Memory-Augmented Attribute Manipulation Networks for Interactive Fashion Search

Bo Zhao[1,2] Jiashi Feng[2] Xiao Wu[1] Shuicheng Yan[3,2]
[1]Southwest Jiaotong University [2]National University of Singapore [3]360 AI Institute
zhaobo@my.swjtu.edu.cn, elezhf@nus.edu.sg, wuxiaohk@swjtu.edu.cn, yanshuicheng@360.cn

Abstract

We introduce a new fashion search protocol where attribute manipulation is allowed within the interaction between users and search engines, e.g. manipulating the color attribute of the clothing from red to blue. It is particularly useful for image-based search when the query image cannot perfectly match user's expectation of the desired product. To build such a search engine, we propose a novel memory-augmented Attribute Manipulation Network (AMNet) which can manipulate image representation at the attribute level. Given a query image and some attributes that need to modify, AMNet can manipulate the intermediate representation encoding the unwanted attributes and change them to the desired ones through following four novel components: (1) a dual-path CNN architecture for discriminative deep attribute representation learning; (2) a memory block with an internal memory and a neural controller for prototype attribute representation learning and hosting; (3) an attribute manipulation network to modify the representation of the query image with the prototype feature retrieved from the memory block; (4) a loss layer which jointly optimizes the attribute classification loss and a triplet ranking loss over triplet images for facilitating precise attribute manipulation and image retrieving. Extensive experiments conducted on two large-scale fashion search datasets, i.e. DARN and DeepFashion, have demonstrated that AMNet is able to achieve remarkably good performance compared with well-designed baselines in terms of effectiveness of attribute manipulation and search accuracy.

1. Introduction

Image based *fashion search* is to retrieve fashion products based on query images that reflect users' needs, such as those containing clothes [14, 22, 23, 24, 32, 36], shoes [18, 34, 36] or bags [36]. Image based retrieval provides a convenient way for users to search for products with rich details that are hard to verbally and comprehensively describe.

However, in many scenarios, users may also want to change the available query images interactively in order to match their specific mental models better. The WittleSearch [18] partially addresses this problem by providing additional relative attribute description, such as "more bright in color", in an iterative manner. One drawback of this method is that the user cannot easily change the attribute from one to another. More recently, the Generative Visual Manipulation model [36] performs the attribute manipulation by editing the query image using several predefined operations, *i.e.* coloring, sketching, and warping, which requires more interactions with the search engine.

Figure 1. Fashion search with attribute manipulation. The user provides a clothing image with additional description about wanted attributes yet not included in the image. The search engine manipulates the representation of the query image to incorporate the user's desired attributes and retrieves images in the gallery.

For building a more precise and convenient fashion search engine, we consider a new search protocol which takes advantage of textual description and visual editing, that is to retrieve fashion products through exploiting a query image plus descriptions on how to change certain visual properties of the query image. More specifically, with this search engine, in addition to a query image, a user is also allowed to provide extra keywords to describe the desired product attributes that are absent in the query image. A usage case is provided in Figure 1. Based on the query image (a pink dress with round collar), the girl also tells the search engine her desired color (one type of attributes) is "blue" instead of "pink", and desired collar type is "plush" (another type of attributes) instead of "round". Beyond be-

ing capable of calculating visual similarity, the search engine can also understand the request that the color and collar type attribute should be changed in the retrieved products. We call this new search problem as *fashion search with attribute manipulation*.

The most significant challenge introduced by this new search problem is how to change a specific attribute of a query image to the desired one without modifying others. Intuitively, the prior knowledge about the desired attribute can provide powerful guidance for such modification. Inspired by the work of Long Short-Term Memory [13] and Neural Turing Machine [11], which both have a memory to store information and maintain a long/short term dependence between elements in the input sequence, we propose a novel memory-augmented Attribute Manipulation Network (AMNet) that is capable of manipulating the image representation at attribute level. AMNet learns image representation that is discriminative for attributes and introduces an internal memory block to memorize "typical" representations of various attributes. Then given a query image and attribute manipulation request, AMNet updates the query representation by "replacing" specific attribute representation with the desired one which can be read from its memory. Then the retrieval becomes easy: the product with desired attributes can be found through comparison between the updated query representation and the gallery image representations. To ensure the AMNet to precisely manipulate image representation, the new network architecture and the training scheme are carefully designed, as explained below.

Given a query image and the target attribute users want to manipulate, the proposed AMNet first learns representation of the query image that is discriminative for attribute prediction. Meanwhile, the prototype representation for the specified attribute is retrieved from the memory, which dynamically memorizes the learned prototype attribute representations for all attributes. AMNet accesses this attribute memory block through a neural controller that is responsible for retrieving the specific prototype representation and updating the memory during the training process. Then, AMNet modifies the intermediate representation, embedding the specified attribute to the desired one without changing others. To enable AMNet to manipulate the attribute representation precisely, besides the memory block, we also introduce a multi-branch architecture and a triplet ranking objective for better retrieval representation learning. Finally the modified representation is employed for computing its similarity with all the gallery images, and the returned images with desired attributes are obtained. Extensive experiments have validated the effectiveness of our proposed approach. The main contributions are summarized as follows:

(1) We propose a novel memory-augmented attribute manipulation network for interactive fashion search, which is experimentally proved to be effective and efficient. The attributes representation learning, manipulating, and visual similarity constraint are integrated into the model to learn the retrieval feature uniformly.

