# Effectively Leveraging Attributes for Visual Similarity

Samarth Mishra*[1] Zhongping Zhang*[1] Yuan Shen[2] Ranjitha Kumar[2]
Venkatesh Saligrama[1] Bryan Plummer[1]
[1]Boston University [2]University of Illinois at Urbana Champaign

## Abstract

*Measuring similarity between two images often requires performing complex reasoning along different axes (*e.g.*, color, texture, or shape). Insights into what might be important for measuring similarity can be provided by annotated attributes. Prior work tends to view these annotations as complete, resulting in them using a simplistic approach of predicting attributes on single images, which are, in turn, used to measure similarity. However, it is impractical for a dataset to fully annotate every attribute that may be important. Thus, only representing images based on these incomplete annotations may miss out on key information. To address this issue, we propose the Pairwise Attribute-informed similarity Network (PAN), which breaks similarity learning into capturing similarity conditions and relevance scores from a joint representation of two images. This enables our model to identify that two images contain the same attribute, but can have it deemed irrelevant (*e.g.*, due to fine-grained differences between them) and ignored for measuring similarity between the two images. Notably, while prior methods of using attribute annotations are often unable to outperform prior art, PAN obtains a 4-9% improvement on compatibility prediction between clothing items on Polyvore Outfits and a 5% gain on few shot classification of images using Caltech-UCSD Birds (CUB), and over 1% boost to Recall@1 on In-Shop Clothes Retrieval.*

## 1. Introduction

Learning similarity metrics between images is a central problem in computer vision with wide-ranging applications such as face recognition [14, 23], image retrieval [7, 16, 36], prototype based few shot image classification [11, 24, 26, 31], continual learning of image classification [2, 22, 25], and fashion compatibility or recommendation [5, 27, 28, 29, 30, 39]. There has been a recent trend of learning these metrics by decomposing the problem into multiple axes of similarity or *similarity conditions*, which has improved performance on a variety of tasks [9, 13, 17, 18, 19, 27, 28, 29]. Generally speaking, methods that automatically learn what these conditions represent [18, 27] have reported

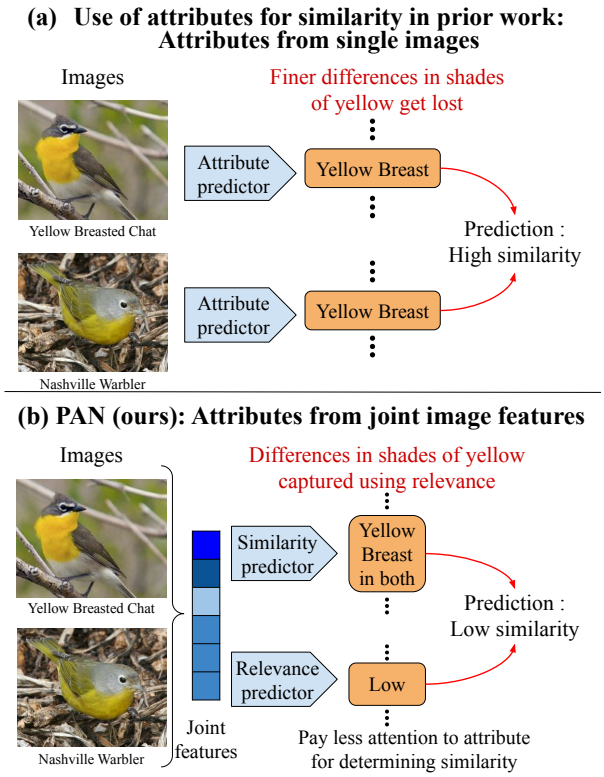


Figure 1: In prior work (*e.g.* [12, 37, 28, 19]), shown in (a), attributes used for image similarity are typically predicted for each image and then are used as input to the image similarity model. However, this can result in loss of important information about how attributes are expressed (*e.g.*, different shades of the attribute *yellow breast*). Thus, in our work, shown in (b), we avoid this loss of information by using a joint representation of the two images to compute multiple disentangled similarity scores, each corresponding to an attribute, and relevances of each similarity score in the final similarity prediction. This allows for more fine-grained reasoning about different attribute manifestations, boosting performance.

