016.

Nikhil Garg, Londa Schiebinger, Dan Jurafsky, and James Zou. Word embeddings quantify 100 years of gender and ethnic stereotypes. *Proceedings of the National Academy of Sciences (PNAS)*, 115(16), 2018.

Timnit Gebru, Jamie Morgenstern, Briana Vecchione, Jennifer Wortman Vaughan, Hanna Wallach, Hal Daumé III, and Kate Crawford. Datasheets for datasets. In *Fairness, Accountability, and Transparency in Machine Learning (FAT/ML)*, 2018.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11), 2020.

Hila Gonen and Yoav Goldberg. Lipstick on a pig: Debiasing methods cover up systematic gender biases in word embeddings but do not remove them. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

Sixue Gong, Xiaoming Liu, and Anil K Jain. Jointly de-biasing face recognition and demographic attribute estimation. In *European Conference on Computer Vision (ECCV)*, 2020.

Yunchao Gong, Svetlana Lazebnik, Albert Gordo, and Florent Perronnin. Iterative quantization: A procrustean approach to learning binary codes for large-scale image retrieval. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35 (12), 2012.

Abel Gonzalez-Garcia, Joost van de Weijer, and Yoshua Bengio. Image-to-image translation for cross-domain disentanglement. In *Neural Information Processing Systems (NeurIPS)*, 2018.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Neural Information Processing Systems (NeurIPS)*, 2014.

Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep Learning*. MIT Press, 2016.

Albert Gordo and Diane Larlus. Beyond instance-level image retrieval: Leveraging captions to learn a global visual representation for semantic retrieval. In *Computer*

Vision and Pattern Recognition (CVPR), 2017.

Yandong Guo, Lei Zhang, Yuxiao Hu, X. He, and Jianfeng Gao. MS-Celeb-1M: A dataset and benchmark for large-scale face recognition. In *European Conference on Computer Vision (ECCV)*, 2016.

Raia Hadsell, Sumit Chopra, and Yann LeCun. Dimensionality reduction by learning an invariant mapping. In *Computer Vision and Pattern Recognition (CVPR)*, 2006.

Xintong Han, Zuxuan Wu, Phoenix X. Huang, Xiao Zhang, Menglong Zhu, Yuan Li, Yang Zhao, and Larry S. Davis. Automatic spatially-aware fashion concept discovery. In *International Conference on Computer Vision (ICCV)*, 2017.

Moritz Hardt, Eric Price, and Nati Srebro. Equality of opportunity in supervised learning. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2020.

Lisa Anne Hendricks, Kaylee Burns, Kate Saenko, Trevor Darrell, and Anna Rohrbach. Women also snowboard: Overcoming bias in captioning models. In *European Conference on Computer Vision (ECCV)*, 2018.

Alexander Hermans, Lucas Beyer, and Bastian Leibe. In defense of the triplet loss for person re-identification. *arXiv:1703.07737*, 2017.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. In *Neural Information Processing Systems Workshops (NeurIPSw)*, 2014.

Wei-Lin Hsiao and Kristen Grauman. Creating capsule wardrobes from fashion images. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Conghui Hu, Da Li, Yi-Zhe Song, Tao Xiang, and Timothy M Hospedales. Sketch-a-classifier: sketch-based photo classifier generation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018a.

Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2018b.

Rui Hu and John Collomosse. A performance evaluation of gradient field hog descriptor for sketch based image retrieval. *Computer Vision and Image Understanding (CVIU)*, 117(7), 2013.

Rui Hu, Tinghuai Wang, and John Collomosse. A bag-of-regions approach to sketch-based image retrieval. In *International Conference on Image Processing (ICIP)*, 2011.

Junshi Huang, Si Liu, Junliang Xing, Tao Mei, and Shuicheng Yan. Circle & search: Attribute-aware shoe retrieval. *Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 11(1), 2014.

Junshi Huang, Rogerio S Feris, Qiang Chen, and Shuicheng Yan. Cross-domain image retrieval with a dual attribute-aware ranking network. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Ben Hutchinson, Andrew Smart, Alex Hanna, Emily Denton, Christina Greer, Oddur Kjartansson, Parker Barnes, and Margaret Mitchell. Towards accountability for machine learning datasets: Practices from software engineering and infrastructure. In *Fairness, Accountability, and Transparency (FAccT)*, 2021.

Sunhee Hwang, Sungho Park, Pilhyeon Lee, Seogkyu Jeon, Dohyung Kim, and Hyeran Byun. Exploiting transferable knowledge for fairness-aware image classification. In *Asian Conference on Computer Vision (ACCV)*, 2020.

Sarah Ibrahimi, Shuo Chen, Devanshu Arya, Arthur Câmara, Yunlu Chen, Tanja Crijns, Maurits van der Goes, Thomas Mensink, Emiel van Miltenburg, Daan Odijk, William Thong, Jiaojiao Zhao, and Pascal Mettes. Interactive exploration of journalistic video footage through multimodal semantic matching. In *ACM Multimedia (MM)*, 2019.

Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning (ICML)*, 2015.

Charles E Jacobs, Adam Finkelstein, and David H Salesin. Fast multiresolution image querying. In *International Conference on Computer Graphics and Interactive Techniques (SIGGRAPH)*, 1995.

Yushi Jing, David Liu, Dmitry Kislyuk, Andrew Zhai, Jiajing Xu, Jeff Donahue, and Sarah Tavel. Visual search at pinterest. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2015.