(2) We design a memory block consisting of a memory and a neural controller in the proposed networks. The memory stores the prototype attribute representations for different attributes, and the controller has a well designed interaction mechanism with the memory to facilitate the attribute manipulation.

(3) We exploit a joint optimization method for attribute representation and retrieval feature learning. Multiple fully-connected layers are added into the proposed model to comprehensively capture the attribute information. Meanwhile, the triplet loss objective guides the networks to learn better retrieval representation.

2. Related Work

Attribute Recognition. Attribute is an informative and compact representation for visual data. Attribute-related research has gained increasing popularity in recent years in many computer vision tasks ranging from zero-shot learning [21, 26, 27], image retrieval [17, 18, 29] to fine-grained recognition [8, 20, 35]. Early works on attribute modeling rely on hand-crafted features such as SIFT [25] and HOG [37], and recently deep convolutional networks are used to learn attribute representation and achieves superior performance [7, 9, 28, 35]. In terms of clothing attributes, Chen *et al*. [4] proposed a fully automated system to generate a list of nameable attributes for clothing on human body; Vittayakorn *et al*. [33] proposed automatic attribute discovery from a noisy collection of clothing images and their text descriptions on the Web. Unlike the binary attributes which can be either present or absent, Singh's method [31] can simultaneously localize and rank relative strength of the visual attributes. Similar to [31], Liu *et al*. [24] proposed the FashionNet to jointly predict attributes and landmarks of the clothing image. By integrating the attribute recognition into AMNet, it learns better to manipulate the representation at attribute level.

Clothing Retrieval. Many clothing retrieval approaches [2, 4, 6, 14, 16, 23, 24] have been proposed in recent years, most of which consider the problem from a cross-domain perspective. In some works on attribute based fashion search [2, 4, 14, 24], users are allowed to search visual content based on fine-grained descriptions, such as "blue striped shirt". Though many works proposed, the retrieval with attribute manipulation is rarely explored. Kovashka *et al*. developed the WhittleSearch [18], which allows the user to upload a query image and give description such as "show me shoe images like these, but sportier" to search desired images. They used the concept of relative attributes proposed by Parikh and Grauman [27]

for relevance feedback, thus only the relative description of the attribute can be used to "whittle away" irrelevant portions of the visual feature space. However, it is difficult or needs more iterations to change the attribute from one to another. More recently, Zhu *et al*. proposed the Generative Visual Manipulation (GVM) model [36] to edit the image on the natural image manifold and generate the new query image using a Generative Adversarial Nets [10] for search. Several editing operations, including coloring, sketching and warping, are defined to transfer the original image to different appearance. Generally, more interactions are needed when using GVM, and its retrieval result highly depends on the quality of the generated image. Moreover, some attributes, *e.g*. style or pattern, are difficult to depict, which constrains its usage. Instead of editing the image, we resort to additional attribute description to communicate with the search engine and refine the search result, which is more convenient and efficient.

3. The Attribute Manipulation Networks

3.1. Problem Setup

Visual attributes provide middle-level descriptions (*e.g*. color, style) for products which are specific for certain properties, and of rich semantic meaning and less abstract than category labels. Thus attributes are usually preferred for fashion product search. In this work, we also consider building a fashion search engine based on attribute description. Different from other attribute based search engines [18, 36], we expect ours to be capable of performing attribute manipulation and to provide better interactive experience for users.

We here provide a formal description on the problem of fashion search with attribute manipulation. Suppose we have a pre-defined attribute set consisting of M different attributes. A query clothing image I_q can be described by its associated attributes $(a_1, \cdots, a_{m_1}, \cdots, a_{m_2}, \cdots, a_M)$. A user expects to find another clothing image I_t from the gallery which shares all the attributes with I_q except for certain attribute(s). Take the attributes a_{m_1} and a_{m_2} here for an example. The target image I_t only differs from I_q at values of the attributes a_{m_1} and a_{m_2}, and the expected attribute description of I_t is denoted as $(a_1, \cdots, a_{m_1^*}, \cdots, a_{m_2^*}, \cdots, a_M)$.

Solely using I_q to retrieve visually similar images will not give the desired I_t. Thus further manipulation on the attribute-level representation of I_q during the search process is necessary: the representation of I_q should be properly manipulated so that its attributes a_{m_1} and a_{m_2} are changed to the desired $a_{m_1^*}$ and $a_{m_2^*}$ while keeping others unchanged. We give an example in Figure 2.

To this end, we propose AMNet which is able to manipulate the particular attributes of the query image, and meanwhile learns the representation to retrieve images with desired attributes from the gallery.

Figure 2. Fashion search with attribute manipulation. By manipulating representation of the query image, two unwanted attributes "pink" color and "round" collar are replaced by the desired ones. Then the manipulated representation is used to retrieve matched images from the gallery, which meet the user's expectation better.