*Indicates equal contribution

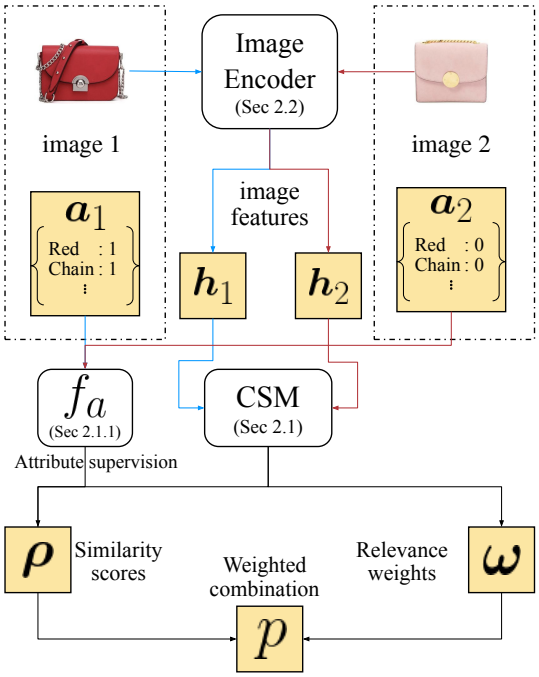


Figure 2: **PAN overview.** Given a pair of images we generate feature vectors for each using an image encoder. The image features are then fed into the Concept-conditioned Similarity Module (CSM) that generates a set of concept similarity scores and corresponding relevance weights.w The final similarity score $p \in [0, 1]$ is produced using a weighted combination of the similarity conditions and their relevance. Different colored lines (blue, pink) represent information flow pertaining to individual images.

better performance than those that predefine this knowledge using information like labeled image attributes and item categories [13, 29, 28, 19].

In this paper, we introduce a Pairwise Attribute-informed similarity Network (PAN) that breaks this trend. Prior approaches using attributes predict their presence on single images (*e.g.* [12, 37, 28, 19]), subsequently using these predictions for predicting similarity. This results in an information loss about unannotated fine-grained attributes, such as information loss about the different shades of the attribute yellow breast in the case of the two birds in Fig 1(a), consequently leading to poorer similarity predictions. Our PAN model avoids this issue by first comparing images in a feature space rather than attribute space, as illustrated in Fig 1(b). Using the joint image features it then predicts both a similarity score and a relevance for different similarity conditions defined by the attributes. Even when the similarity score may coarsely indicate that the two images are similar since they have the same attribute, the model can pick up on finer attribute differences and decide that the mere presence of the same attribute is of low relevance to a positive similarity prediction. As our experiments will show, this difference can make a dramatic impact on the performance of the learned image similarity model.

To leverage attribute annotations in PAN we must first convert the single-image annotations to labels representing a pair of images. The best mechanism for this conversion is often task-dependant. For example, for the birds in Fig 1 images containing the same bird should match attributes. However, in fashion compatibility, where two images are deemed similar if they complement each other when worn together in an outfit, image pairs with different attributes (*e.g.*, black and orange) can indicate they are highly compatible. In our experiments we treat the attribute label conversion method as a hyperparameter setting and select the approach that performs the best (*e.g.*, select between making the label to "1" if both images contain the attribute vs. making the label "1" if either image contains the attribute). Notably, PAN obtains a 4-9% improvement on fashion compatibility on Polyvore Outfits [28], a 5% gain on few shot classification using Caltech-UCSD Birds (CUB) [32], and over 1% boost to Recall@1 on In-Shop Clothes Retrieval [15].

## 2. Pairwise Attribute-informed similarity Network (PAN)

Given images $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, PAN first uses an Image encoder (see Sec 2.2) to get features $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$. These features are fed into our Concept-conditioned Similarity Module (see Sec 2.1) to compute similarity score $p \in [0, 1]$.

