Learning Attribute-driven Disentangled Representations for Interactive Fashion Retrieval

Yuxin Hou*
Aalto University
yuxin.hou@aalto.fi

Eleonora Vig
Amazon
eleonov@amazon.com

Michael Donoser
Amazon
donoserm@amazon.de

Loris Bazzani
Amazon
bazzanil@amazon.de

Abstract

*Interactive retrieval for online fashion shopping provides the ability to change image retrieval results according to the user feedback. One common problem in interactive retrieval is that a specific user interaction (*e.g.*, changing the color of a T-shirt) causes other aspects to change inadvertently (*e.g.*, the retrieved item has a sleeve type different than the query). This is a consequence of existing methods learning visual representations that are semantically entangled in the embedding space, which limits the controllability of the retrieved results. We propose to leverage on the semantics of visual attributes to train convolutional networks that learn attribute-specific subspaces for each attribute to obtain disentangled representations. Thus operations, such as swapping out a particular attribute value for another, impact the attribute at hand and leave others untouched. We show that our model can be tailored to deal with different retrieval tasks while maintaining its disentanglement property. We obtain state-of-the-art performance on three interactive fashion retrieval tasks: attribute manipulation retrieval, conditional similarity retrieval, and outfit complementary item retrieval. Code and models are publicly available*[1].

1. Introduction

Content-based image retrieval has been a fundamental computer vision task for decades. More recently, this task has evolved in the direction of enabling users to provide additional forms of interaction (*e.g.*, sentences, attributes and clicks) along with the query image. Interactive image retrieval [11, 15, 57] is relevant in the context of online shopping, specifically for product categories for which appearance is one of the pre-eminent factors for selection, such as fashion items. In this context, it is not only necessary to train expressive visual representations of images [22, 33, 16, 42, 46], but also to empower the model with the ability of understanding interactions of the user and modify the search results accordingly.

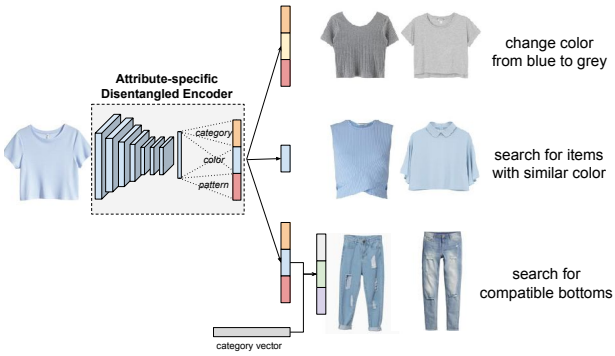

Figure 1: Disentangled representation of images with a subspace for each attribute that can be used for attribute manipulation retrieval, conditional similarity retrieval and outfit complementary item retrieval.

One of the main limitations of existing methods for interactive image retrieval [11, 26, 15, 10, 57, 3, 36, 48] is that representations are semantically entangled in the embedding space. An interaction that involves a specific aspect of the image (*e.g.*, changing the color of a T-shirt) causes other entangled aspects to change inadvertently (*e.g.*, sleeve type or neck style). In our work, we advocate that disentanglement plays a fundamental role in interactive fashion retrieval for obtaining more controllability and interpretability of the search results, and therefore handling the aforementioned limitation. We leverage on the semantics of visual attributes to train convolutional networks that learn attribute-specific subspaces via separate loss functions for each attribute. Our disentangled representations thus consist of the concatenation of attribute-specific embeddings, as shown in Figure 1. In this way, it is possible to apply operators directly on the desired subspace selected by the

*Work done during an internship with Amazon

[1] https://github.com/amzn/fashion-attribute-disentanglement

interaction without affecting the other subspaces. Thus operations, such as swapping out a particular attribute value for another, impact the attribute at hand and leave others untouched.

The proposed disentangled representation is effective on several interactive retrieval tasks as showed in Fig. 1: attribute manipulation retrieval [57, 3], where the interaction comes from a change of attribute; conditional similarity retrieval [45, 35], where the retrieved results are conditioned on specific attributes; and outfit complimentary item retrieval [44, 18, 31]. In the latter problem, we are given an incomplete fashion outfit, and the user can search for items in the missing category (*e.g*. tops or bottoms) that are compatible with the given outfit. Different from previous work where disentanglement is optimized to deal with one specific task [45, 31, 3], our model can be tailored to preserve disentanglement of attribute-specific embeddings for different interactive retrieval tasks.

