Fashion Retrieval via Graph Reasoning Networks on a Similarity Pyramid

Zhanghui Kuang[1]*, Yiming Gao[12]*, Guanbin Li[2], Ping Luo[3], Yimin Chen[1], Liang Lin[2], Wayne Zhang[1]†
[1]SenseTime Research [2]Sun Yat-sen University [3]The University of Hong Kong
{kuangzhanghui, chenyimin, wayne.zhang}@sensetime.com gaoym9@mail2.sysu.edu.cn
liguanbin@mail.sysu.edu.cn pluo@cs.hku.hk linliang@ieee.org

Abstract

Matching clothing images from customers and online shopping stores has rich applications in E-commerce. Existing algorithms encoded an image as a global feature vector and performed retrieval with the global representation. However, discriminative local information on clothes are submerged in this global representation, resulting in sub-optimal performance. To address this issue, we propose a novel Graph Reasoning Network (GRNet) on a Similarity Pyramid, which learns similarities between a query and a gallery cloth by using both global and local representations in multiple scales. The similarity pyramid is represented by a Graph of similarity, where nodes represent similarities between clothing components at different scales, and the final matching score is obtained by message passing along edges. In GRNet, graph reasoning is solved by training a graph convolutional network, enabling to align salient clothing components to improve clothing retrieval. To facilitate future researches, we introduce a new benchmark FindFashion, containing rich annotations of bounding boxes, views, occlusions, and cropping. Extensive experiments show that GRNet obtains new state-of-the-art results on two challenging benchmarks, e.g. *pushing the top-1, top-20, and top-50 accuracies on DeepFashion to 26%, 64%, and 75%* (i.e. *4%, 10%, and 10% absolute improvements), outperforming competitors with large margins. On FindFashion, GRNet achieves considerable improvements on all empirical settings.*

1. Introduction

Fashion image retrieval between customers and online shopping stores has various applications for E-commerce. Given a street-snapshot of clothing image, this task is to search the same garment item in the online store. It is a key step for future applications such as generating descriptions of clothes, brands, materials, and styles. While matching clothes across modalities appears effortless for human vision, it is extremely challenging for machine vision. The same cloth may exhibit large variations due to occlusions, cropping, and viewpoints. More importantly, garments may differ in small local regions such as logos only.

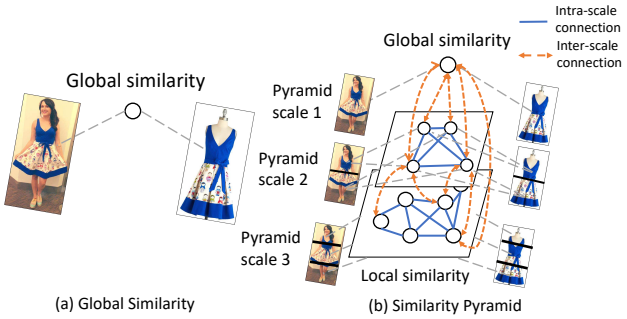

Figure 1: Comparison between global similarity and similarity pyramid with graph reasoning. The left illustrates the global similarity. The right shows the similarity pyramid with graph reasoning, where scale 1 computes the global similarity while scale 2 and 3 compute local similarities between all possible combinations of local patches from one image pair. The dash gray lines indicate one similarity is related to two patches. Pyramid similarities (including the global and the local) are reasoned mutually. The blue lines indicate interactions between similarities at one scale while the red dash lines indicate those across scales (best viewed in color).

The task of customer-to-shop clothes retrieval has great progresses [22, 25, 34, 24, 44, 9, 13, 8, 58] by using convolutional neural networks (CNNs) [28, 19, 14, 21, 39]. Existing methods often employed the global similarity pipeline. For example, they first aggregate local features into compact global features, and then compute global similarities between query and gallery images by using cosine or Euclidean distance (see Figure 1 (a)). In the procedure of global feature aggregation, the discriminative local regions

*They contributed equally to this work
†Wayne Zhang is the corresponding author

of clothes would be submerged in this global representation. In contrast, human vision verifies whether two clothes are the same by simultaneously comparing the query and the gallery in terms of both global features such as fabric, colors, textures and categories (*e.g*. "dress" or "t-shirt"), as well as local features such as sleeve, collar, and logos. Moreover, human vision only focuses on common parts between the query and the gallery, while ignores those regions only exist in the query (or the gallery) due to occlusions, cropping or viewpoints. We conjecture that for clothing retrieval and verification, comparing clothes in both global and local ways is complementary.

Inspired by the procedure above, we design a novel *Graph Reasoning Network (GRNet)* on a *Similarity Pyramid* to compare a query and a gallery image both globally and locally at different similarity scales. As illustrated in Figure 1 (b), we extract CNN features for all spatial regions at each pyramid scale. An important problem for matching clothes is that the local clothing regions are often mismatched. In order to solve misalignment between the query and the gallery, we have to enumerate all the region pairs in the same scale to calculate their similarities. However, as the local regions are not equally important, similarities between aligned regions should be dominated, while those between misaligned pairs should be ignored.

To this end, we construct a pyramid defined by similarities between clothing regions. This *similarity pyramid* can be formulated as a *graph*, where each node of the graph is the similarity between two corresponding clothing regions in the same scale, while each edge connected two nodes is the normalized similarity of them. The final similarity (matching score) between a query and a gallery image can be achieved by reasoning on this graph. GRNet contains a key component of a graph convolutional network (GCN), which performs graph reasoning by propagating messages between nodes.

The proposed GRNet greatly suppresses the performance degradation caused by occlusions, cropping, viewpoints and small logos, outperforming existing methods with large margins as shown in Figure 2. Specifically, on the DeepFashion [34] benchmark, GRNet *absolutely* improves the top-1, top-20, and top-50 accuracies of the best ever reported results by 12%, 21% and 18%, and the best results of two state-of-the-art deep matching methods [43, 54][1] by 4%, 10%, and 10% respectively On Street2Shop [25] benchmark, GRNet achieves new state-of-the-art results on all five categories *i.e*. "tops", "dresses", "skirts", "pants" and "outerwear".

Furthermore, existing benchmarks such as Street2Shop [25], DARN [22], and DeepFashion [34] have progressed the researches of customer-to-shop clothes retrieval. However, the detailed annotations of occlusions, cropping and views are limited, impeding ablation studies of this task. And they are not suitable to analyze which and how variations affect the retrieval performance.

[1]We used the codes released by authors and retrained the models on DeepFashion.