### 2.1. Concept-conditioned Similarity Module

Given features $\boldsymbol{h}_i, \boldsymbol{h}_j \in \mathbb{R}^d$ for two images, our Concept-conditioned Similarity Module (CSM) generates a set of $M$ similarity scores $\boldsymbol{\rho} = [\rho_1, \ldots, \rho_M] \in \mathbb{R}^M$ and corresponding relevance weights $\boldsymbol{\omega} = [\omega_1, \ldots, \omega_M] \in \mathbb{R}^M$ which represent the importance of each similarity condition:

$$\boldsymbol{\rho} = \sigma\left(\boldsymbol{W}_1^\top |\boldsymbol{h}_i - \boldsymbol{h}_j| + \boldsymbol{b}_1\right) \tag{1}$$

$$\boldsymbol{\omega} = \text{softmax}\left(\boldsymbol{W}_2^\top |\boldsymbol{h}_i - \boldsymbol{h}_j| + \boldsymbol{b}_2\right) \tag{2}$$

where $M$ is the number of distinct similarity conditions, $|\cdot|$ represents an element-wise absolute value, and $\sigma(\cdot)$ an element-wise sigmoid function. $\boldsymbol{W}_1, \boldsymbol{W}_2 \in \mathbb{R}^{d \times M}$, and $\boldsymbol{b}_1, \boldsymbol{b}_2 \in \mathbb{R}^M$ are learnable parameters. Note that $\boldsymbol{\rho}$ is supervised using attribute labels, but the relevance scores $\boldsymbol{\omega}$ are treated as latent variables and are automatically learned. The final similarity score $p \in [0, 1]$ is calculated as the sum of similarity conditions weighed by their relevance, *i.e.*,

$$p = \sum_{m=1}^{M} \rho_m \omega_m = \boldsymbol{\rho}^\top \boldsymbol{\omega}. \tag{3}$$

#### 2.1.1 Defining Similarity Conditions

Depending on the availability of labelled attributes for the images, we can choose to supervise similarity conditions to give them semantic meaning. This choice results in two kinds of similarity conditions as described below:

**Unsupervised similarity conditions.** Similarity conditions are treated as latent variables as done in [27], except we use a joint representation of the two images, unlike in [27] conditions were computed per-image.

**Supervised similarity conditions.** Rather than treating each similarity condition as a latent variable, supervised similarity conditions are trained to reflect a specific concept. Since attribute annotations are defined per image, and we predict attributes based off a joint representation, we convert these labels to represent both images, as described next.

Suppose the images have $M$ labelled binary attributes. Each image $i$ is then accompanied with an $M$ dimensional vector $\boldsymbol{a}_i \in \{0,1\}^M$. For a pair of images $i$ and $j$, we can use a function $f_a : \{0,1\} \times \{0,1\} \rightarrow \{0,1\}$ to get an $M$ dimensional vector $\boldsymbol{a}_{i,j} = f_a(\boldsymbol{a}_i, \boldsymbol{a}_j)$. Elements of $\boldsymbol{a}_{i,j}$ can then be used as labels for supervising the similarity conditions in the model output scores $\boldsymbol{\rho}$. If there are missing entries in $\boldsymbol{a}_{i,j}$ because of missing attribute labels, these can be handled by zeroing out the loss resulting from them using a binary mask over the indices of $\boldsymbol{a}_{i,j}$.

We create labels for image pairs by selecting $f_a$ from the best performing logical function. These functions have clear semantic meanings, *e.g.*, in $f_a = logical\ AND$ similarity condition $\boldsymbol{\rho}_i$ predicts if both images has an attribute, whereas $OR$ represents if either image does.

### 2.2. Types of Image Encoder

As mentioned previously, the image encoder generates a lower dimensional feature representation $\boldsymbol{h}$ for an image $\boldsymbol{x}$. We experiment with three different image encoders.