To perform attribute manipulation, we introduce a memory module which stores the prototype embeddings of each attribute value. The memory module enables our model to swap out the attribute representation of the query image that should be modified with the stored prototype of the desired attribute. This generates a residual embedding which is composed to the attribute-specific representation of the original image to obtain the target representation, which is used for retrieval. In order to allow the manipulation of disentangled representations, we enforce the memory block to be block-diagonal and introduce a memory-block loss to preserve its structure and update the prototype embeddings. Moreover, we introduce a novel visual-semantic consistency loss that aims to align the prototype embedding projected from the attribute label with the embedding extracted from the image. For the conditional similarity retrieval task, we can directly select the subspace related to the conditioning attribute to perform the retrieval. Compared to [45, 35], our method is simpler, yet effective, leading to state-of-the-art results.

Previous work for outfit compatibility and outfit complementary item retrieval [44, 18, 31] does not investigate semantic disentanglement of representations. Our hypothesis is that disentanglement enables to capture complementarity in the different subspaces focusing on specific attributes. In other words, a matching outfit should be composed of items that match along several attributes, *e.g*. in color, style, etc. We tailor our disentangled embedding via learnable attention weights which depend on the category of the query image and of the desired target. Our model shares similarities with [31], however our embedding can be separated and disentangled into attribute-specific subspaces.

To prove the effectiveness of our method, we ran experiments and a thorough ablation study, obtaining state-of-the-art results for: attribute manipulation retrieval, +2.63% top-10 accuracy on Shopping100k [4] and +8.58% on DeepFashion [33]; conditional similarity retrieval, +1.24% accuracy on Shopping100k [4] and +0.58% on Zappos50k [54]; outfit complimentary item retrieval, +1.48% recall at 30 on Polyvore-Outfit [44].

To summarize, our **contributions** are the following: 1) We demonstrate that learning attribute-driven disentangled representations improves controllability and effectiveness of models on different interactive retrieval tasks. 2) We tailor our model to attribute manipulation retrieval while introducing a novel visual-semantic consistency loss and a block-diagonal memory module. 3) We show that disentangled representations can learn conditional similarity for image retrieval and compatibility for outfit complementary item retrieval. 4) We achieve state-of-the-art performance in three different applications.

2. Related Work

Disentangled representations. Disentangled representation learning [1, 6, 34] has recently gained attention due to its properties of robustness, interpretability and controllability required in many applications. Recent unsupervised approaches rely on variants of variational autoencoders [9, 20, 25] and Generative Adversarial Networks [7, 21, 32, 49] (GAN). Other approaches use a supervisory signal in the form of visual attributes to encourage disentanglement of target subspaces [56, 58, 59]. In the fashion domain, disentanglement remains less explored given the challenges of obtaining clean data and annotations. [38] extends correlation networks [8] by initializing the weight matrix of the projection layer to be diagonal in correspondence of three attribute types. [53] proposes a conditional GAN to generate garment images with control of color, texture and shape characteristics. Our method relies on the supervisory signal of fashion attributes to maintain disentanglement in the embedding space, which is tailored to different retrieval tasks.

Attribute manipulation retrieval. Attribute manipulation retrieval of fashion images is a recent research problem. AMNet [57] proposes a memory block with an internal memory and a neural controller, and an attribute manipulation fusion module. FashionSearchNet [3, 2] uses attribute locations to perform the manipulation using attribute activation maps that are trained in a weakly-supervised way. More recent methods rely on GANs to synthesize images while modifying certain attributes [5, 52, 30, 28], but their focus is often on novel image generation rather than retrieval of matching images. AMGAN [52] is trained via a metric learning-based loss function for retrieval, while [41] manipulates image features directly rather than working in the pixel space. Attribute manipulation methods [55, 24, 30] have been proposed when queries are specified in form of natural language modifications. Existing models do not consider disentanglement, *e.g*., AMNet has an attribute ma-

nipulator module which is a fully connected layer and FashionSearchNet learns the global representation that fuses different subspaces and thus introduces entanglement.

Conditional similarity retrieval. Measuring the similarity between images is essential in many retrieval applications [12, 39]. In practice, similarity is often conditioned on a specific property, especially when user interactions are allowed, *e.g*. search for items with similar color. [45] proposes a conditional similarity network that factorizes learned embeddings into distinct latent spaces via learned re-weighted masks. [35] introduces attribute-specific spatial attention and channel attention to learn multiple embedding spaces. [43] proposes a model that learns representations with different notions of similarity without category supervision via a set of parallel similarity condition masks. In comparison with previous works that learn masks to select relevant dimensions, our model can be used directly for retrieval by simply extracting the attribute-specific representation of each individual semantic subspace.