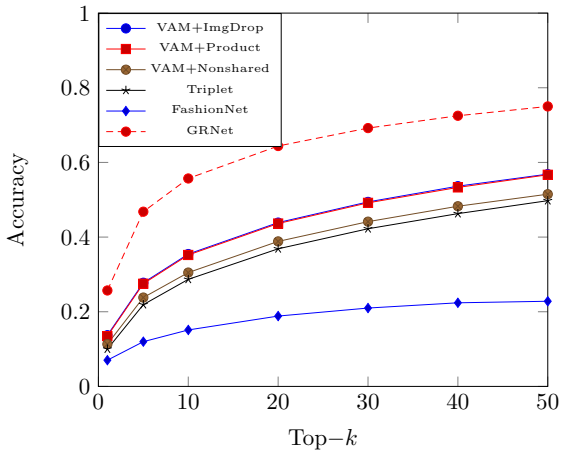

Figure 2: Comparison with state-of-the-art methods on DeepFashion consumer-to-shop dataset [34]. ImgDrop+GoogleNet and Product+GoogleNet are the best two results ever reported [53].

To this end, we build a new customer-to-shop clothing retrieval benchmark, named *FindFashion*, by revisiting existing datasets, and annotating attributes in terms of occlusions, cropping, and views. FindFashion allows in depth analysis of the impacts of each variation on clothes retrieval. We further introduce four new evaluation protocols of varying difficulties, including *Easy*, *Hard-View*, *Hard-Occlusion*, and *Hard-Cropping*. The splits of training, validation, and test set on FindFashion will be released for fair comparisons.

Our main **contributions** are summarized in three aspects. (1) We propose an effective approach for clothing retrieval, *Graph Reasoning Network (GRNet)* on a *Similarity Pyramid*. GRNet computes similarities between a query and a gallery image at different local clothing regions and scales. GRN has an important component of graph convolutional neural network to propagate similarities on the pyramid, performing graph reasoning and producing state-of-the-art performance. (2) We validate the effectiveness of GRNet on two popular datasets, DeepFashion and Street2Shop. GRNet outperforms state-of-the-art methods with significantly large margins. (3) We annotate different variations and build a new customer-to-shop retrieval benchmark named FindFashion, which allows the in-depth analysis of the effect of variations for clothing retrieval. Extensive experiments demonstrate that GRNet is more robust against occlusions, cropping, or non-front views than previous methods.

2. Related Work

Datasets	Street2Shop [25]	DARN [22]	DeepFashion [34]	**Our**
#images	416,840	182,780	239,557	565,041
#pairs	39,479	91,390	195,540	382,230
Public split	✓	✗	✓	✓
Bbox	✓	✗	✓	✓
View	✗	✗	✗	✓
Occlusion	✗	✗	✗	✓
Cropping	✗	✗	✗	✓

Table 1: Comparison of customer-to-shop clothes retrieval datasets.

Clothes retrieval. Pioneer work [50, 10, 12, 13] on clothing retrieval utilized conventional features such as SIFT and semantic preserving visual phrases. Recently, deep neural networks have been widely applied in clothing retrieval and have pushed the research into a new phase [22, 25, 34, 24, 44, 9, 8, 58]. These methods usually follow a global similarity computation and matching pipeline, *i.e*. aggregating local features into a single global representation and then performing similarity computation. [22, 34] explored attributes via multi-task learning to learn representations which are related to specific tags such as "Crew neck", "Short sleeves" and "Rectangle-shaped"; [25, 29] investigated different network architectures which are adept at extracting global features for customer-to-shop clothes retrieval. Instead, [58, 9] attempted to train models with weakly or noisy supervised signals to reduce the dependency of data annotation and increase the global feature learning efficiency. Recently, [24] utilized attribute labels to pay more attention to local discriminative regions. Similarly, [53] focused on clothes regions and ignored cutter background via a cloth parsing subnetwork. Both the two work employed attention mechanisms in the global feature aggregation to suppress local distractive regions and up-weight the discriminative ones to some extent. However, they were highly dependent on explicit knowledge such as label and cloth parsing category definition which might be unavailable in real application scenarios. On the contrast, we conduct clothes matching computation via pyramid similarity (including both global and local ones) learning on a relation graph, which can obtain salient component alignment through similarity propagation, and thus achieve more accurate matching. Notably, the proposed approach achieves similarities weighting by end-to-end classification training without any explicit supervised signals. Therefore, it is very practical.

There also exist some variants, such as dialog based clothes search [17] , video based clothes retrieval [8], and attribute feedback based clothes retrieval [18, 59]. Their application scenarios and settings are different from ours.

Customer-to-shop clothes retrieval datasets. There exist some customer-to-shop clothes retrieval datasets as listed in Table 1. Kiapour *et al*. [25], collected Street2Shop dataset from a large online retail store. It consists of 78,958 images, 39,479 customer-to-shop pairs, and 396,483 gallery images. Huang *et al*. [22] collected DARN dataset which is composed of upper-clothing images. It has 182,780 images, 91,390 pairs, and 91,390 gallery images, in which only query images are of bounding boxes. However, the training/testing split is not available and thus prevent other research from making a fair comparison. Liu *et al*. [34] released DeepFashion dataset. It has 239,557 images, 195,540 customer-to-shop pairs, and 45,392 gallery images. It is later revisited for fine grained attribution recognition [57]. All the above datasets are lack of detailed attributes which are most related to clothes retrieval performance. Our benchmark FindFashion contains detailed attribute annotations (*e.g*. views, occlusions and cropping), so that the impacts of attributes on the retrieval performance can be analyzed in detail. We have also noticed that there exist other clothes datasets such as [3], [5], [60], [23], [33] and [1]. These datasets mainly target at clothes segmentation, attribution prediction and fashion comments, but not customer-to-shop clothes retrieval, and are lack of clothes pairs for evaluation. [18] released Fashion 200k which aims at attribution discovery and clothes retrieval with attribute manipulation, and is very different from our task.

Graph reasoning. Graph naturally models the dependencies between concepts, which facilitate the research on graph reasoning such as Graph CNN [11, 27, 40], and Gated Graph Neural Network (GGNN) [30]. These graph neural networks have been widely employed in various tasks of computer vision and have made very promising progress, *e.g*. object parsing [31, 32], multi-label image recognition [52], visual question answer [46], social relationship understanding [51], person re-identification [42] and action recognition [49]. These work create knowledge graph based on the relationship of different entities, *e.g*. images, objects, proposals, and semantic categories. Instead, we are the first to explore the use of knowledge graph to express the similarity between different pairs of local regions, and apply it to a new field of customer-to-shop clothes retrieval. It can realize the weighting of local region pairs and the enhancement of global matching through the iteration of propagation between pyramid similarities relations, and thus obtain more accurate matching computation.