**Convolutional Network.** Unless specified otherwise, we use a simple convolutional neural network (CNN), specifically a ResNet [8] to represent images.

**Graph Encoder (GE) [5].** This encoder uses a graph convolutional network (GCN) to compute contextualized similarly scores between images.

**ProxyNCA++ [35].** This encoder learns a distance metric between images based on learning proxy feature representations for each image class, which turns similarity learning into a classification task during training.

### 2.3. Model Objective and Training

The final objective function on a pair of images $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ is then defined as:

$$\mathcal{L}(\boldsymbol{x}_i, \boldsymbol{x}_j, e_{i,j}, \boldsymbol{a}_{i,j}) = \mathcal{L}^{BCE}(e_{i,j}, p) + \lambda \mathcal{L}_{el}^{BCE}(\boldsymbol{a}_{i,j}, \boldsymbol{\rho}), \tag{4}$$

where $\lambda$ is tunable hyperparameter, $e_{i,j} \in \{0,1\}$ is the ground truth similarity label between images $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$, $\mathcal{L}^{BCE}$ is the binary cross-entropy loss and $\mathcal{L}_{el}^{BCE}$ is the mean element-wise binary cross-entropy. Note that when there are no supervision attributes, the second term in Equation 4 is 0. For training, an equal number of positive and negative pairs are sampled randomly from the training split and the model is trained to predict similarity between them.

## 3. Datasets and Tasks

**Polyvore Outfits [28]** contains 53K outfits (sets of fashion items) for training, 5K for validation and 10K for testing. We use the 205 sparsely annotated attributes from [20] as labels for supervising similarity conditions. Performance is evaluated over two tasks: in fill-in-the-blank (FITB) outfit completion a model is evaluated by whether it correctly selects the best item from a set of choices to complete an outfit, whereas in outfit compatibility we rank complete outfits using the area under a receiver operating characteristic curve (AUC). Following [28, 5], outfit scores are computed by averaging the similarity prediction over all pairs of items in the outfit. There are 10K FITB questions and 10K each positive and negative samples for outfit compatibility (20K total) in the test split. We also created new *resampled* split by adding more choices to FITB (10 choices rather than 4), and creating negative outfits by doing partial random replacements in outfits (the original splits replaced all items).

**CUB-200-2011 [32]** consists of 200 classes and a total of 11,768 images of birds. We use the split provided by [3] for our experiments, containing 100 base classes, 50 validation and 50 novel classes. The CUB dataset also has 312 fine-grained binary attributes labelled for each image. We use the 5-way 5-shot classification task for evaluation. Reported accuracies are averaged over 3 training runs from different random initializations accompanied by 95% confidence intervals. A test *episode* consists of a random sample of 5 classes and 5 support images from the 50 classes in the novel split of the dataset. 16 query images, distinct from support images, are also sampled for each of these 5 classes. The accuracy for an episode is the 5-way accuracy of a classifier over the 16 x 5 = 80 query images. A few shot learning model is evaluated using its average classification accuracy over 600 randomly generated test episodes.

**In-Shop Retrieval [15]** contains 52,712 images of clothes from 11,967 classes. There are 14,218 query images and 12,612 gallery images for testing. Given a query image, the task is to retrieve an image of the same item from the gallery set. Note that the query and gallery sets do not overlap with the training set. There are 463 attributes of clothes in total, we use these attribute labels for our PAN-Supervised model. Methods are ranked based on Recall@1.