Outfit compatibility. The main objective is to model the compatibility between fashion items in an outfit [44, 50]. Earlier approaches learn a transformation (*e.g*. using Siamese nets) from images into a latent feature space that expresses compatibility [46, 29]. [18] considers the outfit as an ordered sequence of items encoded with a Bi-LSTM to predict the next item. [13] uses graph CNNs to capture relational information between items and model compatibility. Our method is related to the attribute-based fashion compatibility learning approaches [37, 33, 47]. Some of these approaches utilize attributes to model the compatibility in an interpretable way [17, 50, 14]. Since the outfit items belong to different apparel categories, some methods [31, 43, 44] introduce category-specific embedding subspaces. [51] models category relationships via a vector translation operation. For complementary item retrieval, [31] proposes an outfit ranking loss that operates on entire outfits rather than pairs of fashion items. Our method is inspired by [31] with the advantage that our subspaces retain their semantics (attributes) and are kept disentangled in the final embedding via category-specific weights.

3. Method

First, we introduce the architecture for learning attribute-driven disentangled representations in Sec 3.1. Then, we describe novel frameworks which rely on the disentangled representation for three retrieval tasks: attribute manipulation retrieval in Sec 3.2, conditional similarity retrieval in Sec 3.3 and outfit complementary item retrieval in Sec 3.4.

3.1. Attribute-driven Disentanglement

Our objective is to create a model that is able to disentangle semantic factors in different subspaces and is effective for different fashion retrieval tasks. Therefore we consider visual attributes as supervisory signal to guide the learning of disentangled representations. Figure 1 shows the architecture for the attribute-specific representations. We first use a deep CNN, *e.g*. AlexNet [27] or ResNet [19], as the backbone network to encode the representation $\mathbf{f}_n$ of an image I_n. The choice of the backbone comes from previous works that we compare to.

Let us assume we have a predefined list of attributes (*e.g*., color, style, fabric) of length A which we index with the symbol a. Each attribute a is associated with a list of possible attribute values $(v_a^1, v_a^2, \ldots, v_a^{J_a})$, where J_a is the total number of possible values for that attribute. The image representation $\mathbf{f}_n$ is fed into a fully-connected two-layer network for each attribute a which maps $\mathbf{f}_n$ to attribute-specific subspaces $\mathbf{r}_{n,a} = \phi_a(\mathbf{f}_n)$. Then, the representation $\mathbf{r}_{n,a}$ is used to predict the attribute values for the given image via a classification layer made of a fully-connected layer with softmax: $\hat{y}_{n,a} = softmax(\mathbf{r}_{n,a})$.

We supervise the training of such subspaces via independent multi-label attribute-classification tasks defined in the form of a cross-entropy loss as follows:

$$L_{cls} = -\sum_{n=1}^{N}\sum_{a=1}^{A} log(p(y_{n,a}|\hat{y}_{n,a})), \tag{1}$$

where $y_{n,a}$ is the ground-truth label of the image I_n for attribute a, $\hat{y}_{n,a}$ is the output of the softmax layer, and N is the number of samples in the training set.

The disentangled representation of a given image I_n is obtained by concatenating the attribute-specific embeddings $\mathbf{r}_n = (\mathbf{r}_{n,1}, \ldots, \mathbf{r}_{n,A})$, where $\mathbf{r}_n \in \mathbb{R}^{A \cdot d}$ and d is the dimension of each attribute-specific embedding. We call the proposed CNN architecture that extracts such a disentangled representation for an image the Attribute-Driven Disentangled Encoder (ADDE).

3.2. Attribute Manipulation Retrieval

We formulate attribute manipulation retrieval as follows. Given a query image I_q, which has associated attribute values $\mathbf{v}_q = (v_q^1, v_q^2, \ldots, v_q^J)$, the goal is to find a target image I_p, whose attribute description is $\mathbf{v}_p = (v_p^1, v_p^2, \ldots, v_p^J)$, and differs from $\mathbf{v}_q$ only for a subset of selected attributes. Note that to simplify the notation we merged all attribute values for different attributes into a single list $\mathbf{v} = (v^1, v^2, \ldots, v^J)$, where $J = \sum_{a=1}^{A} J_a$. It is always possible to group back the values into attribute-specific subspaces, so that we can maintain their semantics. We use one-hot encoding for each v^j: attribute values present in an image are encoded with 1s, the rest with 0s.