Jeff Johnson, Matthijs Douze, and Hervé Jégou. Billion-scale similarity search with gpus. *arXiv:1702.08734*, 2017.

Justin Johnson, Ranjay Krishna, Michael Stark, Li-Jia Li, David Shamma, Michael Bernstein, and Li Fei-Fei. Image retrieval using scene graphs. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Kimmo Karkkainen and Jungseock Joo. Fairface: Face attribute dataset for balanced race, gender, and age for bias measurement and mitigation. In *Winter Conference on Applications of Computer Vision (WACV)*, 2021.

Andrej Karpathy and Li Fei-Fei. Deep visual-semantic alignments for generating image descriptions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Toshikazu Kato. Database architecture for content-based image retrieval. In *Image Storage and Retrieval Systems*, volume 1662, 1992.

Matthew Kay, Cynthia Matuszek, and Sean A Munson. Unequal representation and gender stereotypes in image search results for occupations. In *Conference on Human Factors in Computing Systems (CHI)*, 2015.

M Hadi Kiapour, Kota Yamaguchi, Alexander C Berg, and Tamara L Berg. Hipster

wars: Discovering elements of fashion styles. In *European Conference on Computer Vision (ECCV)*, 2014.

M Hadi Kiapour, Xufeng Han, Svetlana Lazebnik, Alexander C Berg, and Tamara L Berg. Where to buy it: Matching street clothing photos in online shops. In *International Conference on Computer Vision (ICCV)*, 2015.

Been Kim, Oluwasanmi Koyejo, Rajiv Khanna, et al. Examples are not enough, learn to criticize! criticism for interpretability. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Byungju Kim, Hyunwoo Kim, Kyungsu Kim, Sungjin Kim, and Junmo Kim. Learning not to learn: Training deep neural networks with biased data. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Gunhee Kim, Seungwhan Moon, and Leonid Sigal. Ranking and retrieval of image sequences from multiple paragraph queries. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015.

Adriana Kovashka, Devi Parikh, and Kristen Grauman. Whittlesearch: Image search with relative attribute feedback. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Internation Conference in Computer Vision Workshops (ICCVw)*, 2013.

Alex Krizhevsky and Geoffrey Hinton. Learning multiple layers of features from tiny images. 2009.

Neeraj Kumar, Alexander Berg, Peter N Belhumeur, and Shree Nayar. Describable visual attributes for face verification and image search. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 33(10), 2011.

Vinay Kumar Verma, Gundeep Arora, Ashish Mishra, and Piyush Rai. Generalized zero-shot learning via synthesized examples. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Christoph H Lampert, Hannes Nickisch, and Stefan Harmeling. Attribute-based classification for zero-shot visual object categorization. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 36(3), 2014.

Tian Lan, Weilong Yang, Yang Wang, and Greg Mori. Image retrieval with structured object queries using latent ranking svm. In *European Conference on Computer Vision (ECCV)*, 2012.

Kuang-Huei Lee, Xi Chen, Gang Hua, Houdong Hu, and Xiaodong He. Stacked cross attention for image-text matching. In *European Conference on Computer Vision (ECCV)*, 2018.

Bo Li, Yijuan Lu, Afzal Godil, Tobias Schreck, Masaki Aono, Henry Johan, Jose M

Saavedra, and Shoki Tashiro. Shrec'13 track: large scale sketch-based 3d shape retrieval. In *Eurographics workshop on 3D object retrieval*, 2013.

Bo Li, Yijuan Lu, Afzal Godil, Tobias Schreck, Benjamin Bustos, Alfredo Ferreira, Takahiko Furuya, Manuel J Fonseca, Henry Johan, Takahiro Matsuda, et al. A comparison of methods for sketch-based 3d shape retrieval. *Computer Vision and Image Understanding (CVIU)*, 119, 2014a.

Bo Li, Yijuan Lu, Chunyuan Li, Afzal Godil, Tobias Schreck, Masaki Aono, Martin Burtscher, Hongbo Fu, Takahiko Furuya, Henry Johan, et al. Shrec'14 track: Extended large scale sketch-based 3d shape retrieval. In *Eurographics workshop on 3D object retrieval*, 2014b.

Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Deeper, broader and artier domain generalization. In *International Conference on Computer Vision (ICCV)*, 2017.

Jingjing Li, Mengmeng Jing, Ke Lu, Zhengming Ding, Lei Zhu, and Zi Huang. Leveraging the invariant side of generative zero-shot learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Li Liu, Fumin Shen, Yuming Shen, Xianglong Liu, and Ling Shao. Deep sketch hashing: Fast free-hand sketch-based image retrieval. In *Computer Vision and Pattern Recognition (CVPR)*, 2017a.

Qing Liu, Lingxi Xie, Huiyu Wang, and Alan Yuille. Semantic-aware knowledge preservation for zero-shot sketch-based image retrieval. In *International Conference on Computer Vision (ICCV)*, 2019.

Shichen Liu, Mingsheng Long, Jianmin Wang, and Michael I Jordan. Generalized zero-shot learning with deep calibration network. In *Neural Information Processing Systems (NeurIPS)*, 2018.

Si Liu, Zheng Song, Guangcan Liu, Changsheng Xu, Hanqing Lu, and Shuicheng Yan. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2017b.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *International Conference on Computer Vision (ICCV)*, 2015.

Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations (ICLR)*, 2017.

Peng Lu, Gao Huang, Yanwei Fu, Guodong Guo, and Hangyu Lin. Learning large euclidean margin for sketch-based image retrieval. *arXiv:1812.04275*, 2018.

Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research (JMLR)*, 9(Nov), 2008.

Utkarsh Mall, Kevin Matzen, Bharath Hariharan, Noah Snavely, and Kavita Bala. GeoStyle: Discovering fashion trends and events. In *International Conference on Computer Vision (ICCV)*, 2019.

Christopher D. Manning, Prabhakar Raghavan, and Hinrich Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton van Den Hengel. Image-based recommendations on styles and substitutes. In *ACM SIGIR*, 2015.

Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 35(11), 2013.

Michele Merler, Nalini Ratha, Rogerio S Feris, and John R Smith. Diversity in faces. *arXiv:1901.10436*, 2019.

Pascal Mettes, Elise van der Pol, and Cees G. M. Snoek. Hyperspherical prototype networks. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Pascal Mettes, William Thong, and Cees G. M. Snoek. Object priors for classifying and localizing unseen actions. *International Journal of Computer Vision (IJCV)*, 129(6), 2021.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. Efficient estimation of word representations in vector space. In *International Conference on Learning Representations (ICLR)*, 2013.

Yair Movshovitz-Attias, Alexander Toshev, Thomas K Leung, Sergey Ioffe, and Saurabh Singh. No fuss distance metric learning using proxies. In *International Conference on Computer Vision (ICCV)*, 2017.

Vinod Nair and Geoffrey E Hinton. Rectified linear units improve restricted boltzmann machines. In *International Conference on Machine Learning (ICML)*, 2010.

Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP)*, 2008.

Mark Palatucci, Dean Pomerleau, Geoffrey E Hinton, and Tom M Mitchell. Zero-shot learning with semantic output codes. In *Neural Information Processing Systems (NeurIPS)*, 2009.

Devi Parikh and Kristen Grauman. Relative attributes. In *International Conference on Computer Vision (ICCV)*, 2011.

Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory

Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Neural Information Processing Systems (NeurIPS)*, 2019.

Genevieve Patterson and James Hays. Sun attribute database: Discovering, annotating, and recognizing scene attributes. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

Kenny Peng, Arunesh Mathur, and Arvind Narayanan. Mitigating dataset harms requires stewardship: Lessons from 1000 papers. *arXiv:2108.02922*, 2021.

Xingchao Peng, Ben Usman, Neela Kaushik, Judy Hoffman, Dequan Wang, and Kate Saenko. Visda: The visual domain adaptation challenge. *arXiv:1710.06924*, 2017.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *International Conference on Computer Vision (ICCV)*, 2019.

Jeffrey Pennington, Richard Socher, and Christopher Manning. Glove: Global vectors for word representation. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2014.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. Deep contextualized word representations. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.

Bui Tuong Phong. Illumination for computer generated pictures. *Communications of the ACM*, 18(6), 1975.

Anran Qi, Yi-Zhe Song, and Tao Xiang. Semantic embedding for sketch-based 3d shape retrieval. In *British Machine Vision Conference (BMVC)*, 2018.

Yonggang Qi, Yi-Zhe Song, Honggang Zhang, and Jun Liu. Sketch-based image retrieval via siamese convolutional neural network. In *International Conference on Image Processing (ICIP)*, 2016.

Mohammad Rastegari, Ali Diba, Devi Parikh, and Ali Farhadi. Multi-attribute queries: To merge or not to merge? In *Computer Vision and Pattern Recognition (CVPR)*, 2013.

Hareesh Ravi, Lezi Wang, Carlos Muniz, Leonid Sigal, Dimitris Metaxas, and Mubbasir Kapadia. Show me a story: Towards coherent neural story illustration. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Scott Reed, Zeynep Akata, Honglak Lee, and Bernt Schiele. Learning deep representations of fine-grained visual descriptions. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Radim Řehůřek and Petr Sojka. Software Framework for Topic Modelling with Large Corpora. In *LREC Workshop on New Challenges for NLP Frameworks*. ELRA, 2010.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2019.

Mengye Ren, Eleni Triantafillou, Sachin Ravi, Jake Snell, Kevin Swersky, Joshua B Tenenbaum, Hugo Larochelle, and Richard S Zemel. Meta-learning for semi-supervised few-shot classification. In *International Conference on Learning Representations (ICLR)*, 2018.

Bernardino Romera-Paredes and Philip Torr. An embarrassingly simple approach to zero-shot learning. In *International Conference on Machine Learning (ICML)*, 2015.

Eleanor Heider Rosch. Principles of categorization. *Cognition and categorization*, 1978.

Rasmus Rothe, Radu Timofte, and Luc Van Gool. Deep expectation of real and apparent age from a single image without facial landmarks. *International Journal of Computer Vision (IJCV)*, 126(2-4):144–157, 2018.

Sebastian Ruder, Joachim Bingel, Isabelle Augenstein, and Anders Søgaard. Latent multi-task architecture learning. In *AAAI Conference on Artificial Intelligence (AAAI)*, volume 33, 2019.

Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *International Journal of Computer Vision (IJCV)*, 115(3), 2015.

Jose M Saavedra. Sketch based image retrieval using a soft computation of the histogram of edge local orientations (s-helo). In *International Conference on Image Processing (ICIP)*, 2014.

Mohammad Amin Sadeghi and Ali Farhadi. Recognition using visual phrases. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

Kuniaki Saito, Kohei Watanabe, Yoshitaka Ushiku, and Tatsuya Harada. Maximum classifier discrepancy for unsupervised domain adaptation. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Patsorn Sangkloy, Nathan Burnell, Cusuh Ham, and James Hays. The sketchy database: Learning to retrieve badly drawn bunnies. *ACM Transactions on Graphics (TOG)*, 2016.