## 4. Results

Table 1, Table 2, and Table 3 compare the best settings (encoder, number of unsupervised similarity conditions, attribute combination $f_a$, *etc*.) used by our model to representative state-of-the-art results reported in prior work on Polyvore Outfits, CUB, and In-Shop Retrieval, respectively. As shown in Table 1 we obtain a 4% better FITB accuracy and 9% AUC boost over the state-of-the-art on the fashion compatibility task using our more challenging resampled test set for both tasks, while also increasing FITB accuracy by 8% on the original split. Similarly, in Table 2 and Table 3 we observe a 5% and 1% performance improvement over the state-of-the-art on fine-grained few shot classification and In-Shop Retrieval, respectively. Improvement over the diverse set of tasks demonstrates PAN's ability to generalize. Our model can also be useful when no supervision is provided, as our PAN-Unsupervised model obtains a 3-4% gain over prior work on Polyvore Outfits and CUB, while also boosting performance on In-Shop Retrieval. Note that fashion compatibility benefited from using a graph image encoder (GE), while few-shot classification reported best performance with a CNN encoder. Also, we found that $f_a = OR$ performed best for tasks on Polyvore Outfits and CUB, while $XNOR$ performed best for In-Shop Retrieval.

Table 1, Table 2, and Table 3 also provides two baselines that leverage attributes for similarity learning. In "X + Attr. Multitask" we use a hard parameter sharing multitask approach [1], where the image encoder is shared, but have separate output heads for each of the two tasks (one being attribute classification, the other similarity prediction). "Attr. Similarity" predicts attributes for each image which are provided as input to a fully connected layer that predicts similarity (the general framework used by [6, 12]). Notably, these methods both underperform PAN without using any supervision, which reflects observations in prior work [18, 27]. In contrast, PAN-supervised outperforms all other methods, including for fashion compatibility where we report a staggering 6-17% boost over the attribute baselines on the resampled test set.

## 5. Conclusion

We presented PAN, a method of incorporating additional attribute annotations in image datasets to learn a better similarity predictor. We saw that PAN's method of decomposing similarity prediction into multiple conditions is general, functions with a range of different image encoders and is flexible in using attribute annotation, possibly sparse, when available. PAN outperformed state of the art on two diverse tasks—by 4-9% on fashion item compatibility prediction on Polyvore Outfits and 5% on few shot classification on CUB and over 1% Recall@1 on In-Shop Clothing Retrieval—contrary to prior approaches of using attribute supervision, which were unable to outperform methods that automatically learned concepts in different similarity conditions.

| | | Original | | Resampled | |
|---|---|---|---|---|---|
| | **Method** | **FITB** | **AUC** | **FITB** | **AUC** |
| **(a)** | TAN [28] | 57.6 | 0.88 | 38.1 | 0.66 |
| | SCE-Net [27] | 61.6 | 0.91 | 43.4 | 0.68 |
| | CSA-Net [13] | 63.7 | 0.91 | – | – |
| | CGAE [5] | 74.1 | **0.99** | 60.8 | 0.67 |
| **(b)** | X + Attr. Multitask-GE | 73.8 | **0.99** | 57.6 | 0.65 |
| | Attr. Similarity-GE | 69.5 | 0.98 | 52.9 | 0.65 |
| | PAN-Unsupervised-GE | 78.4 | **0.99** | 64.1 | 0.70 |
| | PAN-Supervised-GE | **82.3** | **0.99** | **69.7** | **0.71** |

Table 1: Comparison of PAN on fashion compatibility on Polyvore Outfits to (a) results reported in prior work or reproduced with the author's code and (b) other PAN and attribute supervision approaches.

| | **Method** | **Accuracy** |
|---|---|---|
| **(a)** | Baseline++ [3] | 83.58 |
| | ProtoNet [24] | 87.42 |
| | TriNet [4] | 84.10 |
| | TEAM [21] | 87.17 |
| | CGAE [5] | 88.00 $\pm$ 1.13 |
| **(b)** | X + Attr. Multitask-GE | 89.29 $\pm$ 0.57 |
| | Attr. Similarity | 92.21 $\pm$ 0.21 |
| | PAN-Unsupervised | 92.60 $\pm$ 0.10 |
| | PAN-Supervised | **92.77** $\pm$ 0.30 |