We introduce a **memory block** $\mathcal{M} \in \mathbb{R}^{A \cdot d \times J}$, which enables ADDE to perform attribute manipulation. It stores prototype ADDE embeddings for each attribute value, *e.g*.,

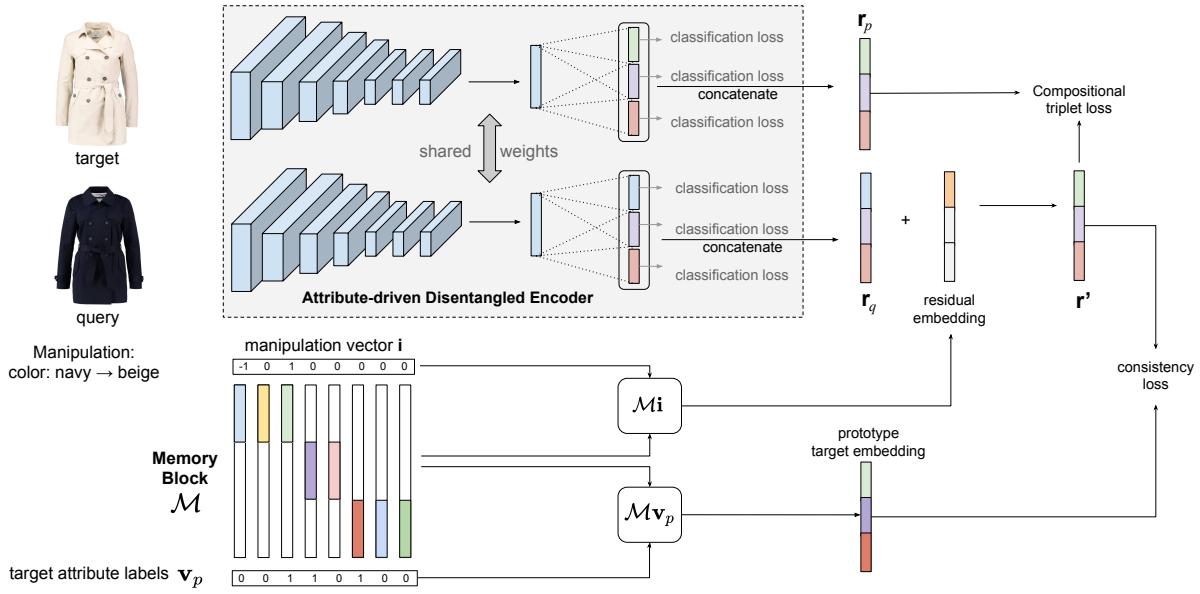

Figure 2: Model for attribute manipulation retrieval. The Attribute-Driven Disentangled Encoder (ADDE) extracts image representations. The memory block $\mathcal{M}$, which stores prototype disentangled embeddings, is combined with the manipulator vector $\mathbf{i}$ and added to the representation of the query image $\mathbf{r}_q$. The compositional representation $\mathbf{r}'$ should be as close as possible to the representation of the target image $\mathbf{r}_p$. See text for more details.

for the color attribute, we will have a prototype embedding for each specific color in the dataset. Inspired by [57], we initialize the memory block by averaging the ADDE embeddings (Sec 3.1) of those training images that have the same attribute value. These representations comprise the initial prototype embeddings and are stored in the columns of the memory block:

$$\mathcal{M} = \begin{pmatrix} \mathbf{e}_1^1 & \dots & \mathbf{e}_1^{J_1} & 0 & \dots & 0 & 0 & \dots & 0 \\ 0 & \dots & 0 & \mathbf{e}_2^1 & \dots & \mathbf{e}_2^{J_2} & 0 & \dots & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & \dots & 0 & 0 & \dots & 0 & \mathbf{e}_A^1 & \dots & \mathbf{e}_A^{J_A} \end{pmatrix},$$

where $\mathbf{e}_a^j$ denotes the prototype embedding for the j-th attribute value of the attribute a.