3.2. AMNet Architecture Overview

The overall architecture of AMNet is illustrated in Figure 3. It consists of following four components. (1) An image representation learner which is a dual-path CNN, sharing parameters with each other like a Siamese Network [3]. (2) A memory block which includes a memory and a controller within it. The memory stores all the prototype representations of different attributes, while the controller is to interact with the contents of the memory. (3) An attribute manipulator which modifies the original representation extracted by the representation learner based on the prototype representation retrieved from the memory. (4) A loss layer calculating two types of loss functions for training the AMNet. One is the attribute classification loss to encourage the learned representation to be discriminative for attributes and ensure the learned attribute manipulation to be correct; the other one is the triplet ranking loss for learning better retrieval representations. Details of each component are given below.

3.3. Attribute Representation Learning

To learn the discriminative attribute representation which is favorable for attribute-level manipulation, AMNet chooses the deep CNN architecture, *e.g*. Alex [19] or VGGNet [30], as the base networks. Two paths of the selected CNNs are included in the representation learner, which share parameters with each other, one to extract the attribute representation of the reference image, *i.e*. the query image, for later manipulation, and the other for the positive image and the negative image in a triplet when training. The positive image is the one whose attributes perfectly meet the user's requirement, while the negative image is a randomly sampled image dissimilar to the reference image. An example of such triplet images can be seen in Figure 3.

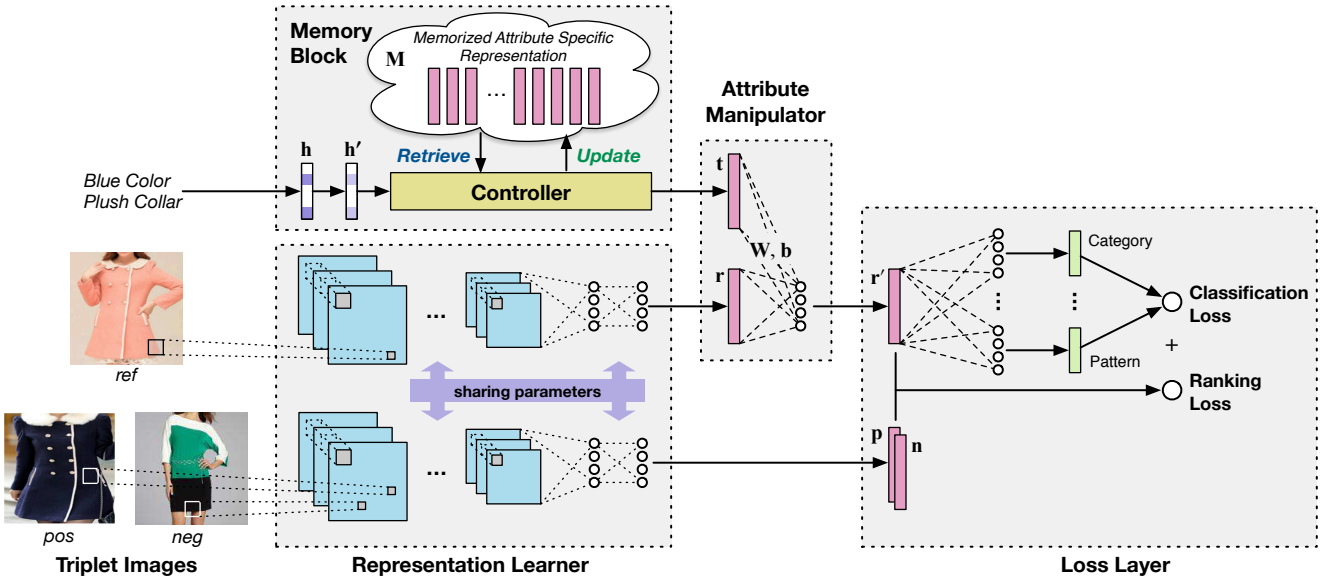


Figure 3. Architecture of AMNet. Triplet images (*i.e.* the *ref*, *pos*, and *neg* images) pass through the dual representation learner with shared parameters. The representation of the *ref* image is then manipulated based on the prototype representation retrieved from the memory block according to the attributes that need to change. Attribute classification loss and the triplet ranking loss are then computed based on the manipulated representation of *ref* image and those of *pos* and *neg* images.

3.4. Memory Block for Attribute Manipulation

AMNet has a memory block which contains an internal memory and a neural controller for facilitating the attribute-specific representation manipulation. The memory stores the learned prototype representations of different attributes, and the controller performs two operations over the memory: retrieve specific prototype attribute representation and update the contents of the memory in the training phase to learn better representations of attributes. A feed-forward network is chosen as the controller in the AMNet. We now introduce the two operations of the controller formally, followed by prototype attribute representation generation.

Memory Addressing. Let $\mathbf{M}$ be the contents within the memory described by a $C \times Q$ memory matrix, where $C = \sum_{m=1}^{M} C_m$ is the number of memory locations. Here M is the attribute number, C_m counts how many different values the m-th attribute can take, and Q is the dimension of each attribute representation. Each row of $\mathbf{M}$ is a prototype attribute representation. Let $\mathbf{h}$ be a binary vector of length C indicating which attribute(s) need to manipulate. For example, if i corresponds to the attribute of "red color" and $\mathbf{h}(i) = 1$ means the target image should be in red color. The attribute manipulation indicator $\mathbf{h}$ is normalized to be a valid probability vector $\mathbf{h}' = h/\sum_{i=1}^{C} h_i$ such that $\sum_{i=1}^{C} \mathbf{h}'(i) = 1, 0 \leq \mathbf{h}'(i) \leq 1, \forall i$. Then, the target prototype attribute representation $\mathbf{t}$ of length Q thus can be retrieved via