Image retrieval. Our work is related to image retrieval approaches [35, 16, 2, 38, 15, 47, 55, 6, 7]. They target at retrieving rigid objects such as buildings, or scenes, and often aggregate regional features into compact representations to compute global similarities. Different from them, our GRNet aims at retrieving more challenging non-rigid clothes. Moreover, our GRNet captures both local and global similarities, and conducts graph reasoning on a similarity

pyramid.

Metric learning. Our work is also related to general deep metric learning [36, 56, 26, 37, 54]. However, they only conducted experiments on InShop clothes retrieval dataset while our work focuses on customer-to-shop clothes retrieval which is much more challenging as analyzed in [34]. We have also compared the proposed GRNet with the state-of-the-art method [54] in our experiments.

3. Methods

3.1. Motivation

The setup of the customer-to-shop clothes retrieval is the following. Given one customer clothes image query $\mathbf{x}$ and one shop clothes gallery set $\mathbb{G} = \{\mathbf{y}\}$, it computes the similarities s between $\mathbf{x}$ and $\mathbf{y}$, and ranks them. $\mathbf{x} = \{\mathbf{x}^i\}$ and $\mathbf{y} = \{\mathbf{y}^i\}$, where $\mathbf{x}^i \in \mathbb{R}^{C\times 1}$ and $\mathbf{y}^i \in \mathbb{R}^{C\times 1}$ are local features of the customer clothes image and the shop one respectively. Previous customer-to-shop clothes retrieval approaches [22, 25, 34, 24, 44, 9, 13, 8, 58] adopt the following global similarity as:

$$s_g = S_g(A(\mathbf{x}), A(\mathbf{y})), \tag{1}$$

where $A(\cdot)$ is the aggregation function and $S_g(\cdot,\cdot)$ is the scalar global similarity function. The aggregation function is usually the average pooling or max-pooling operator. The similarity function often adopts the cosine similarity or Euclidean distance. Ordinarily, the global similarity can reliably estimate the similarity between the query and the gallery. However, the aggregation function might aggregate noisy features such as clutter background, other objects, or unique regions which can only be observed in the query or the gallery when existing occlusions, cropping or different views. This undoubtedly greatly degrades the clothes retrieval performance.

To suppress the above issues, [48, 4] computed the similarity between the query and the gallery by summing up local similarities between local feature pairs with a greedy strategy as follows:

$$s_l = \sum_{i,j} w_l^{ij} S_l(\mathbf{x}^i, \mathbf{y}^j), \tag{2}$$

where $S_l(\cdot,\cdot)$ is the scalar local similarity function, and w_l^{ij} is the scalar weight of local similarities $S_l(\mathbf{x}^i, \mathbf{y}^j)$, which is given by

$$w_l^{ij} = \begin{cases} 1, & \text{if } j = \text{argmax}_k(S_l(\mathbf{x}^i, \mathbf{y}^k)). \\ 0, & \text{otherwise.} \end{cases} \tag{3}$$

However, greedily finding local feature pairs easily leads to misalignment, which accumulates errors in the final estimated similarity s_l.

We attempt to make full use of both the global and local similarities, and learn the importance of different similarities (*i.e.* w_l^{ij}) automatically to mitigate the above issues.

3.2. Graph Reasoning Network

For each query (or gallery) image, instead of extracting local features $\mathbf{x}^i$ (or $\mathbf{y}^i$) and global features $A(\mathbf{x})$ (or $A(\mathbf{y})$), we extract multi-scale features at pyramid spatial windows, and obtains $\{\mathbf{x}_l^i \in \mathbb{R}^{C\times 1}\}$ (or $\{\mathbf{y}_l^i \in \mathbb{R}^{C\times 1}\}$) with $\mathbf{x}_l^i$ (or $\mathbf{y}_l^i$) being the i^{th} local feature for pyramid scale l, where $l \in \{1, \cdots, L\}$ indicates scale index from top to down. Therefore, $\mathbf{x}_1^1$ and $\mathbf{y}_1^1$ refer to the global feature vector of the query and that of the gallery (*i.e.*, $A(\mathbf{x})$ and $A(\mathbf{y})$) respectively. For each scale l, assuming there exist $R_l \times C_l$ local spatial windows for each image, we totally have $\sum_l R_l C_l$ pyramid features.

Similarity pyramid graph. We build a similarity pyramid graph with all region pair similarities being the graph nodes, and the relations between two similarities being the edges. Formally, given a pair of local feature $\mathbf{x}_l^i$ and $\mathbf{y}_l^j$ from the same pyramid scale l, we compute their similarity vector $\mathbf{s}_l^{ij} \in \mathbb{R}^{D\times 1}$ instead of a similarity scalar in Equation 1 and 2, by a vector similarity function given by

$$S_p(\mathbf{x}_l^i, \mathbf{y}_l^j) = \frac{\mathbf{P}\left|\mathbf{x}_l^i - \mathbf{y}_l^j\right|^2}{\left\|\mathbf{P}\left|\mathbf{x}_l^i - \mathbf{y}_l^j\right|^2\right\|_2}, \tag{4}$$

where $|\cdot|^2$ and $\|\cdot\|_2$ indicate element-wise square and l_2-norm respectively. $\mathbf{P} \in \mathbb{R}^{D\times C}$ is a projection matrix which projects pyramid feature difference vectors from C dimension to a lower D dimension. Similarity vectors are guaranteed to have the same magnitude by performing l_2-normalization. For any node pair in the graph $\mathbf{s}_{l_1}^{ij}$ and $\mathbf{s}_{l_2}^{mn}$, we define a scalar edge weight $w_p^{l_1 ij, l_2 mn}$, which is given by

$$w_p^{l_1 ij, l_2 mn} = \frac{\exp((\mathbf{T}_{out}\mathbf{s}_{l_1}^{ij})^\intercal(\mathbf{T}_{in}\mathbf{s}_{l_2}^{mn}))}{\sum_{l,p,q}\exp((\mathbf{T}_{out}\mathbf{s}_{l_1}^{ij})^\intercal(\mathbf{T}_{in}\mathbf{s}_{l}^{pq}))}, \tag{5}$$