Walter J Scheirer, Neeraj Kumar, Peter N Belhumeur, and Terrance E Boult. Multi-attribute spaces: Calibration for attribute fusion and similarity search. In *Computer Vision and Pattern Recognition (CVPR)*, 2012.

Edgar Schönfeld, Sayna Ebrahimi, Samarth Sinha, Trevor Darrell, and Zeynep Akata. Generalized zero-and few-shot learning via aligned variational autoencoders. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Shreya Shankar, Yoni Halpern, Eric Breck, James Atwood, Jimbo Wilson, and D Sculley. No classification without representation: Assessing geodiversity issues in open data sets for the developing world. In *NeurIPSw*, 2017.

Claude Elwood Shannon. A mathematical theory of communication. *Bell system technical journal*, 27(3), 1948.

Yuming Shen, Li Liu, Fumin Shen, and Ling Shao. Zero-shot sketch-image hashing. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Philip Shilane, Patrick Min, Michael Kazhdan, and Thomas Funkhouser. The Princeton shape benchmark. In *Shape Modeling International*, June 2004.

Behjat Siddiquie, Rogerio S Feris, and Larry S Davis. Image ranking and retrieval based on multi-attribute queries. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2014.

Richa Singh, Akshay Agarwal, Maneet Singh, Shruti Nagpal, and Mayank Vatsa. On the robustness of face recognition algorithms against attacks and bias. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Neural Information Processing Systems (NeurIPS)*, 2017.

Kihyuk Sohn. Improved deep metric learning with multi-class n-pair loss objective. In *Neural Information Processing Systems (NeurIPS)*, 2016.

Hyun Oh Song, Yu Xiang, Stefanie Jegelka, and Silvio Savarese. Deep metric learning via lifted structured feature embedding. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

George B. Sproles. Analyzing fashion life cycles—principles and perspectives. *Journal of Marketing*, 45(4), 1981.

Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *Journal of Machine Learning Research (JMLR)*, 15(1), 2014.

Pierre Stock and Moustapha Cisse. Convnets and imagenet beyond accuracy: Understanding mistakes and uncovering biases. In *European Conference on Computer Vision (ECCV)*, 2018.

Gjorgji Strezoski, Nanne van Noord, and Marcel Worring. Many task learning with task routing. In *International Conference on Computer Vision (ICCV)*, 2019.

Hang Su, Subhransu Maji, Evangelos Kalogerakis, and Erik G. Learned-Miller. Multi-view convolutional neural networks for 3d shape recognition. In *International Conference on Computer Vision (ICCV)*, 2015.

Ilya Sutskever, James Martens, George Dahl, and Geoffrey Hinton. On the importance

of initialization and momentum in deep learning. In *International Conference on Machine Learning (ICML)*, 2013.

Flora Ponjou Tasse and Neil Dodgson. Shape2vec: Semantic-based descriptors for 3d shapes, sketches and images. *ACM Transactions on Graphics (TOG)*, 2016.

William Thong and Cees G. M. Snoek. Bias-awareness for zero-shot learning the seen and unseen. In *British Machine Vision Conference (BMVC)*, 2020.

William Thong and Cees G. M. Snoek. Feature and label embedding spaces matter in addressing image classifier bias. In *British Machine Vision Conference (BMVC)*, 2021.

William Thong and Cees G. M. Snoek. Diversely-supervised visual product search. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 18(1), 2022.

William Thong, Pascal Mettes, and Cees G. M. Snoek. Open cross-domain visual search. *Computer Vision and Image Understanding (CVIU)*, 200, 2020.

William Thong, Jose Costa Pereira, Ale Leonardis, Sarah Parisot, and Steven McDonagh. Content-diverse comparisons improve image quality assessment learning. In *submission*, 2022.

Antonio Torralba and Alexei A Efros. Unbiased look at dataset bias. In *Computer Vision and Pattern Recognition (CVPR)*, 2011.

Mehmet Ozgur Turkoglu, William Thong, Luuk Spreeuwers, and Berkay Kicanaoglu. A layer-based sequential framework for scene generation with gans. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2019.

Eric Tzeng, Judy Hoffman, Trevor Darrell, and Kate Saenko. Simultaneous deep transfer across domains and tasks. In *International Conference on Computer Vision (ICCV)*, 2015.

Douwe van den Brink, Gaby Odekerken-Schröder, and Pieter Pauwels. The effect of strategic and tactical cause-related marketing on consumers' brand loyalty. *Journal of Consumer Marketing*, 2006.

Andreas Veit, Balazs Kovacs, Sean Bell, Julian McAuley, Kavita Bala, and Serge Belongie. Learning visual clothing style with heterogeneous dyadic co-occurrences. In *International Conference on Computer Vision (ICCV)*, 2015.

Andreas Veit, Serge Belongie, and Theofanis Karaletsos. Conditional similarity networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval - an empirical odyssey. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.

Angelina Wang, Arvind Narayanan, and Olga Russakovsky. REVISE: A tool for measuring and mitigating bias in visual datasets. In *European Conference on Computer Vision (ECCV)*, 2020a.

Fang Wang, Le Kang, and Yi Li. Sketch-based 3d shape retrieval using convolutional neural networks. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Hao Wang, Yitong Wang, Zheng Zhou, Xing Ji, Dihong Gong, Jingchao Zhou, Zhifeng Li, and Wei Liu. Cosface: Large margin cosine loss for deep face recognition. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Mei Wang, Weihong Deng, Jiani Hu, Xunqiang Tao, and Yaohai Huang. Racial faces in the wild: Reducing racial bias by information maximization adaptation network. In *International Conference on Computer Vision (ICCV)*, 2019a.