$$\mathbf{t} = \sum_{i=1}^{C} \mathbf{h}'(i) \otimes \mathbf{M}(i,:). \tag{1}$$

A larger value within the manipulation indicator $\mathbf{h}'$ makes the controller pay more attention on the corresponding memory location, while an value of zero means the corresponding prototype representation in the memory should be ignored. This attention mechanism allows AMNet to retrieve specific information in memory efficiently and effectively. The combination of the row-vectors within memory is output as the retrieved prototype attribute representation for the later attribute manipulation. Figure 4 illustrates the content addressing and the prototype attribute representation generation over the internal memory.

Memory Updating. It can be easily proved that $\mathbf{t}$ in Equation (1) is differentiable with respect to the memory $\mathbf{M}$. Thus the memory is end-to-end trainable to memorize typical attribute representations after proper training. The contents in the memory are updated during the training by back-propagation and chain rule. The gradients are computed as

$$\nabla \mathbf{M} = \frac{\partial \mathbf{t}}{\partial \mathbf{M}} \nabla \mathbf{t} = \mathbf{h}' \cdot \nabla \mathbf{t}^{\top}, \tag{2}$$

where $\nabla \mathbf{t}$ denotes the gradients passed back by the attribute manipulator. Note that only the specific locations in the memory block will be updated. Updating the memory dynamically along with the networks training allows the memory to capture the most representative attribute representation that is beneficial for the following attribute manipulation.

Memory Initialization. Prototype attributes are critical for attribute manipulation in the proposed AMNet. It pro-

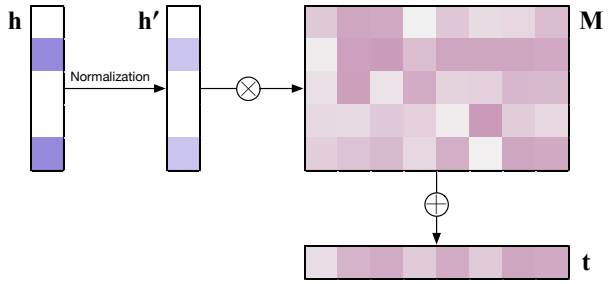


Figure 4. Prototype attribute retrieval process. An attributes manipulation indicator is generated according to the additional attribute description provided by the user. After normalization, the indicator multiplies the contents within memory. Only the locations with corresponding non-zero values in the indicator are focused and weightedly summarized to form the final prototype representation.

vides useful guidance for the attribute manipulator to eliminate the unwanted attributes and insert the new attributes information. A CNN (the same as the one chosen in the representation learner) with multiple fully-connected layers is trained to classify multiple attributes of the image. Then all the representations of the training images with the same attribute value, *e.g*. all the images with red color attribute, are averaged as the prototype representation for a specific attribute value. These prototype representations are initially stored in the memory.

3.5. Attribute Manipulator

Attribute manipulation aims to change the representation of the original image with some undesired attributes. More specifically, it changes the representation related to the target attributes. This can be achieved by fusing the retrieved representation from memory and the current image representation. The fusion is learned through a fully-connected layer in AMNet. It takes in the original image representation $\mathbf{r}$ and the prototype attribute representation $\mathbf{t}$ retrieved from the memory $\mathbf{M}$, and transforms the two representations into a new one which has the same dimension as the original representation and the prototype attribute representation. Formally, the attribute manipulation is defined as

$$\mathbf{r}' = \mathbf{W} \cdot (\mathbf{r}, \mathbf{t}) + \mathbf{b}, \tag{3}$$

where $(\mathbf{r}, \mathbf{t})$ denotes concatenating two feature vectors, and $\mathbf{W}$ and $\mathbf{b}$ are the parameters to transform the concatenated features into the original size as $\mathbf{r}$ and $\mathbf{t}$.

3.6. Loss Layer

Two types of losses are used to train the AMNet: the classification loss to train the AMNet to predict attributes and the ranking loss to learn the retrieval representation. The details of the two losses and how they are used to jointly train AMNet are introduced in the following, respectively.

Classification Loss Computation. Several fully-connected layers are added on top of the attribute manipulator to predict the attributes of the image. Finally, M fully-connected layers (equal to the number of attributes) are added to AMNet. Such a multi-branch structure does not increase the model complexity too much, but makes AMNet better learn the semantic attribute representation and attribute manipulation. The softmax loss is adopted in AMNet, which is defined as

$$L_a = \sum_{i=1}^{N} \sum_{m=1}^{M} -\log(p(a_{im}|ref_i)), \tag{4}$$

where N denotes the number of training samples, a_{im} denotes the ground truth of m-th attribute of i-th ref image, and $p(a_{im}|ref_i)$ encodes the posterior probability of the image ref_i being classified as the attribute label a_{im}. Note that a_{im} may be changed according to the attribute manipulation indicator. We accumulate the loss of M attribute classification branches for each training image.