where $\mathbf{s}^\intercal$ indicates the transpose of the vector $\mathbf{s}$. $\mathbf{T}_{in} \in \mathbb{R}^{D\times D}$ and $\mathbf{T}_{out} \in \mathbb{R}^{D\times D}$ are the linear transformations of incoming edges and outgoing edges for each graph node respectively. When $l_1 = l_2$, $w_p^{l_1 ij, l_2 mn}$ are intra-scale edges. *i.e.*, their two connected similarity nodes come from the same scale. When $l_1 \neq l_2$, $w_p^{l_1 ij, l_2 mn}$ are inter-scale edges. *i.e.*, their two nodes come from different scales. Inter-scale edges enable similarities with different scales to propagate messages from each other. In this way, the similarity pyramid graph is defined as $G = (\mathbb{N}, \mathbb{E})$, where $\mathbb{N} = \{\mathbf{s}_l^{ij}\}$ and $\mathbb{E} = \{w_p^{l_1 ij, l_2 mn}\}$.

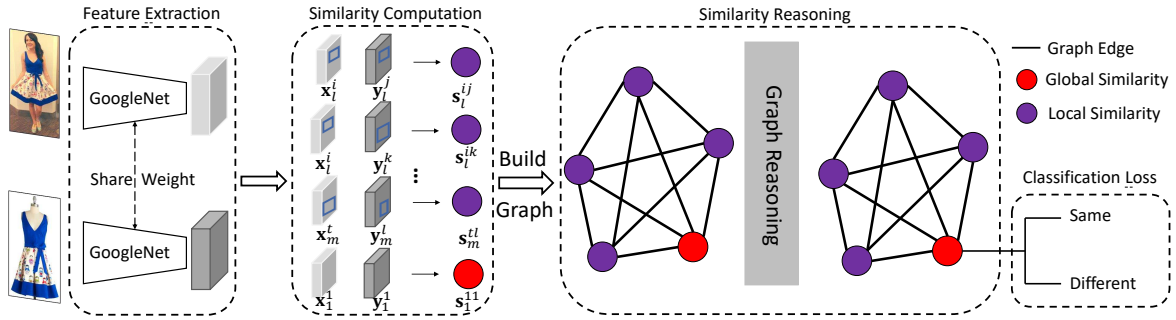

Figure 3: The overall framework of the proposed GRNet. Given one query and gallery pair, their features extracted by deep convolutional networks are fed into Similarity Computation to build a similarity pyramid graph with all region pair similarities being the graph nodes. In the Similarity Computation, $\mathbf{x}_l^i$ is the i^{th} local feature of the query at scale l while $\mathbf{y}_l^j$ is the j^{th} one of the gallery, and $\mathbf{s}_l^{ij}$ is their similarity vector. Further, the global and local similarities are propagated and updated via Similarity Reasoning. It finally outputs whether the input image pair belong to the same cloth or not.

Similarity reasoning. We reason the similarity $\mathbf{s}_l^{ij}$ by conducting a sequence of similarity propagation, linear transformation, and non-linear activation operator. Concretely, similarity is first propagated as

$$\hat{\mathbf{s}}_{l_1}^{ij} = \sum_{l_2,m,n} w_p^{l_1 ij, l_2 mn} \mathbf{s}_{l_2}^{mn} \tag{6}$$

$$= \sum_{l_2,m,n} w_p^{l_1 ij, l_2 mn} S_p(\mathbf{x}_{l_2}^m, \mathbf{y}_{l_2}^n). \tag{7}$$

Then, the linear transformation and the non-linear activation are conducted as

$$\mathbf{h}_{l_1}^{ij} = \text{ReLU}(\mathbf{W}\hat{\mathbf{s}}_{l_1}^{ij}), \tag{8}$$

where $\mathbf{W} \in \mathbb{R}^{C' \times D}$ is the learnable parameters. Equation 6 and 8 can be easily implemented by graph convolution [27], followed by the nonlinear ReLU. We iteratively reason the similarity pyramid T times by setting $\mathbf{s}_{l_2}^{mn}$ in the right hand side of Equation 6 at current step to $\mathbf{h}_{l_2}^{mn}$ from previous step.

End-to-end training. We use the cross entropy loss over the final reasoned global similarity vector (*i.e.*, $\mathbf{h}_1^{11}$) and the ground truth $\bar{s}$ corresponding to the query and the gallery $(\mathbf{x}, \mathbf{y})$ to train the whole network end-to-end. In this way, similarities, and their importance are jointly learned.

Network architecture. Figure 3 illustrates the overall framework of the proposed graph reasoning network. It consists of four modules including feature extraction, similarity computation, similarity reasoning and classification loss. In the feature extraction module, we employ GoogleNet [45] as the backbone, and extract pyramid features by performing max-pooling on its last convolution activation over spatial windows with different pyramid sizes. Both the query and gallery share the same feature extractor. In the similarity computation module, we compute the similarity between all possible local feature combinations between the query and the gallery at the same pyramid scale. In the similarity reasoning module, we employ a stack of graph convolution and ReLU operators.

Setups	*E*	*HO*	*HC*	*HV*
#Validation	125863	4920	15883	47164
#Test	30746	1250	3883	11383

Table 2: Statistics of four evaluation setups on FindFashion.

4. FindFashion

We build a new benchmark named FindFashion by revisiting the publicly available datasets. *i.e.*, Street2Shop [25], and DeepFashion [34]. We labeled 3 attributes (*i.e.*, occlusions, views, and cropping) which mostly affect clothes retrieval performance. According to the attributes of query, we divided the benchmark into 4 subsets with different difficulty levels. *i.e.*, *Easy*, *Hard-Cropping*, *Hard-Occlusion*, and *Hard-View*.

We adopt the same evaluation measure, *i.e.*, top-k accuracy, to evaluate the performance as in [25, 34].

Data Collection and cleaning. We first merged the two existing datasets (*i.e.*, Street2Shop [25], DeepFashion [34]), and formed a large dataset containing 382,230 image pairs and 565,041 images, and then we asked the annotators to screen out the image pairs that are clearly not of the same clothes.