Table 2: Comparison of PAN on 5-way 5-shot classification on CUB-200-2011 to (a) results reported in prior work or reproduced with the author's code and (b) other PAN and attribute supervision approaches. Intervals provided are 95% confidence intervals over 3 different runs with different random model initializations

| | **Method** | **Recall@1** |
|---|---|---|
| **(a)** | MS [33] | 89.7 |
| | NormSoftMax[38] | 89.4 |
| | HORDE [10] | 90.4 |
| | Cont. w/M [34] | 91.3 |
| | ProxyNCA++[35] | 90.9 |
| **(b)** | ProxyNCA++ & Attr. Multitask | 90.8 |
| | ProxyNCA++ & Attr. Similarity | 86.4 |
| | ProxyNCA++ & PAN-Unsupervised | 91.4 |
| | ProxyNCA++ & PAN-Supervised | **92.1** |

Table 3: Comparison of PAN on In-Shop Clothing Retrieval to (a) results reported in prior work and (b) other PAN and attribute supervision approaches.

## Acknowledgements

This work was funded in part by DARPA

## References


[1] Rich Caruana. Multitask learning: A knowledge-based source of inductive bias. In *Proceedings of the Tenth International Conference on International Conference on Machine Learning*, 1993. 4

[2] Francisco M Castro, Manuel J Marín-Jiménez, Nicolás Guil, Cordelia Schmid, and Karteek Alahari. End-to-end incremental learning. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 233–248, 2018. 1

[3] Wei-Yu Chen, Yen-Cheng Liu, Zsolt Kira, Yu-Chiang Frank Wang, and Jia-Bin Huang. A closer look at few-shot classification. *arXiv preprint arXiv:1904.04232*, 2019. 3, 4

[4] Zitian Chen, Yanwei Fu, Yinda Zhang, Yu-Gang Jiang, Xiangyang Xue, and Leonid Sigal. Multi-level semantic feature augmentation for one-shot learning. *IEEE Transactions on Image Processing*, pages 1–1, 2019. 4

[5] Guillem Cucurull, Perouz Taslakian, and David Vazquez. Context-aware visual compatibility prediction. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12617–12626, 2019. 1, 3, 4

[6] Ali Farhadi, Ian Endres, Derek Hoiem, and David Forsyth. Describing objects by their attributes. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1778–1785. IEEE, 2009. 4

[7] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep image retrieval: Learning global representations for image search. In *European conference on computer vision*, pages 241–257. Springer, 2016. 1

[8] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 3

[9] Ruining He, Charles Packer, and Julian McAuley. Learning compatibility across categories for heterogeneous item recommendation. In *International Conference on Data Mining (ICDM)*, 2016. 1

[10] Pierre Jacob, David Picard, Aymeric Histace, and Edouard Klein. Metric learning with horde: High-order regularizer for deep embeddings. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 6539–6548, 2019. 4

[11] Gregory Koch, Richard Zemel, and Ruslan Salakhutdinov. Siamese neural networks for one-shot image recognition. In *ICML deep learning workshop*, volume 2, 2015. 1

[12] Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. Concept bottleneck models. In *International Conference on Machine Learning*, pages 5338–5348. PMLR, 2020. 1, 2, 4

[13] Yen-Liang Lin, Son Tran, and Larry S. Davis. Fashion outfit complementary item retrieval. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1, 2, 4

[14] Weiyang Liu, Yandong Wen, Zhiding Yu, Ming Li, Bhiksha Raj, and Le Song. Sphereface: Deep hypersphere embedding for face recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 212–220, 2017. 1

[15] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 2, 3

[16] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3456–3465, 2017. 1

[17] Bryan A. Plummer, M. Hadi Kiapour, Shuai Zheng, and Robinson Piramuthu. Give me a hint! Navigating Image Databases using Human-in-the-loop Feedback. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, 2019. 1

[18] Bryan A Plummer, Paige Kordas, M Hadi Kiapour, Shuai Zheng, Robinson Piramuthu, and Svetlana Lazebnik. Conditional image-text embedding networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 249–264, 2018. 1, 4