Ranking Loss Computation. Besides the classification loss, we also impose a ranking loss to learn the retrieval representations of the triplet images according to their relevance of attributes. Specifically, the triplet-based ranking loss is used to constrain the feature similarities of the images in a triplet. The objective function of the triplet ranking loss is defined as

$$L_t = \sum_{i=1}^{N} \max\{0, d(\mathbf{r}'_i, \mathbf{p}_i) - d(\mathbf{r}'_i, \mathbf{n}_i) + m\}, \tag{5}$$

where $d(\cdot, \cdot)$ represents the distance between two features, *e.g*. Euclidean distance, and $\mathbf{r}'_i$, $\mathbf{p}_i$, $\mathbf{n}_i$ denote the (manipulated) representations of the ref, pos and neg image in a triplet. Ideally, we expect the distance from the ref image to any neg image with different attributes is larger than that to the pos image with the same attributes by a certain margin $m > 0$.

Networks Optimization. We integrate the two types of losses, the attributes loss and the triplet loss, through a weighted combination:

$$L = \lambda L_a + (1 - \lambda) L_t, \tag{6}$$

where λ is the weight to control the trade-off between two types of losses. We optimize Equation (6) using the standard stochastic gradient descent with momentum.

4. Fashion Search with Attribute Manipulation

We now describe the implementation details of our propose search engine for clothing images with attribute manipulation.

Training Phase. AlexNet [19] with multiple fully-connected layers to predict the attributes is chosen to learn the prototype attribute representations. Finally, the *fc7* features (4,096-D) extracted from the training images with the same attribute value are averaged to be the prototype representation for a specific attribute value. They are stored in the memory of AMNet initially. We also adopt AlexNet for representation learning in AMNet.

We then generate the training triplets $\langle ref, pos, neg \rangle$ and the attribute manipulation indicators. In each triplet, the first image is the reference image, some of whose attributes need to be manipulated according to its attribute manipulation indicator. The second image is the positive image, whose attributes perfectly meet the user's requirement. The last image is a dissimilar one randomly sampled from the training image set. During training, each mini-batch contains multiple such triplets and the corresponding attributes manipulation indicators. We then calculate the gradients for each loss function (cross-entropy loss and triplet ranking loss) w.r.t. each sample. The gradients from the classification loss and the ranking loss are back propagated to each individual sub-network. The prototype attribute representations in the memory block are also updated by the back propagated errors. Regarding the hyper-parameters, we empirically set the margin m in the triplet loss objective as 0.5 and the weight λ in Equation (6) as 0.2.

Clothing Search. Given a query image and the attribute(s) that need to change, our model manipulates the outputs of the representation learner based on the retrieved prototype representations retrieved from the memory. For the gallery images, the output of the representation learner is directly stored without manipulation. The Euclidean distances between the representations of the query image and gallery images are computed to rank the images.

5. Experiments

5.1. Experiment Setting

Datasets. Despite several existing clothing datasets [1, 4, 5, 12, 14, 23, 24] proposed, the majority only contain a limited number of images or lack attribute annotations. To conduct attribute manipulation, two clothes datasets with enough annotated attributes, *i.e.* DARN [14] and DeepFashion [24], are chosen for our retrieval experiments, which contain around 320,000 and 290,000 clothes images respectively. The DARN has 9 attributes with totally 179 possible values, while the DeepFashion has 6 attributes (including the clothing category) and 1,050 different attribute values. Some example clothing attributes and attribute values are listed in Table 1.

It can be seen that most of the attributes defined in DARN are at middle level, *e.g.* clothes color or sleeve length, which are descriptive and convenient for clothes search. We use all of these attributes for attribute manipulation search. Different from DARN, most attributes defined in DeepFashion are more abstract, like "gaucho" from shape, "free" from style. Clothes with the same high-level attributes may have different middle level attributes such as color, sleeve length, *etc*. These useful middle level attributes are not included in DeepFashion, which increases the difficulties of attribute manipulation. As AMNet focuses on manipulating attributes describing visual patterns, in our experiments on DeepFashion, only the texture attribute is used for manipulation evaluation and the more abstract attributes are left for future investigation.

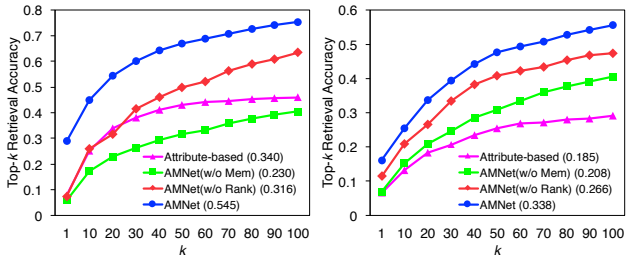
Table 1. Example clothing attributes and values of the DARN and DeepFashion datasets.

Attributes	Values	Total
DARN		
Clothing Color	Black, White, Red, Blue, $\cdots$	56
Clothes Shape	Slim, Straight, Cloak, $\cdots$	10
Sleeve Length	Long, Short, Sleeveless, $\cdots$	7
Collar Type	Round, Lapel, V-Neck, $\cdots$	25
$\cdots$	$\cdots$	
DeepFashion		
Texture	Abstract, Animal, Baroque $\cdots$	156
Shape	Cami, Gaucho, Longline, $\cdots$	180
Style	Athletic, Doodle, Free, $\cdots$	230
$\cdots$	$\cdots$	

From each dataset, we first sample 200 images for each attribute value to train the attribute representation learning networks as described in Section 4. We also reserve 200,000 images from each dataset to construct the retrieval gallery, and 5,000 images with attribute manipulation indicators as the queries to test the retrieval performance. Using the remaining images of each dataset, we generate the training triplets and the attributes manipulation indicators to train the AMNet respectively.