Tianlu Wang, Jieyu Zhao, Mark Yatskar, Kai-Wei Chang, and Vicente Ordonez. Balanced datasets are not enough: Estimating and mitigating gender bias in deep image representations. In *International Conference on Computer Vision (ICCV)*, 2019b.

Yu-Xiong Wang and Martial Hebert. Learning to learn: Model regression networks for easy small sample learning. In *European Conference on Computer Vision (ECCV)*, 2016.

Zeyu Wang, Klint Qinami, Ioannis Christos Karakozis, Kyle Genova, Prem Nair, Kenji Hata, and Olga Russakovsky. Towards fairness in visual recognition: Effective strategies for bias mitigation. In *Computer Vision and Pattern Recognition (CVPR)*, 2020b.

Zihao Wang, Xihui Liu, Hongsheng Li, Lu Sheng, Junjie Yan, Xiaogang Wang, and Jing Shao. Camp: Cross-modal adaptive message passing for text-image retrieval. In *International Conference on Computer Vision (ICCV)*, 2019c.

Kilian Q Weinberger and Lawrence K Saul. Distance metric learning for large margin nearest neighbor classification. *Journal of Machine Learning Research (JMLR)*, 10, 2009.

Yandong Wen, Kaipeng Zhang, Zhifeng Li, and Yu Qiao. A discriminative feature learning approach for deep face recognition. In *European Conference on Computer Vision (ECCV)*, 2016.

Michael J. Wilber, Chen Fang, Hailin Jin, Aaron Hertzmann, John Collomosse, and Serge Belongie. Bam! the behance artistic media dataset for recognition beyond photography. In *International Conference on Computer Vision (ICCV)*, 2017.

Adina Williams, Nikita Nangia, and Samuel R Bowman. A broad-coverage challenge corpus for sentence understanding through inference. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2018.

Ludwig Wittgenstein. *Philosophical Investigations*. Basil Blackwell, Oxford, 1953.

Zhirong Wu, Alexei A Efros, and Stella X Yu. Improving generalization via scalable neighborhood component analysis. In *European Conference on Computer Vision (ECCV)*, 2018.

Yongqin Xian, Zeynep Akata, Gaurav Sharma, Quynh Nguyen, Matthias Hein, and Bernt Schiele. Latent embeddings for zero-shot classification. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Yongqin Xian, Christoph H Lampert, Bernt Schiele, and Zeynep Akata. Zero-shot learning-a comprehensive evaluation of the good, the bad and the ugly. *Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 2018a.

Yongqin Xian, Tobias Lorenz, Bernt Schiele, and Zeynep Akata. Feature generating networks for zero-shot learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2018b.

Yongqin Xian, Saurabh Sharma, Bernt Schiele, and Zeynep Akata. f-vaegan-d2: A feature generating framework for any-shot learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Jin Xie, Guoxian Dai, Fan Zhu, and Yi Fang. Learning barycentric representations of 3d shapes for sketch-based 3d shape retrieval. In *Computer Vision and Pattern Recognition (CVPR)*, 2017.

Ruijia Xu, Ziliang Chen, Wangmeng Zuo, Junjie Yan, and Liang Lin. Deep cocktail network: Multi-source unsupervised domain adaptation with category shift. In *Computer Vision and Pattern Recognition (CVPR)*, 2018.

Fan Yang, Ajinkya Kale, Yury Bubnov, Leon Stein, Qiaosong Wang, M Hadi Kiapour, and Robinson Piramuthu. Visual search at ebay. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2017.

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 *Fairness, Accountability, and Transparency (FAccT)*, 2020.

Linjie Yang, Ping Luo, Chen Change Loy, and Xiaoou Tang. A large-scale car dataset for fine-grained categorization and verification. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

Linwei Ye, Zhi Liu, and Yang Wang. Learning semantic segmentation with diverse supervision. In *Winter Conference on Applications of Computer Vision (WACV)*, 2018.

Sasi Kiran Yelamarthi, Shiva Krishna Reddy, Ashish Mishra, and Anurag Mittal. A zero-shot framework for sketch based image retrieval. In *European Conference on Computer Vision (ECCV)*, 2018.

Xi Yin, Xiang Yu, Kihyuk Sohn, Xiaoming Liu, and Manmohan Chandraker. Feature transfer learning for face recognition with under-represented data. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Aron Yu and Kristen Grauman. Fine-grained visual comparisons with local learning. In *Computer Vision and Pattern Recognition (CVPR)*, 2014.

Roshanak Zakizadeh, Michele Sasdelli, Yu Qian, and Eduard Vazquez. Improving the annotation of deepfashion images for fine-grained attribute recognition.

arXiv:1807.11674, 2018.

Andrew Zhai and Hao-Yu Wu. Making classification competitive for deep metric learning. In *British Machine Vision Conference (BMVC)*, 2019.

Andrew Zhai, Dmitry Kislyuk, Yushi Jing, Michael Feng, Eric Tzeng, Jeff Donahue, Yue Li Du, and Trevor Darrell. Visual discovery at pinterest. In *WWW*, 2017.

Andrew Zhai, Hao-Yu Wu, Eric Tzeng, Dong Huk Park, and Charles Rosenberg. Learning a unified embedding for visual search at pinterest. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2019.