Annotations. Gallery images from Street2Shop have no clothes bounding boxes, we first train a Faster RCNN [39] detector over DeepFashion to detect their bounding boxes, and then manually correct them. We annotate three attributes (*i.e.*, views, occlusions and cropping) for all im-

Methods	Top-1	Top-20	Top-50
FashionNet [34]	7.0	18.8	22.8
Triplet [53]	10.0	37.0	49.9
VAM+Nonshared [53]	11.3	38.8	51.5
VAM+Product [53]	13.4	43.6	56.7
VAM+ImgDrop [53]	13.7	43.9	56.9
DREML(192,48) [54]	18.6	51.0	59.1
KPM [43]	21.3	54.1	65.2
GRNet	**25.7**	**64.4**	**75.0**

Table 3: Comparison with state-of-the-art methods on DeepFashion consumer-to-shop benchmark [34].

Method	Tops	Dresses	Skirts	Pants	Outerwear
Kiapour *et al.* [25]	38.1	37.1	54.6	29.2	21.0
VAM+ImgDrop [53]	52.3	62.1	70.9	–	–
Trip. [53]	44.9	56.0	69.0	–	–
Trip.+Partial [53]	47.0	58.3	72.3	–	–
GRNet	**58.3**	**64.2**	**72.5**	**48.5**	**38.6**

Table 4: Comparison with state-of-the-art methods on Street2Shop [25] in terms of top-20 accuracy.

ages. For views, we labeled each clothes images as front, side, or back. Clothes with the yaw angle in $[-45°, 45°]$ are labelled as front, those with yaw angle in $(45°, 135°)$ or $(-135°, -45°)$ are labelled as side while $[135°, 225°]$ as back. For occlusions, clothes with more than 30% occluded by other things such as other clothes, mobile phone or belt are labelled as occluded otherwise as un-occluded. For cropping, clothes with more than 30% cropped are labelled as cropped otherwise as un-cropped.

Images in FindFahsion are of large variance in terms of views, cropping, and occlusions. 8% of images are cropped. 3% of them are occluded. Front view, side view, and back view account for 74%, 20%, and 6% respectively.

Evaluation protocol. As done in [34], we report top-k accuracy to evaluate the retrieval performance. It reflects the quality of the results of a search engine as they would be visually inspected by a user. Four evaluation setups of different difficulty levels are defined according to the query attribute while keeping the gallery unchanged in the test set:

(1) *Easy (E)*, queries are captured from the front view without cropping or occlusion.

(2) *Hard-Cropping (HC)*, queries are with cropping.

(3) *Hard-Occlusion (HO)*, queries are occluded.

(4) *Hard-View (HV)*, queries are of non-frontal view. Namely, side or back view.

We do not split training dataset according to the above four evaluation setups as we found using maximum training data can achieve better results in all the setups. The detailed statistics of our evaluation protocols are listed in Table 2.

Projection dim. D	Channel num. C'	Accuracy		
		Top-1	Top-20	Top-50
512	128	25.73	64.38	75.00
512	256	25.52	64.50	74.43
512	512	25.92	64.75	75.54
256	128	24.06	63.02	73.33
256	256	25.10	64.48	74.17
128	128	24.69	63.64	74.38

Table 6: Impacts of Dimensions.

5. Experiments

5.1. Implementation Details

Our implementation on customer-to-shop clothes retrieval follows the practice in [34]. We train our models with PyTorch. We perform standard data augmentation with random horizontal flipping. All cropped images are resized to 224×224 before being fed into the networks. Optimization is performed using synchronous SGD with momentum 0.9, and weight decay 0.0005 on servers with 8 GPUs. The initial learning rate is set to 0.01 and decreased by a factor of 10 every 20 epochs. All compared models including ours are trained using the same training set for 60 epochs. The feature extractor is initialized with its pre-trained model on ImageNet while the similarity computation module and the similarity reasoning module are randomly initialized as with [20].

In the feature extraction module, we have totally 7 scales including the global one (*i.e.*, $L = 7$). The whole spatial window of images is divided into 1×1, 1×2, 2×1, 2×2, 1×3, 3×1 and 3×3 from scale 1 to 7 respectively. In the similarity reasoning module, we use three (*i.e.*, $T = 3$) graph convolution layers with channel number C' set to 128. The projection dimension (*i.e.*, D) is set to 512.

We set the batch size to 64 during training. Each batch consists of 32 clothes with 2 images per clothes. The query and gallery pairs of the same clothes construct positive training samples while other combinations negative ones.

5.2. Results on DeepFashion [34]

Table 3 compares the proposed GRNet with state-of-the-art methods, including FashionNet [34], triplet-based metric learning approach, and Visual Attention Model (VAM) and its variants (VAM+ImgDrop, VAM+Product, and VAM+Nonshared) [53], on DeepFashion [34]. Except FashionNet, all counterparts use the same backbone GoogleNet [45]. The proposed GRNet outperforms existing methods with an impressive margin. Specifically, it obtains an accuracy of 25.7, 64.4 and 75.0, and absolutely improves the best ever reported results (VAM+Product) by 12%, 21% and 18% respectively. Notably, VAM uses an attention subnetwork which needs clothes segmentation dataset for training. The GRNet is trained with only query-gallery image

#	Local similarity						Intra-scale connection	Inter-scale connection	Accuracy		
	1×2	2×1	2×2	3×1	1×3	3×3			top−1	top−20	top−50
1	-	-	-	-	-	-	-	-	14.06	47.60	60.62
2	✓	✓	✓	-	-	-	✓	✓	22.60	62.71	73.25
3	-	-	-	✓	✓	✓	✓	✓	23.96	64.48	74.32
4	✓	✓	✓	✓	✓	✓	-	-	24.48	63.85	74.17
5	✓	✓	✓	✓	✓	✓	-	✓	24.79	64.17	74.27
6	✓	✓	✓	✓	✓	✓	✓	-	24.58	63.85	73.44
7	✓	✓	✓	✓	✓	✓	✓	✓	**25.73**	**64.38**	**75.00**

Table 5: Ablation experiments on DeepFashion [34].

Methods	Easy			Hard-View			Hard-Occlusion			Hard-Croping		
	Top-1	Top-20	Top-50	Top-1	Top-20	Top-50	Top-1	Top-20	Top-50	Top-1	Top-20	Top-50
Baseline	16.9	53.6	67.6	10.4	37.8	53.2	4.5	25.3	35.8	7.3	35.4	49.9
DREML(192,48) [54]	20.7	54.2	68.2	17.2	44.3	54.0	6.3	31.3	43.8	10.6	43.4	55.2
KPM [43]	22.9	56.2	69.2	18.3	45.8	55.8	5.8	25.5	35.4	9.7	34.8	46.7
GRNet	**27.1**	**65.1**	**75.2**	**23.3**	**57.9**	**69.6**	**7.8**	**35.0**	**45.0**	**14.9**	**48.4**	**61.1**

Table 7: Comparison with state-of-the-art methods on FindFashion.