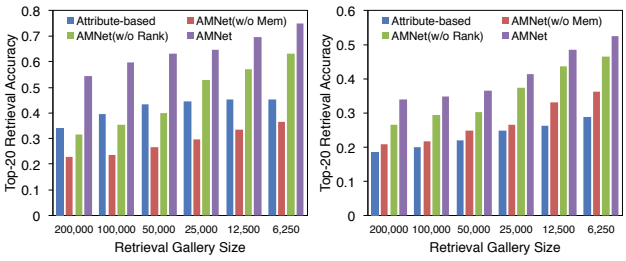
Baselines. Since attribute manipulation search is a new problem, there are no existing works exactly on this problem. A few related works include the WhittleSearch [18] and the more recent Generative Visual Manipulation (GVM) model [36]. However, the WhittleSearch only supports *relative* attributes description instead of replacing attributes, and thus is not suitable for attribute manipulation search. As for GVM, human interactions are expected to "edit" the image using some pre-defined operations. It is time-consuming and subjective to edit the query image manually, and some attributes such as texture are not easy to depict. Thus they are not chosen in the experiments.

Instead, we develop three different models as our baselines. (1) The attribute-based retrieval model, which uses the same CNN chosen in AMNet, *i.e.* AlexNet, to predict attributes of the query image and substitute the unwanted



(a) DARN (b) DeepFashion

Figure 5. Top-k accuracy of one attribute manipulation search on two galleries with 200,000 images. The number in the parentheses is the top-20 retrieval accuracy.



(a) DARN (b) DeepFashion

Figure 6. Top-20 accuracy of one attribute manipulation search on two galleries with different size.

attributes with the desired ones. The *fc7* feature is then used to retrieve the most similar images in the feature space with the desired attributes from the gallery. (2) AMNet without memory block, which directly concatenates the attribute manipulation indicator $\mathbf{h}$ with $\mathbf{r}$ to manipulate the attribute representation. (3) AMNet without ranking loss. To demonstrate generalization ability of the proposed networks, we further evaluate the effectiveness of AMNet w.r.t. different numbers of retrieval results and different gallery sizes.

Evaluation Metrics. Two metrics are used to measure the performance of retrieval models. (1) The top-k retrieval accuracy. We denote a *hit* if the method finds one clothing image with exactly the same attributes as indicated by the query in the top k results; otherwise there is a *miss*. (2) Normalized Discounted Cumulative Gain (NDCG@k) [15] defined as $\frac{1}{Z}\sum_{j=1}^{k}\frac{2^{\text{rel}(j)}-1}{\log(j+1)}$, where rel$(j)$ is the attributes relevance score between the query image and the j-th ranked image, and Z is a normalization constant to ensure that the correct ranking results are with a score of 1. The relevance score rel(j) is defined as the matched attribute number between the desired ones and the j-th ranked image divided by the total number of query attributes.

5.2. One Attribute Manipulation Search Analysis

We report top-k retrieval accuracy results of different methods when manipulating one attribute of the clothing in Figure 5, with varying values of k. We also list the top-20 retrieval accuracy of each model in the parentheses.

The proposed AMNet with both memory-augmentation and ranking loss achieves the best performance, giving 54.5% and 33.8% top-20 accuracy on DARN and DeepFashion, respectively. Removing the memory of AMNet decreases the top-20 retrieval accuracy to 23.0% and 20.8% respectively. We also observe that removing the ranking loss for training AMNet leads to 22.9% and 7.2% drop in top-20 retrieval accuracy. This indicates that both the memory block and the ranking loss contribute significantly to the manipulation ability of AMNet. The performance of the attribute-based retrieval approach is better than the AMNet without memory in DARN, due to good performance of the attribute classification CNN. However, we do not observe this on DeepFashion. This is because the attributes classification on DeepFashion is more difficult than DARN. The attribute-based retrieval approach thus works not so well due to inaccurate attributes prediction.

We give one retrieval example by AMNet and its variants in the left column of Figure 7. The image with green boundary means containing the matched clothing with desired attributes. It can be seen that, without the memory block, the model has difficulties to generate proper representation for retrieving desired images. The retrieved images ranked in higher positions still contain the unwanted plush collar, which indicates the failure of attribute manipulation. Meanwhile, the ranking loss helps to learn better retrieval representations, more matched images retrieved by AMNet than the one without ranking loss, as shown in the second and third row. More attribute manipulation search examples can be seen in the right column of Figure 7.

5.3. Attribute-aware Clothing Search Evaluation

An essential part of the attribute manipulation search is preserving the attributes relevance. Users expect that the retrieved images contain the desired attributes via manipulation and meanwhile the other attributes stay unchanged. With the attributes representation learning of AMNet, the learned features have strong semantic meaning. When using these representations to retrieve the images which are close in the feature space, the retrieval results also present strong attribute-level matching.