Brian Hu Zhang, Blake Lemoine, and Margaret Mitchell. Mitigating unwanted biases with adversarial learning. In *Artificial Intelligence, Ethics, and Society (AIES)*, 2018a.

Hua Zhang, Si Liu, Changqing Zhang, Wenqi Ren, Rui Wang, and Xiaochun Cao. Sketchnet: Sketch classification with web images. In *Computer Vision and Pattern Recognition (CVPR)*, 2016.

Yanhao Zhang, Pan Pan, Yun Zheng, Kang Zhao, Yingya Zhang, Xiaofeng Ren, and Rong Jin. Visual search at alibaba. In *Conference on Knowledge Discovery and Data Mining (KDD)*, 2018b.

Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan. Memory-augmented attribute manipulation networks for interactive fashion search. In *Computer Vision and Pattern Recognition (CVPR)*, 2017a.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2017b.

Jieyu Zhao, Yichao Zhou, Zeyu Li, Wei Wang, and Kai-Wei Chang. Learning gender-neutral word embeddings. In *Empirical Methods in Natural Language Processing (EMNLP)*, 2018a.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Ryan Cotterell, Vicente Ordonez, and Kai-Wei Chang. Gender bias in contextualized word embeddings. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

Xiangyun Zhao, Haoxiang Li, Xiaohui Shen, Xiaodan Liang, and Ying Wu. A modulation module for multi-task learning with applications in image retrieval. In *European Conference on Computer Vision (ECCV)*, 2018b.

Shizhan Zhu, Raquel Urtasun, Sanja Fidler, Dahua Lin, and Chen Change Loy. Be your own prada: Fashion synthesis with structural coherence. In *International Conference on Computer Vision (ICCV)*, 2017.

Junbao Zhuo, Shuhui Wang, Shuhao Cui, and Qingming Huang. Unsupervised open domain recognition by semantic discrepancy minimization. In *Computer Vision and Pattern Recognition (CVPR)*, 2019.

Samenvatting

Dit proefschrift onderzoekt hoe visuele overeenkomsten helpen bij het leren van modellen die robuust zijn tegen vooroordelen voor computer vision taken. Modellen moeten zich constant kunnen aanpassen aan nieuwe en veranderende omgevingen zonder bevooroordeeld te zijn door wat ze tijdens het trainen hebben gezien. In dit proefschrift concentreren we ons op de onderzoeksvraag: *hoe leer je visuele overeenkomsten die robuust zijn tegen vooroordelen?* We onderzoeken deze vraag door middel van de meerdere facetten van vooroordelen door ze aan te pakken met een gemeenschappelijk thema over visuele overeenkomsten. We beginnen met een categorisering over meerdere domeinen, onderzoeken vervolgens het vermogen om combinaties van zichtbare en onzichtbare attributen te verkrijgen, gevolgd door de studie van het vertrouwen van beeld classificatie modellen tegenover geziene klassen, en ten slotte de identificatie en beperking van ongunstige voorspellingen in beeld classificaties.

Hoofdstuk 2 behandelt visueel zoeken tussen domeinen, waarbij visuele zoekopdrachten categorie voorbeelden ophalen uit een ander domein. Wanneer we bijvoorbeeld een vliegtuig willen schetsen en foto's van vliegtuigen verkrijgen. Ondanks aanzienlijke vooruitgang vindt de zoektocht plaats in een gesloten setting tussen twee vooraf gedefinieerde domeinen. In dit hoofdstuk maken we de stap naar een open setting waar meerdere visuele domeinen beschikbaar zijn. Dit vertaalt zich met name in een zoekopdracht tussen elk paar domeinen, vanuit een combinatie van domeinen of binnen meerdere domeinen. We introduceren een eenvoudige, maar effectieve aanpak. We formuleren de zoekopdracht als een mapping van elk visueel domein naar een gemeenschappelijke semantische ruimte, waar categorieën worden weergegeven door hyper sferische prototypes. Open visueel zoeken tussen domeinen wordt vervolgens uitgevoerd door te zoeken in de gemeenschappelijke semantische ruimte, ongeacht welke domeinen als bron worden gebruikt of als doel. Domeinen worden gecombineerd in de gemeenschappelijke ruimte om tegelijkertijd vanuit of binnen meerdere domeinen te zoeken. Het afzonderlijk trainen van elke domeinspecifieke mapping-functie maakt een efficiënte

schaling naar een willekeurig aantal domeinen mogelijk zonder de zoekprestaties te beïnvloeden. We illustreren empirisch ons vermogen om open visuele zoekopdrachten tussen domeinen uit te voeren in drie verschillende scenario's. Onze aanpak is ook concurrerend met betrekking tot bestaande gesloten settings, waar we state-of-the-art resultaten verkrijgen op verschillende benchmarks voor drie op schetsen gebaseerde zoektaken.

Hoofdstuk 3 introduceert een diverse gesuperviseerde visuele zoekopdracht naar een product, waarbij queries een diverse reeks labels specificeren waarnaar moet worden gezocht. Waar eerdere werken zich richtten op het afzonderlijk weergeven van attribuut-, instantie- of categorielabels, beschouwen wij ze samen om een gevarieerde set labels te creëren voor het visueel beschrijven van producten. We leren een inbedding van het gesuperviseerde signaal van elk label om hun onderlinge relaties te encoderen. Eenmaal getraind, heeft elk label een bijbehorende visuele representatie in de inbeddingsruimte, wat een samenvoeging is van geselecteerde items uit de trainingsset. Tijdens het zoeken halen samengestelde query-representaties afbeeldingen op die overeenkomen met een specifieke set van diverse labels. We vormen samengestelde query representaties door het gemiddelde te nemen van de geaggregeerde representaties van elk diverse label in de specifieke set. Voor evaluatie breiden we bestaande product datasets van auto's en kleding uit met een diverse verzameling aan labels. Experimenten tonen de voordelen aan van onze inbedding voor het visueel zoeken naar producten onder toezicht van verschillende soorten in zichtbare en onzichtbare product combinaties en voor het ontdekken van product ontwerpstijlen.