Fig. 2 depicts the proposed model for attribute manipulation. The query and target images are encoded using ADDE, as in Sec. 3.1, into $\mathbf{r}_q$ and $\mathbf{r}_p$ respectively. To represent the attribute manipulations, we build a manipulation vector $\mathbf{i} = \mathbf{v}_p - \mathbf{v}_q = (i^1, i^2, \dots, i^J)$, where $i \in \{-1, 1, 0\}$ corresponds to removing an attribute value, adding it, or keeping it unchanged. Given the query embedding $\mathbf{r}_q$, the manipulation vector $\mathbf{i}$ and the memory block $\mathcal{M}$, we compute the target compositional representation $\mathbf{r}'$ as:

$$\mathbf{r}' = \mathbf{r}_q + \mathcal{M}\mathbf{i}. \tag{2}$$

The intuition of Eq. 2 is that the original representation $\mathbf{r}_q$ is modified by a subset of prototype attribute-specific representations in $\mathcal{M}$ which are positively or negatively signed by the manipulation vector. This compositional embedding is used to search the database of images to find those images that have the target attribute values specified in the manipulation vector. Technically, our model can deal with the manipulation of multiple attributes based on how the manipulation vector $\mathbf{i}$ is constructed.

During training, we jointly optimize ADDE and the memory block with different loss functions described below.

Memory block loss. During training, the prototypes in the memory block are updated. To ensure that we preserve the disentanglement, we need to enforce the memory block to maintain its block-diagonal structure with off-block-diagonal zeros. We introduce a regularization loss on the non-diagonal elements, which is inspired by [38]:

$$L_{mem} = ||\mathcal{M} \circ \mathcal{N}||_1, \tag{3}$$

$$\mathcal{N} = \mathbf{1}_D - \mathcal{D},\ \mathcal{D} = \begin{pmatrix} \mathbf{1}_{\mathcal{M}_1} & 0 & \dots & 0 \\ 0 & \mathbf{1}_{\mathcal{M}_2} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \mathbf{1}_{\mathcal{M}_A} \end{pmatrix}, \tag{4}$$

where $\mathbf{1}_{\mathcal{M}_a}$ denotes a matrix of ones of size $d \times J_a$, $\circ$ is the element-wise multiplication, $\mathcal{D}$ indicates the elements in diagonal blocks and $\mathcal{N}$ corresponds to the non-diagonal elements. The L1-norm regularization loss helps to curb the mixing of different attribute-specific embeddings during training.

Compositional triplet loss. We introduce a compositional triplet loss to encourage the compositional representation $\mathbf{r}'$ to be close to the positive representations with the desired attributes. Given the query image and a randomly generated manipulation vector, we select a positive sample that has the desired target attribute labels, and randomly choose a negative sample that has different attribute labels. Then, the compositional triplet loss is defined as:

$$L_{ct} = \max(0, d(\mathbf{r}', \mathbf{r}_{p^{ct}}) - d(\mathbf{r}', \mathbf{r}_{n^{ct}}) + m), \quad (5)$$

where $\mathbf{r}_{p^{ct}}$ and $\mathbf{r}_{n^{ct}}$ are the normalized disentangled representations of the positive and negative sample respectively, m is the margin parameter, and $d(\cdot)$ is the L2 distance.

Consistency loss. Because of the diagonal structure of our memory, we can project the attribute label vector into the disentangled embedding space directly. Intuitively, as the attribute label vector and the image characterize the same image, they should encode the same semantic information, hence the representation extracted from the image should be close to the representation projected from the attribute label vector. To this end we introduce a novel loss function that encourages this semantic consistency:

$$L_c = d(\mathbf{r}_q, \mathcal{M}\mathbf{v}_q) + d(\mathbf{r}', \mathcal{M}\mathbf{v}_{p^{ct}}) + d(\mathbf{r}_{n^{ct}}, \mathcal{M}\mathbf{v}_{n^{ct}}), \quad (6)$$

where $\mathbf{v}_q, \mathbf{v}_{p^{ct}}, \mathbf{v}_{n^{ct}}$ are the attribute label vectors of the reference image, positive sample and negative sample generated according to the manipulation task. The consistency loss helps to align the prototype embeddings in the memory block with learned embeddings, which is beneficial for attribute manipulation. On the other hand, the prototype embeddings can be regarded as pseudo-supervision for attribute-specific representation learning.

Label triplet loss. We finally add a last triplet loss that encourages images with the same attributes to have similar representations:

$$L_{lt} = \max(0, d(\mathbf{r}_q, \mathbf{r}_{p^{lt}}) - d(\mathbf{r}_q, \mathbf{r}_{n^{lt}}) + m), \quad (7)$$

where $\mathbf{r}_{p^{lt}}$ and $\mathbf{r}_{n^{lt}}$ are the normalized disentangled representations for the positive and negative samples respectively. The positive samples are those that have the same ground truth attribute labels as the reference images.

The overall loss is the weighted sum of individual losses and is described in the Supplementary material.