We use NDCG to measure the attribute-level matching performance. The score is larger when the images with more matching attributes are ranked higher. We report the NDCG score of the top-20 retrieved images in Table 2. One

Table 2. The NDCG@20 results of one attribute manipulation search on two galleries with 200,000 images.

Model	Mem.	Rank	DARN	DF
Attribute-based			0.33	0.21
AMNet(w/o Mem)		√	0.23	0.23
AMNet(w/o Rank)	√		0.32	0.28
AMNet	√	√	**0.46**	**0.39**

Figure 7. The top-4 retrieval results of DARN and DeepFashion. The query images with the attribute(s) manipulation descriptions are in the first column, followed by the images retrieved from respective 200,000 images gallery.

can observe that adding the memory block increases the NDCG score from 0.23 to 0.46, and 0.23 to 0.39 on the two datasets respectively. While the ranking loss improves the value from 0.32 to 0.46 and 0.28 to 0.39, which contributes less than the memory block. Similar to the performance of top-k accuracy, the attribute-based retrieval approach can retrieve more relevant images in DARN but fails on DeepFashion. Although the attributes prediction for DeepFashion is challenging, AMNet still achieve 0.39 NDCG score.

5.4. Performance with Different Gallery Sizes

To further demonstrate the ability of AMNet to learn robust features, we give the top-20 retrieval accuracy of different retrieval models with varying gallery sizes in Figure 6.

We calculate the accuracy increment ratio with reducing gallery sizes to evaluate the robustness of features. Intuitively, the smaller increase ratio indicates more robustness to gallery size reduction, thus the learned features are more robust. Specifically, on DARN, the top-20 retrieval accuracy of the attribute-based retrieval approach, AMNet without memory, AMNet without ranking, and AMNet increase by 33.0%, 58.9%, 99.4%, and 37.4% from largest retrieval gallery to smallest gallery, respectively. Although the attribute-based approach achieves the smallest increase ratio, its retrieval accuracy is much lower than AMNet. The increase ratio of AMNet is much lower than both AMNet without memory and ranking. On DeepFashion, the increase ratio of AMNet is even slightly lower than the attribute-based approach, which is 55.0% *vs*. 55.8%. The increase ratio of AMNet without memory and without ranking are 74.3% and 74.5%, respectively. This observation verifies that the AMNet can learn robust and effective features than the baselines.

5.5. Two Attributes Manipulation Search Analysis

Besides single attributes manipulation, AMNet can also manipulate multiple attributes at the same time. To demonstrate this point, we conduct the two attributes manipulation retrieval experiments on DARN due to its suitable attribute setting. The top-20 accuracy on the 200,000 images gallery is 46.4% and its NDCG value is 0.41, which shows its effectiveness of two attributes manipulation search. We give one example of two attributes manipulation search in the last row of the right column in Figure 7, which successfully retrieves the images with red color and slim shape. Another way of two or even more attributes manipulation search is by doing one attribute manipulation search iteratively. One attribute of the query image is manipulated at each time. The user chooses a partially matched image from the returns and further manipulates the remaining unwanted attributes.

5.6. System Running Time

Our retrieval system runs on a server with the Intel i7-4939K CPU(@3.4 GHz) and 64 GB RAM memory, with dual NVIDIA TITAN X GPUs. On average, the feature extraction process costs about 11 seconds per 1,000 images. Given a query and the attributes need to manipulate, it cost about 0.2 second for attribute manipulation and clothing retrieval in our retrieval experiment.

6. Conclusions

We presented the memory-augmented attribute manipulation networks for interactive fashion search. Different from the previous approaches, our method can manipulate some unwanted attributes of the image and retrieve the desired images in the gallery. We demonstrated our approach in a practical clothing retrieval application, showing substantial improvement over other baselines.

Acknowledgement This work was supported partially by the National Natural Science Foundation of China under Grant 61373121, the Program for Sichuan Provincial Science Fund for Distinguished Young Scholars under Grant 13QNJJ0149, the National University of Singapore startup grant R-263-000-C08-133, Ministry of Education of Singapore AcRF Tier One grant R-263-000-C21-112, and the China Scholarship Council under Grant 201507000032.

References

[1] L. Bossard, M. Dantone, C. Leistner, C. Wengert, T. Quack, and L. Van Gool. Apparel classification with style. In *ACCV*, pages 321–335, 2012. 6

[2] L. Bourdev, S. Maji, and J. Malik. Describing people: A poselet-based approach to attribute classification. In *ICCV*, pages 1543–1550, 2011. 2

[3] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah. Signature verification using a siamese time delay neural network. *International Journal of Pattern Recognition and Artificial Intelligence*, 7(04):669–688, 1993. 3

[4] H. Chen, A. Gallagher, and B. Girod. Describing clothing by semantic attributes. In *ECCV*, pages 609–623, 2012. 2, 6

[5] Q. Chen, J. Huang, R. Feris, L. M. Brown, J. Dong, and S. Yan. Deep domain adaptation for describing people based on fine-grained clothing attributes. In *CVPR*, pages 5315–5324, 2015. 6

[6] Z.-Q. Cheng, Y. Liu, X. Wu, and X.-S. Hua. Video ecommerce: Towards online video advertising. In *Proceedings of the 2016 ACM on Multimedia Conference*, pages 1365–1374, 2016. 2

[7] J. Chung, D. Lee, Y. Seo, and C. D. Yoo. Deep attribute networks. In *NIPS Workshop*, 2012. 2