[19] Bryan A. Plummer, Arun Mallya, Christopher M. Cervantes, Julia Hockenmaier, and Svetlana Lazebnik. Phrase localization and visual relationship detection with comprehensive image-language cues. In *The IEEE International Conference on Computer Vision (ICCV)*, 2017. 1, 2

[20] Bryan A. Plummer, Mariya I. Vasileva, Vitali Petsiuk, Kate Saenko, and David Forsyth. Why do these match? Explaining the behavior of image similarity models. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 3

[21] Limeng Qiao, Yemin Shi, Jia Li, Yaowei Wang, Tiejun Huang, and Yonghong Tian. Transductive episodic-wise adaptive metric for few-shot learning. In *The IEEE International Conference on Computer Vision (ICCV)*, October 2019. 4

[22] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 1

[23] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 815–823, 2015. 1

[24] Jake Snell, Kevin Swersky, and Richard Zemel. Prototypical networks for few-shot learning. In *Advances in Neural Information Processing Systems*, pages 4077–4087, 2017. 1, 4

[25] Stefan Stojanov, Samarth Mishra, Ngoc Anh Thai, Nikhil Dhanda, Ahmad Humayun, Chen Yu, Linda B Smith, and James M Rehg. Incremental object learning from contiguous views. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8777–8786, 2019. 1

[26] Flood Sung, Yongxin Yang, Li Zhang, Tao Xiang, Philip HS Torr, and Timothy M Hospedales. Learning to compare: Relation network for few-shot learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1199–1208, 2018. 1

[27] Reuben Tan, Mariya I Vasileva, Kate Saenko, and Bryan A Plummer. Learning similarity conditions without explicit

supervision. In *The IEEE International Conference on Computer Vision (ICCV)*, pages 10373–10382, 2019. 1, 3, 4

[28] Mariya I Vasileva, Bryan A Plummer, Krishna Dusad, Shreya Rajpal, Ranjitha Kumar, and David Forsyth. Learning type-aware embeddings for fashion compatibility. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 390–405, 2018. 1, 2, 3, 4

[29] Andreas Veit, Serge Belongie, and Theofanis Karaletsos. Conditional similarity networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 830–838, 2017. 1, 2

[30] Andreas Veit, Balazs Kovacs, Sean Bell, Julian McAuley, Kavita Bala, and Serge Belongie. Learning visual clothing style with heterogeneous dyadic co-occurrences. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 4642–4650, 2015. 1

[31] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in neural information processing systems*, pages 3630–3638, 2016. 1

[32] 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. 2, 3

[33] Xun Wang, Xintong Han, Weilin Huang, Dengke Dong, and Matthew R Scott. Multi-similarity loss with general pair weighting for deep metric learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5022–5030, 2019. 4

[34] Xun Wang, Haozhi Zhang, Weilin Huang, and Matthew R Scott. Cross-batch memory for embedding learning. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 4

[35] Eu Wern Teh, Terrance DeVries, and Graham W Taylor. Proxynca++: Revisiting and revitalizing proxy neighborhood component analysis. *arXiv*, pages arXiv–2004, 2020. 3, 4

[36] Fan Yang, Ryota Hinami, Yusuke Matsui, Steven Ly, and Shin'ichi Satoh. Efficient image retrieval via decoupling diffusion into online and offline processing. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9087–9094, 2019. 1

[37] Xun Yang, Xiangnan He, Xiang Wang, Yunshan Ma, Fuli Feng, Meng Wang, and Tat-Seng Chua. Interpretable fashion matching with rich attributes. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 775–784, 2019. 1, 2

[38] Andrew Zhai and Hao-Yu Wu. Classification is a strong baseline for deep metric learning. *arXiv preprint arXiv:1811.12649*, 2018. 4

[39] Zhongping Zhang, Tianlang Chen, Zheng Zhou, Jiaxin Li, and Jiebo Luo. How to become instagram famous: Post popularity prediction with dual-attention. In *2018 IEEE International Conference on Big Data (Big Data)*, pages 2383–2392. IEEE, 2018. 1