Testing. We first extract disentangled representations $\mathbf{r}_n$ for each image to create the index. To perform attribute manipulation retrieval, given a query image I_q and the manipulation vector $\mathbf{i}$, we compute the compositional representation (Eq. 2) and perform KNN search of the index to find the items with the matching modified attributes.

3.3. Conditional Similarity Retrieval

ADDE embeddings described in Sec. 3.1 encode conditional similarity naturally, *i.e*. when searching for images with a given attribute (*e.g*., similar color). In particular, we finetune ADDE for this task using a standard triplet loss. Then, we can directly select the subspace of the query embedding specified by the provided condition and perform KNN search to find the relevant items conditioned on the user-specified attribute(s).

3.4. Outfit Complementary Item Retrieval

Outfit item compatibility can be addressed in the semantic space of attributes. Intuitively, to determine if two items are compatible, we can verify if their attributes are harmonious, *e.g*. if the color blue goes well with yellow, or if the A-shape top fits well with the skinny pants. Therefore, ADDE can be adopted for modeling compatibility.

Figure 3 illustrates the overview of our model. ADDE from Sec 3.1 extracts disentangled representations $\mathbf{r}_n$ of each input image I_n of an outfit. We then add one fully-connected layer for each attribute: $\mathbf{z}_{n,a} = \psi_a(\mathbf{r}_{n,a})$. The category of the item in an outfit and the category of the target item to retrieve are encoded as one-hot vectors (c_r and c_t respectively), concatenated and fed into a fully-connected two-layer network with softmax output as in [31]: $\mathbf{w} = \kappa((c_r, c_t))$. The vector $\mathbf{w} \in \mathbb{R}^A$ contains attentional weights (as in Figure 3) that are multiplied to each attribute embedding to change the focus on specific attributes that are important for the provided target category: $\boldsymbol{\gamma}_{n,a} = w_a \cdot \mathbf{z}_{n,a}$.

Training. We adopt the same procedure proposed in [31], which consists of optimizing the outfit ranking loss that considers distances on entire outfits rather than on single items. Note that, compared to [31], our model retains the attribute semantics in the different output subspaces $\boldsymbol{\gamma}_{n,a}$, and thus preserves disentanglement.

Testing. We create the index by computing the attribute-specific embeddings for each image $\boldsymbol{\gamma}_n = (\boldsymbol{\gamma}_{n,1}, \ldots \boldsymbol{\gamma}_{n,A})$. During retrieval, we compute $\boldsymbol{\gamma}_q$ for each image in the query outfit given its category and the target category. We perform KNN with such representation to retrieve the compatible items for the given outfit. We fuse the ranking scores from items in the same query outfit by taking their average.

4. Experiments

We evaluate our method on three interactive fashion retrieval tasks: attribute manipulation retrieval, conditional similarity retrieval, and outfit complementary item retrieval. Implementation details are provided in the supplementary material.

4.1. Attribute Manipulation Retrieval

We experimented with two datasets commonly-used for the task: Shopping100k [4] and DeepFashion [33]. Shopping100k contains clothing images with 12 attributes and 151 attribute values. We use the same splits provided in

	Shopping100k					DeepFashion				
	Top-10	Top-20	Top-30	Top-40	Top-50	Top-10	Top-20	Top-30	Top-40	Top-50
AMNet [57]	25.62	36.13	42.94	47.71	51.64	14.11	19.39	22.94	25.51	27.58
FSN [3]	38.41	47.44	57.17	61.62	66.70	-	-	-	-	-
ADDE-M w/o L_c	36.67	48.13	55.39	60.54	64.37	21.66	27.28	30.83	33.30	35.24
ADDE-M w/o L_{mem}	39.77	51.95	58.82	63.26	66.72	22.67	27.89	31.11	33.52	35.43
ADDE-M w/o L_{ct}	40.73	52.15	58.41	62.91	66.57	22.82	27.39	30.15	32.23	33.89
ADDE-M w/o L_{lt}	39.56	51.41	57.74	61.91	65.34	22.70	27.95	30.99	33.10	34.93
ADDE-M	**41.17**	**52.93**	**59.81**	**64.10**	**67.29**	**23.60**	**28.58**	**31.52**	**33.98**	**35.91**

Table 1: Top-k retrieval accuracies on Shopping100k and DeepFashion for attribute manipulation. ADDE-M is our method.