[8] K. Duan, D. Parikh, D. Crandall, and K. Grauman. Discovering localized attributes for fine-grained recognition. In *CVPR*, pages 3474–3481, 2012. 2

[9] V. Escorcia, J. C. Niebles, and B. Ghanem. On the relationship between visual attributes and convolutional networks. In *CVPR*, pages 1256–1264, 2015. 2

[10] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014. 3

[11] A. Graves, G. Wayne, and I. Danihelka. Neural turing machines. *arXiv:1410.5401*, 2014. 2

[12] M. Hadi Kiapour, X. Han, S. Lazebnik, A. C. Berg, and T. L. Berg. Where to buy it: Matching street clothing photos in online shops. In *ICCV*, pages 3343–3351, 2015. 6

[13] S. Hochreiter and J. Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997. 2

[14] J. Huang, R. S. Feris, Q. Chen, and S. Yan. Cross-domain image retrieval with a dual attribute-aware ranking network. In *ICCV*, pages 1062–1070, 2015. 1, 2, 6

[15] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002. 7

[16] Y. Kalantidis, L. Kennedy, and L.-J. Li. Getting the look: Clothing recognition and segmentation for automatic product suggestions in everyday photos. In *ICMR*, pages 105–112, 2013. 2

[17] A. Kovashka and K. Grauman. Attribute adaptation for personalized image search. In *ICCV*, pages 3432–3439, 2013. 2

[18] A. Kovashka, D. Parikh, and K. Grauman. Whittlesearch: Image Search with Relative Attribute Feedback. In *CVPR*, pages 2973–2980, 2012. 1, 2, 3, 6

[19] A. Krizhevsky, I. Sutskever, and G. E. Hinton. Imagenet classification with deep convolutional neural networks. In *NIPS*, pages 1097–1105, 2012. 3, 6

[20] N. Kumar, A. C. Berg, P. N. Belhumeur, and S. K. Nayar. Attribute and simile classifiers for face verification. In *ICCV*, pages 365–372, 2009. 2

[21] C. H. Lampert, H. Nickisch, and S. Harmeling. Learning to detect unseen object classes by between-class attribute transfer. In *CVPR*, pages 951–958, 2009. 2

[22] X. Liang, L. Lin, W. Yang, P. Luo, J. Huang, and S. Yan. Clothes co-parsing via joint image segmentation and labeling with application to clothing retrieval. *IEEE Transactions on Multimedia*, 18(6):1175–1186, 2016. 1

[23] S. Liu, Z. Song, G. Liu, C. Xu, H. Lu, and S. Yan. Street-to-shop: Cross-scenario clothing retrieval via parts alignment and auxiliary set. In *CVPR*, pages 3330–3337, 2012. 1, 2, 6

[24] Z. Liu, P. Luo, S. Qiu, X. Wang, and X. Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, pages 1096–1104, 2016. 1, 2, 6

[25] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004. 2

[26] M. Palatucci, D. Pomerleau, G. E. Hinton, and T. M. Mitchell. Zero-shot learning with semantic output codes. In *NIPS*, pages 1410–1418, 2009. 2

[27] D. Parikh and K. Grauman. Relative attributes. In *ICCV*, pages 503–510, 2011. 2

[28] S. Shankar, V. K. Garg, and R. Cipolla. Deep-carving: Discovering visual attributes by carving deep neural nets. In *CVPR*, pages 3403–3412, 2015. 2

[29] B. Siddiquie, R. S. Feris, and L. S. Davis. Image ranking and retrieval based on multi-attribute queries. In *CVPR*, pages 801–808, 2011. 2

[30] K. Simonyan and A. Zisserman. Very deep convolutional networks for large-scale image recognition. In *ILSVRC Workshop*, 2014. 3

[31] K. K. Singh and Y. J. Lee. End-to-end localization and ranking for relative attributes. In *ECCV*, pages 753–769, 2016. 2

[32] A. Veit, B. Kovacs, S. Bell, J. McAuley, K. Bala, and S. Belongie. Learning visual clothing style with heterogeneous dyadic co-occurrences. In *ICCV*, pages 4642–4650, 2015. 1

[33] S. Vittayakorn, T. Umeda, K. Murasaki, K. Sudo, T. Okatani, and K. Yamaguchi. Automatic attribute discovery with neural activations. In *ECCV*, pages 252–268, 2016. 2

[34] Q. Yu, F. Liu, Y.-Z. Song, T. Xiang, T. M. Hospedales, and C.-C. Loy. Sketch me that shoe. In *CVPR*, pages 799–807, June 2016. 1

[35] N. Zhang, M. Paluri, M. Ranzato, T. Darrell, and L. Bourdev. Panda: Pose aligned networks for deep attribute modeling. In *CVPR*, pages 1637–1644, 2014. 2

[36] J.-Y. Zhu, P. Krähenbühl, E. Shechtman, and A. A. Efros. Generative visual manipulation on the natural image manifold. In *ECCV*, pages 597–613, 2016. 1, 3, 6

[37] Q. Zhu, M.-C. Yeh, K.-T. Cheng, and S. Avidan. Fast human detection using a cascade of histograms of oriented gradients. In *CVPR*, pages 1491–1498, 2006. 2