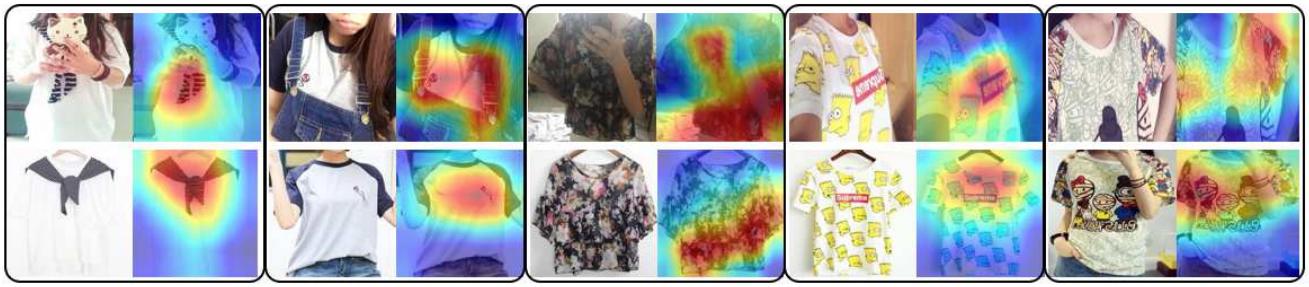

Figure 4: Visualization of important regions in the query and the gallery images. Each 2×2 images in one rectangle show one query-gallery image pair and their corresponding highlights, in which the top-left, the top-right, the bottom-left, and the bottom-right are the query, the query highlights, the gallery, and the gallery highlights respectively. Query 1 and 3 are occluded by hands; query 2 is occluded by trousers; query 4 is side view while its gallery front; query 5 is cropped.

pairs, thus it is more practical. We also compare GRNet with DREML [54], which achieves state-of-the-art performance on multiple general metric learning benchmarks including Inshop [34], recently. We train the DREML model on DeepFashion training set using its open source code with 192 recommended meta classes and 48 ensemble models as done in Table 2 of DREML [54]. Our GRNet is remarkably superior than DREML although DREML employs 48 models for ensemble. Moreover, we also compare GRNet with KPM [43], which achieves state-of-the-art performance on multiple person re-identification benchmarks and uses the same backbone as our GRNet. Again, our GRNet outperforms KPM remarkably.

5.3. Visualization

To investigate why GRNet works effectively, we employ Grad-CAM [41] to visualize the important regions in the query and the gallery images for predicting whether they belong to the same clothes or not in Figure 4. It has been shown that GRNet automatically focuses on local discriminative regions (*e.g.*, scarf, and logo) and shared regions which can be observed in both the query and the gallery while ignores non-discriminative regions (*e.g.*, non-texture regions), occlusions (*e.g.*, hand) or unique regions which can be observed only in one side due to different views or cropping. We visualize the similarity node which contributes most to the final classification by selecting the one whose edge outgoing to the global similarity node has the largest weight, in Figure 5. It has been shown that our GRNet can focus on aligned salient clothing components (*e.g.*, logo).

5.4. Results on Street2Shop

We compare the proposed GRNet with state-of-the-art customer-to-shop clothes retrieval methods on Street2Shop dataset [25] in Table 4. It has been shown that it achieves the best results on all five categories of Street2Shop. Particularly, it absolutely improves the best ever reported results by 11.3% and 5.9% for tops and dresses categories respectively.

5.5. Ablation Study

We investigate the effectiveness of each component in the proposed GRNet by conducting the following ablation studies on DeepFashion dataset [34], shown in Table 5.

Graph reasoning. To validate the effectiveness of graph reasoning, we utilize a GRNet without graph reasoning as our baseline(#1), which computes the global similarity between global features. Comparing #1 and #7, our graph reasoning acquires 11.6% improvement on the top-1 accuracy.

Inter-scale connections. Comparing #6 and #7, it can be observed that the proposed GRNet can achieve 1.15% performance gain on the top-1 accuracy by adding the inter-scale connections (Noted that #6 and #4 keep the connections between the global similarity and the local similarities, but removes the connections between different scales).

Intra-scale connections. As reported in Table 5, by propagating similarities at the same scale, our intra-scale connections acquire 0.9% improvement on the top-1 accuracy (#5 vs #7). It shows that the local similarities are also refined by their interactions at the same scale.

Multi-scale similarities. Comparing #1, #2, #3 and # 7, we observe that the performance is consistently improved when using more scale similarities. Specifically, the accuracy is improved from 14%, 47% and 60% to 22%, 62% and 73% at top-1, top-20, and top-50 after adding 2×1, 1×2, and 2×2. They are improved slightly by further adding 1×3, 3×1 and 3×3 similarities. Moreover, we compare the results of different scale levels of local similarity. Comparing #2 and #3, the fine scale brings very subtle improvement. The result shows that the multi-scale similarities can enhance the global similarity representation.

Figure 5: Examples of the up-weighted nodes in our similarity pyramid graph. Each node represents one similarity of the local patch (indicated by red rectangles) pair from the query (the top row) and the gallery (the bottom row). Each 2×2 images in one black rectangle show one query-gallery image pair and their up-weighted local patch pairs, where the left column shows the most important node before the similarity reasoning and the right shows it after the similarity propagation. GRNet can up-weight the similarity between aligned salient clothing components (*e.g.*, logo) after graph reasoning.

Layer number of graph convolution. We conduct experiments with different number of graph convolutional layers. The top-1 accuracy increases from 16.8%, 22.8%, to 25.7% when the number of graph convolutional layer is set to 1, 2, and 3. We observe a performance drop if the layer number is increased further due to over-fitting. Thus, we fix the graph convolutional layer number to 3.

Projection dimension and channel number in graph CNN. Table 6 evaluates GRNet with different projection dimension D and channel number $C^{'}$. It has been observed that GRNet is insensitive to projection dimension and channel number. Except $D = 128$, there is no obvious performance drop. We fix $D = 512$ and channel number $C^{'}$ to 128 in all our experiments except otherwise noted.