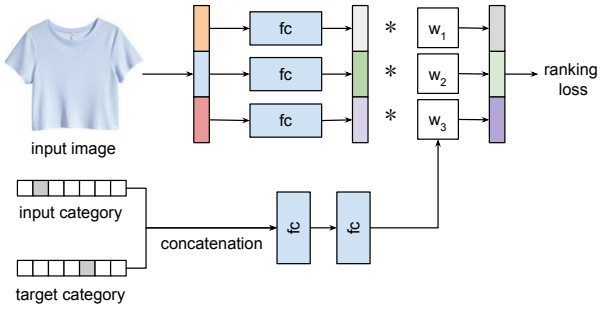

Figure 3: Model for outfit complementary item retrieval. ADDE extracts image representations. The input and target categories are used to predict attentional weights used to compose the target representation.

[3]: 80,586 training images and 20,000 testing images. For DeepFashion, we used 3 out of the 6 attributes (category, texture and shape) as in [3], and removed under-represented attribute values and images that have more than one attribute labels for each type. We are left with 91,285 training images, 22,663 testing images and 202 attribute values. For both datasets, we used 2,000 query images, and for each query every possible manipulation is applied.

We used standard evaluation metrics to measure the performance: 1) top-k retrieval accuracy, defined as the number of "hit" queries divided by the total number of queries. A query is considered a "hit" if there is one image within the returned top-k nearest neighbours that has matching expected attributes; and 2) Normalized Discounted Cumulative Gain (NDCG@k) [23], defined as:

$$\frac{1}{Z}\sum_{j=1}^{k}\frac{2^{rel(j)-1}}{log(j+1)}, \tag{8}$$

where $rel(j)$ is the attribute relevance score for the j-th ranked image defined as the number of matching attributes between the desired label and the ground-truth label of j-th ranked image divided by the total number of attribute types; Z is a normalization constant.

To better measure the ability of preserving attributes that should not be modified, we also compute two variants of the standard NDCG metric: NDCG_t and NDCG_o. Their formula is similar to Eq. 8, the only difference being in how they compute the relevance scores $rel(j)$. NDCG_t focuses specifically on the target attribute to be manipulated, hence $rel(j)$ will only be 0 or 1. On the other hand, NDCG_o only considers the rest of the attributes that should be kept fixed.

We compare our method to two state-of-the-art methods, namely AMNet [57] and FSN [3]. To fairly compare with AMNet, we reimplemented it and experimented with the same train/test splits. We confirmed that results are similar to the ones reported in [3]. We also made sure that all methods use the same backbone (AlexNet) and that the dimensions of final representations are comparable: *e.g.* for Shopping100k, the dimension of AMNet and FSN is 4096 and ours is 4080.

Table 1 reports the top-k retrieval accuracies for the proposed model (named ADDE-M) compared to the state-of-the-art methods. ADDE-M achieves the best performance, with the top-30 accuracy being +2.63% higher than FSN on Shopping100k. Different from FSN, ADDE-M does not use attribute localization or extra parameters to learn global representations, which makes our training simpler. Compared to AMNet, which also utilizes a memory block, ADDE-M shows a significant improvement of +19.14% for top-30 accuracy on Shopping100k and +8.52% on DeepFashion. Table 2 reports results in terms of NDCG metrics with $k = 30$ nearest neighbors. Our method outperforms AMNet[2] on the two datasets using both NDCG_t and NDCG_o variants, indicating that our disentangled representation cannot only perform the attribute manipulation successfully, but also better preserves the rest of the attributes. Figure 4 shows qualitative results for attribute manipulation retrieval.

We further evaluate ADDE-M in an ablation study to demonstrate the effectiveness of the different loss functions L_c, L_{mem}, L_{ct} and L_{lt}. Results in Table 1 show that all loss functions are required to improve performance. This is especially true for the consistency loss L_c and the memory block loss L_{mem}, demonstrating the importance of preserving disentanglement for attribute manipulation retrieval. Interestingly, even without the compositional triplet loss L_{ct}

[2]NDCG is only reported for AMNet and is not used in the FSN paper [3].

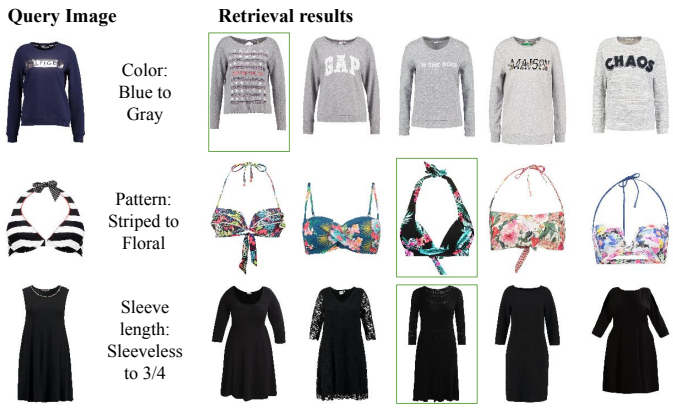

Figure 4: Top-5 retrieval results for attribute manipulation retrieval. The green boxes denote images that match all desired attributes.