Hoofdstuk 4 gaat in op het probleem van gegeneraliseerd zero-shot leren, dat tot doel heeft input van zowel zichtbare als onzichtbare klassen te herkennen. Toch zijn bestaande methoden vaak bevooroordeeld ten opzichte van de klassen die tijdens de training worden gezien. In dit hoofdstuk proberen we deze bevooroordeeldheid te verminderen. We stellen een vooroordeel-bewuste leerder voor om inputs toe te wijzen aan een semantische inbeddingsruimte voor gegeneraliseerd zero-shot leren. Tijdens de training leert het model terug te vallen naar echte klasse-prototypes in de inbeddingsruimte met temperatuurschaling, terwijl een op marges gebaseerde bidirectionele entropie term zichtbare en onzichtbare waarschijnlijkheden regulariseert. Het vertrouwen op een semantische inbeddingsruimte met reële waarde biedt een veelzijdige benadering, aangezien het model kan werken op verschillende soorten semantische informatie voor zowel zichtbare als onzichtbare klassen. Er worden experimenten uitgevoerd op vier benchmarks voor gegeneraliseerd zero-shot leren en deze demonstreren de voordelen van de voorgestelde vooroordeel-bewuste classificatie functie, zowel als een op zichzelf staande methode als in combinatie met gegenereerde features.

Hoofdstuk 5 gaat in op de vooroordelen van beeldclassificatie, met een focus op zowel feature- als label inbeddingsruimten. Eerdere werken hebben aangetoond dat valse correlaties van beschermende kenmerken, zoals leeftijd, gender of huidskleur, ongunstige beslissingen kunnen veroorzaken. Om mogelijke schade

in evenwicht te brengen, is er een groeiende behoefte om vertekening door beeldclassificatie te identificeren en te verminderen. Ten eerste identificeren we in de feature-ruimte een vooroordeel-richting. We berekenen klasse prototypes van elke beschermende attribuutwaarde voor elke klasse en onthullen een bestaande subruimte die de maximale variantie van de bias vastlegt. Ten tweede verminderen we vooroordelen door beeldinvoer toe te wijzen aan inbeddingsruimten van labels. Elke waarde van het beschermde attribuut heeft zijn projectiekop waarin klassen worden ingebed via een latente vector representatie in plaats van een gewone one-hot-codering. Eenmaal getraind, verminderen we in de featureruimte het vooroordeel-effect verder door de richting ervan te verwijderen. Evaluatie van bevooroordeelde afbeeldingsdatasets, voor classificaties met meerdere klassen, meerdere labels en binaire bestanden, toont de effectiviteit aan van het aanpakken van zowel feature- als labelinbeddingsruimten bij het verbeteren van de eerlijkheid van de classificatie voorspellingen, terwijl de classificatie prestaties behouden blijven.

Acknowledgments

First and foremost gratitude is due to my promotor Prof. Cees Snoek. Cees, this PhD journey was a bumpy ride. Not everything went the way we have planned. Thank you for mentoring, supporting and guiding me during this PhD journey. Strong will, grit, resilience, patience, and solid work made it successful in the end. A simple recipe, easier said than done. Our discussions helped me to organize thoughts, motivate messages, improve executions, sharpen writing, and give praise to others. You gave me a unique opportunity to discover and pursue research in computer vision, and trained me to become an independent and autonomous researcher in the field. I have learned invaluable lessons alongside you throughout all these years, and they will definitely be remembered!

I would like to also extend my gratitude to my co-promotor Prof. Arnold Smeulders. I still vividly remember our initial discussions on beauty and how it could be approached from a computer vision perspective. While this thesis omits the concept of *beauty*, it incorporates the concept of *similarities* which I believe is key to make a step towards understanding *beauty*. Arnold, I have appreciated your sharp and direct questions over the years, as they helped shape my way of thinking. Thank you for letting me part of the lab you have built, which fosters human interactions and helps its young and eager researchers grow.

I am grateful to my PhD defense committee members Prof. Marcel Worring, Prof. Maarten de Rijke, Dr. Sennay Ghebreab, Prof. Nicu Sebe, and Dr. Thomas Mensink. It is a pleasure and honor to have you all in my committee.

I would like to extend my gratitude to Dr. Thomas Mensink. Thomas, it was always a pleasure to discuss metric learning methods, loss functions, or research in general with you. A sincere thank you for accepting to be part of my PhD committee. Over the years, I was also fortunate to interact with and receive constructive feedback from several faculty members at the Informatics Institute, a special thank you goes to Prof. Zeynep Akata, Dr. Efstratios Gavves, Prof. Theo Gevers, Dr. Herke van Hoof among others.