5.6. Results on FindFashion

We evaluate the proposed GRNet on our annotated benchmark FindFashion with four evaluation protocols. Namely, *Easy*, *Hard-View*, *Hard-Cropping*, and *Hard-Occlusion*. We also compare it with DREML [54], KPM [43] and our baseline in Table 7. Our GRNet improves the results of the top-20 accuracy up to 65.1 on *Easy*, 57.9 on Hard-View, 35.0 on Hard-Occlusion and 48.4 on Hard-Croping. Comparing with the results of KPM [43] which uses the same backbone as ours, GRNet acquires more improvement on the evaluation protocols of Easy, Hard-View, Hard-Occlusion and Hard-Croping. It demonstrates the proposed method's superiority and capability to take full advantages of different scales information to boost the retrieval performance.

6. Conclusions

In this paper, we focus on a real-world application task of customer-to-shop clothes retrieval and have proposed a Graph Reasoning Network (GRNet), which first represents the multi-scale regional similarities and their relationships as a graph and then perform graph CNN based reasoning over the graph to adaptively adjust both the local and global similarities. GRNet implicitly achieves alignment and more precise matching of salient clothing components through information propagation among nodes of similarities. To facilitate future research, we have also introduced a new benchmark called FindFashion, which contains rich annotations of clothes including bounding boxes, views, occlusions, and cropping. Extensive experimental results show that our proposed method obtains new state-of-the-art results on both the existing datasets and FindFashion.

Acknowledgement This work was supported in part by Beijing Municipal Science and Technology Commission (Grant No. Z181100008918004), and National Natural Science Foundation of China (Grant No. 61702565).

References

[1] Fashionai Dataset. http://fashionai.alibaba.com/datasets. 3

[2] Relja Arandjelović, Petr Gronat, Akihiko Torii, Tomas Pajdla, and Josef Sivic. NetVLAD: CNN architecture for Weakly Supervised Place Recognition. In *CVPR*, 2016. 3

[3] Lukas Bossard, Matthias Dantone, Christian Leistner, Christian Wengert, Till Quack, and Luc Van Gool. Apparel Classification with Style. In *ACCV*, pages 321–335, 2012. 3

[4] Sabri Boughorbel and Jean-philippe Tarel. Non-Mercer Kernels for SVM Object Recognition. In *BMVC*, 2014. 4

[5] Huizhong Chen, Andrew Gallagher, and Bernd Girod. Describing Clothing by Semantic Attributes. In *ECCV*, 2012. 3

[6] Zhenfang Chen, Zhanghui Kuang, Kwan-Yee K. Wong, and Wayne Zhang. Aggregated Deep Feature from Activation Clusters for Particular Object Retrieval. In *ACM MM Thematic Workshops*, 2019. 3

[7] Zhenfang Chen, Zhanghui Kuang, Wayne Zhang, and Kwan-Yee K. Wong. Learning Local Similarity with Spatial Relations for Object Retrieval. In *ACM MM*, 2019. 3

[8] Zhi-Qi Cheng, Xiao Wu, Yang Liu, and Xian-Sheng Hua. Video2Shop: Exact Matching Clothes in Videos to Online Shopping Images. In *CVPR*, pages 4169–4177, 2017. 1, 3, 4

[9] Charles Corbière, Hedi Ben-Younes, Alexandre Ramé, and Charles Ollion. Leveraging Weakly Annotated Data for Fashion Image Retrieval and Label Prediction. In *ICCV Workshop*, 2017. 1, 3, 4

[10] Wei Di, Catherine Wah, Anurag Bhardwaj, Robinson Piramuthu, and Neel Sundaresan. Style Finder: Fine-Grained Clothing Style Detection and Retrieval. In *CVPRW*, 2013. 3

[11] David Duvenaud, Dougal Maclaurin, Jorge Aguileraiparraguirre, G Rafael, Timothy Hirzel, and Ryan P Adams. Convolutional Networks on Graphs for Learning Molecular Fingerprints. In *NIPS*, 2015. 3

[12] Jianlong Fu, Jinqiao Wang, Zechao Li, Min Xu, and Hanqing Lu. Efficient Clothing Retrieval with Semantic-preserving Visual Phrases. In *ACCV*, pages 420–431, 2012. 3

[13] Noa Garcia and George Vogiatzis. Dress Like a Star: Retrieving Fashion Products from Videos. In *ICCVW*, pages 2293–2299, 2017. 1, 3, 4

[14] Ross Girshick. Fast R-CNN. In *ICCV*, pages 1440–1448, 2015. 1

[15] Albert Gordo, Jon Almazán, Jerome Revaud, and Diane Larlus. Deep Image Retrieval: Learning Global Representations for Image Search. In *ECCV*, pages 241–257, 2016. 3

[16] Albert Gordo, Jon Almazan, Jerome Revaud, and Diane Larlus. End-to-end Learning of Deep Visual Representations for Image Retrieval. *IJCV*, 124(2):237–254, 2017. 3

[17] Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, and Rogerio Schmidt Feris. Dialog-based Interactive Image Retrieval. In *NIPS*, pages 1–15, 2018. 3

[18] 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 *ICCV*, pages 1472–1480, 2017. 3

[19] Kaiming He and Ross Girshick. Mask R-CNN. In *arXiv preprint arXiv:1703.06870*, 2017. 1

[20] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Delving Deep into Rectifiers : Surpassing Human-Level Performance on ImageNet Classification. In *ICCV*, 2015. 6

[21] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition. In *CVPR*, pages 770–778, 2016. 1

[22] Junshi Huang, Rogerio Feris, Qiang Chen, and Shuicheng Yan. Cross-domain Image Retrieval with a Dual Attribute-aware Ranking Network. In *ICCV*, pages 1062–1070, 2015. 1, 2, 3, 4

[23] Naoto Inoue, Edgar Simo-Serra, Toshihiko Yamasaki, and Hiroshi Ishikawa. Multi-label Fashion Image Classification with Minimal Human Supervision. In *ICCVW*, pages 2261–2267, 2017. 3

[24] Xin Ji, Wei Wang, Meihui Zhang, and Yang Yang. Cross-Domain Image Retrieval with Attention Modeling. In *ACM MM*, pages 1654–1662, 2017. 1, 3, 4

[25] 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 *ICCV*, pages 3343–3351, 2015. 1, 2, 3, 4, 5, 6, 7

[26] Wonsik Kim, Bhavya Goyal, Kunal Chawla, Jungmin Lee, and Keunjoo Kwon. Attention-based Ensemble for Deep Metric Learning. In *CVPR*, 2018. 4

[27] Thomas N. Kipf and Max Welling. Semi-supervised Classification with Graph Convolutional Networks. In *ICLR*, pages 1–14, 2017. 3, 5