A PhD journey usually involves to spend time in other pastures. Thank you

to Prof. Damian Borth for welcoming me in his research lab in Kaiserslautern. I appreciated our discussions about how to approach research, and it was a pleasure to spend time with Federico Raue, Sebastian Palacio, and Marco Schreyer. I am also indebted to Dr. Steven McDonagh for hosting me remotely in his research team in London. I am glad we finally managed to meet physically after meeting each other virtually for the first time two years ago, and spending more than six months collaborating together virtually. Steven, thank you for being such a great mentor, I hope to be the same with my mentees in the future. It was also a pleasure to work during my visit with Dr. Jose Costa Pereira, Dr. Sarah Parisot, and Prof. Ales Leonardis. Thank you all for the fruitful discussions.

No PhD journey would be successful in the lab without Dennis Koelma. Our daily lunch breaks were without any doubt one of the highlights of my PhD. Whether it was a bad day or a good day, I would try not to miss an opportunity to join you for lunch. Our discussions were always very insightful, full of advice, and always joyful. Dennis, I learned a lot with you and I am indebted to you. Another key person, full of advice and always ready to help, is Virginie Mes. Thank you for making my PhD journey run smoother.

Witnessing the "old guard" grow from their PhD or postdoc years to a successful career in research was a true privilege: Andrew Brown, Spencer Cappallo, Amir Ghodrati, Deepak Gupta, Jörn Jacobsen, Zhenyang Li, Pascal Mettes, Nanne van Noord, Stevan Rudinac, Ran Tao and Shihan Wang. Thank you all for the time we spent together, you are a great source of inspiration and were always present when I needed advice. Pascal, our collaboration was short, refreshing, and very successful! I hope we will repeat that in the future.

A famous saying states that we forge friendship through hardship. As PhD students, we all share the same journey. We are all successful but sometimes we go through bad times, and sharing our experience helps to keep the boat afloat. As a result, a PhD journey would not be same if one doesn't spend the time to wander around office labs and get to know his labmates.

In that regard, I would like to first express my sincere gratitude to Mert Kılıçkaya. It took us about a year to get to know each other, but what a blast it was afterwards! Whether we were down, upset, tired or angry about our research, we always supported each other. And we still are. Humor and derision were clearly keys in making our PhD journey much more pleasant, delightful and simply just fun. That being said, our friendship goes beyond research and I always loved our endless discussions on art, music, cinema, literature, society, politics, food, and many other topics. Our daily coffee walks in the midst of the pandemic were the ideal setting to foster such discussions, and were clearly one of the highlights of my PhD. Getting to know you and spend time with you sparked joy!

I would also like to especially thank Sarah Ibrahimi. Sarah, thank you so much for your energy and passion. I highly appreciated all our discussions, be it about research or beyond. I still remember our open-air cinema viewings, IDFA sessions, theater and dance shows, and many others. Without you, the social

aspect of the PhD, and the lab in general, would have been completely different. And, thank you again for translating the summary of this thesis.

Another token of appreciation goes to Zenglin Shi, Jia-Hong Huang, Tao Hu and David Zhang. Thank you guys for all the honest and sincere discussions about research and life in general. I miss our weekly Friday dinners at a Chinese restaurant where I would get to learn about the Chinese culture and language. Promise, I will improve my Mandarin in the next years! Zenglin, it is a great pleasure to get to know you and I always appreciate our rich research discussion. David, thank you for handling the logistics of my PhD booklets.

I am also grateful to have met Anıl Başlamışlı, Berkay Kıcanaoğlu and Noureldien Hussein. Anıl, we got to know each other more at a summer school, feels like yesterday, and since then I always appreciate our discussions and time together. Berkay, our research collaboration was fun, we should definitely repeat it. Thank you both for your friendship, I miss our gatherings at Anıl's place! Nour, many thanks for all your advice and motivational thoughts throughout the PhD.

I would also like to extend my gratitude to other lab members who played a large role in making my PhD journey much more pleasant: Devanshu Arya (the one whose name should be said fully), Shuo Chen (the one who evolved the most), Yunlu Chen (the gourmet who would travel very far for good food), Kirill Gavrilyuk (the other F1 fan), Inske Groenen (the one who I could speak French with), Sadaf Gulshad (the Asian culture fan), Gjorgji Strezoski (the multitasker in life and research), Jiaojiao Zhao (the best cook in the lab), Riaan Zoetmulder (the Dutchman making fun of French people). As well as to the rest of the lab and beyond: Mehmet Altinkaya, Hazel Doughty, Mina Ghadimi Atigh, Shi Hu, Shuai Liao, Ana Lucic, Artem Moskalev, Peter O'Connor, Changyong Oh, Adeel Pervez, Elise van der Pol, Tom Runia, Tom van Sonsbeek, Ivan Sosnovik, Fida Thoker, Qi Wang, Maurice Weiler, Pengwan Yang, and Yunhua Zhang.

I would also like to sincerely thank the members of the computer vision lab: Hanan ElNaghy, Hoàng-Ân Lê, and Minh Ngô. We were teaching assistants for the computer vision courses together, and I always liked dropping by your office to steal some candies, share some crêpes, borrow an e-identifier, or simply spend a good time together. Thank you for making me feel at home in your lab!

Et enfin, je voudrais remercier tous mes amis de longue date, ainsi que ma famille, qui se reconnaîtront sans aucune peine. Ils ont suivris de près, parfois de trop près, mes tribulations dans le monde de la recherche et tout au long de mon doctorat. On ne s'est pas souvent vus, voire même pas du tout pour certains, mais ce fut un privilège et un honneur que de vous avoir à mes côtés. Un doctorat est à la fois une aventure scientifique mais aussi personnelle, et je n'en sors que grandi. Merci de votre soutien, de votre écoute, ainsi que de vos conseils tout au long de cette aventure.

William Thong
August 2022