[28] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet Classification with Deep Convolutional Neural Networks. In *NIPS*, pages 1097–1105, 2012. 1

[29] Yin Hsi Kuo and Winston H. Hsu. Feature Learning with Rank-Based Candidate Selection for Product Search. In *ICCVW*, pages 298–307, 2017. 3

[30] Yujia Li, Richard Zemel, Marc Brockschmidt, and Daniel Tarlow. Gated Graph Sequence Neural Networks. In *ICLR*, 2016. 3

[31] Xiaodan Liang, Liang Lin, Xiaohui Shen, Jiashi Feng, Shuicheng Yan, and Eric P Xing. Interpretable Structure-Evolving LSTM. In *CVPR*, 2017. 3

[32] Xiaodan Liang, Xiaohui Shen, Jiashi Feng, Liang Lin, and Shuicheng Yan. Semantic Object Parsing with Graph LSTM. In *ECCV*, 2016. 3

[33] Wen Hua Lin, Kuan-Ting Chen, Hung Yueh Chiang, and Winston Hsu. Netizen-Style Commenting on Fashion Photos: Dataset and Diversity Measures. In *arXiv preprint*, 2018. 3

[34] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. DeepFashion: Powering Robust Clothes Recognition and Retrieval with Rich Annotations. In *CVPR*, pages 1096–1104, 2016. 1, 2, 3, 4, 5, 6, 7, 8

[35] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-Scale Image Retrieval with Attentive Deep Local Features. In *ICCV*, 2017. 3

[36] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. BIER - Boosting Independent Embeddings Robustly. In *ICCV*, volume 2017, pages 5199–5208, 2017. 4

[37] Michael Opitz, Georg Waltner, Horst Possegger, and Horst Bischof. BIER: Boosting Independent Embeddings Robustly. In *ICCV*, 2017. 4

[38] Filip Radenović, Giorgos Tolias, and Ondrej Chum. CNN Image Retrieval Learns from BoW: Unsupervised Fine-tuning with Hard Examples. In *ECCV*, pages 3–20, 2016. 3

[39] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards Real-time Object Detection with Region Proposal Networks. In *NIPS*, pages 91–99, 2015. 1, 5

[40] Michael Schlichtkrull, Thomas N Kipf, Peter Bloem, Ivan Titov, and Max Welling. Modeling Relational Data with Graph Convolutional Networks. In *European Semantic Web Conference*, 2018. 3

[41] Ramprasaath R. Selvaraju, Michael Cogswell, Abhishek Das, Devi Vedantam, Ramakrishna Parikh, and Dhruv Batra. Visual Explanations from Deep Networks via Gradient-based Localization. In *ICCV*, 2017. 7

[42] Yantao Shen, Hongsheng Li, Shuai Yi, Dapeng Chen, and Xiaogang Wang. Person Re-identification with Deep Similarity-Guided Graph Neural Network. In *ECCV*, pages 1–20, 2018. 3

[43] Yantao Shen, Tong Xiao, Hongsheng Li, Shuai Yi, and Xiaogang Wang. End-to-end deep kronecker-product matching for person re-identification. In *CVPR*, June 2018. 2, 6, 7, 8

[44] Yang Song, Yuan Li, Bo Wu, Chao Yeh Chen, Xiao Zhang, and Hartwig Adam. Learning Unified Embedding for Apparel Recognition. In *ICCVW*, pages 2243–2246, 2017. 1, 3, 4

[45] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *CVPR*, 2015. 5, 6

[46] Damien Teney, Lingqiao Liu, and Anton van Den Hengel. Graph-Structured Representations for Visual Question Answering. In *CVPR*, pages 1–9, 2017. 3

[47] Giorgos Tolias, Ronan Sicre, and Hervé Jégou. Particular Object Retrieval with Integral Max-pooling of CNN Activations. In *ICLR*, 2016. 3

[48] Christian Wallraven and Barbara Caputo. Recognition with Local Features: the Kernel Recipe. In *ICCV*, 2003. 4

[49] Xiaolong Wang and Abhinav Gupta. Videos as Space-Time Region Graphs. In *ECCV*, 2018. 3

[50] Xianwang Wang and Tong Zhang. Clothes Search in Consumer Photos via Color Matching and Attribute Learning. In *ACM MM*, 2011. 3

[51] Zhouxia Wang, Tianshui Chen, Jimmy Ren, Weihao Yu, Hui Cheng, and Liang Lin. Deep Reasoning with Knowledge Graph for Social Relationship Understanding. In *IJCAI*, 2018. 3

[52] Zhouxia Wang, Tianshui Chen, Ruijia Xu, and Liang Lin. Multi-label Image Recognition by Recurrently Discovering Attentional Regions. In *ICCV*, 2017. 3

[53] Zhonghao Wang, Yujun Gu, Ya Zhang, Jun Zhou, and Xiao Gu. Clothing Retrieval with Visual Attention Model. In *IEEE Visual Communications and Image Processing*, 2017. 2, 3, 6

[54] Hong Xuan, Richard Souvenir, and Robert Pless. Deep Randomized Ensembles for Metric Learning. In *ECCV*, pages 1–12, 2018. 2, 4, 6, 7, 8

[55] Artem Babenko Yandex and Victor Lempitsky. Aggregating Local Deep Features for Image Retrieval. In *ICCV*, 2015. 3

[56] Yuhui Yuan, Kuiyuan Yang, and Chao Zhang. Hard-Aware Deeply Cascaded Embedding. In *ICCV*, pages 814–823, 2017. 4

[57] Roshanak Zakizadeh, Michele Sasdelli, Yu Qian, and Eduard Vazquez. Improving the Annotation of DeepFashion Images for Fine-grained Attribute Recognition. In *arXiv preprint*, 2018. 3

[58] Yanhao Zhang, Pan Pan, Yun Zheng, Kang Zhao, Yingya Zhang, Xiaofeng Ren, and Rong Jin. Visual Search at Alibaba. In *ACM SIGKDD*, pages 993–1001, 2018. 1, 3, 4

[59] Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan. Memory-augmented Attribute Manipulation Networks for Interactive Fashion Search. *CVPR*, pages 6156–6164, 2017. 3

[60] Shuai Zheng, Fan Yang, M. Hadi Kiapour, and Robinson Piramuthu. ModaNet: A Large-Scale Street Fashion Dataset with Polygon Annotations. In *ACM MM*, pages 22–26, 2018. 3