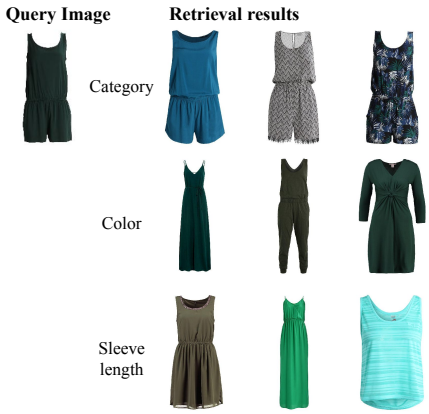

Figure 5: Qualitative results for conditional similarity retrieval.

	Shopping100k		DeepFashion	
	AMNet [57]	ADDE-M	AMNet [57]	ADDE-M
NDCG@30	0.7148	**0.7367**	0.2821	**0.3291**
NDCG_t@30	0.4010	**0.4305**	0.3347	**0.3470**
NDCG_o@30	0.7571	**0.7779**	0.2947	**0.3629**

Table 2: NDCG@30 on Shopping100k and DeepFashion for attribute manipulation.

Method	Shopping100k	Zappos50k
CSN [45]	-	89.27
CSN$^\diamond$ [45]	86.07	89.67
ASEN [54]	-	90.79
ADDE-C	**87.31**	**91.37**

Table 3: Accuracy results for conditional similarity triplet prediction on Shopping100k and Zappos50k. $^\diamond$ denotes that we reproduced the method with the same dimension as ours. ADDE-C is our method.

	Shopping100k		Zappos50k	
	CSN$^\diamond$ [45]	ADDE-C	CSN$^\diamond$ [45]	ADDE-C
MAP@10	90.39	**90.49**	89.25	**90.50**
MAP@30	82.64	**82.98**	81.60	**83.57**
MAP@50	79.63	**80.09**	78.90	**80.96**

Table 4: MAP@k results for conditional similarity retrieval on Shopping100k and Zappos50k.

we obtain better results than FSN, since the consistency loss L_c and the label triplet loss L_{lt} can already guide the modified representations to be close to the target representations.

4.2. Conditional Similarity Retrieval

We experimented on Shopping100k and Zappos50k shoes [54] and considered two tasks: conditional similarity triplet prediction as done in [45] and conditional similarity retrieval. For triplet prediction, we sampled 954,091 training triplets and 118,317 testing triplets for Shopping100k and we use the splits provided by [45] for Zappos50k. We introduce the conditional similarity retrieval task as it represents a more realistic scenario: the objective is to retrieve images having the specified conditional attribute, which is selected from the attributes of the query. For this task, we generated 2,000 queries for each attribute from the test set.

We used prediction accuracy as the performance metric for triplet prediction as done in [45], and additionally Mean Average Precision@k (MAP) [40] for retrieval. We compared our method with CSN [45] and ASEN [54]. To obtain a fair comparison on different datasets and to run experiments on the retrieval task, we reimplemented CSN with the same embedding size as ours ($d = 340$), and all methods use ResNet18 as backbone.

Table 3 reports the accuracy results for triplet prediction on Shopping100k and Zappos50k. One can observe that CSN$^\diamond$ (our implementation) has similar results of the original CSN on Zappos50k. Moreover, our method (named ADDE-C) outperforms all the competitors for both conditional similarity triplet prediction (Table 3) and retrieval (Table 4). It is worth noting that CSN requires both the image and the conditional attribute to obtain the embedding, i.e., multiple embeddings need to be computed by enumerating all conditional attributes. ADDE-C is more efficient since we compute a single embedding per image and then select the subspace specified by the condition. Qualitative results for conditional similarity retrieval are shown in Figure 5.

4.3. Outfit Complementary Item Retrieval

We experimented with the Polyvore-Outfit dataset [44], which contains 11 apparel categories and has two sets (disjoint and non-disjoint). The non-disjoint set contains 53,306 outfits for training and 10,000 outfits for test. The disjoint

Figure 6: Top-10 retrieval results for outfit complementary retrieval. The green boxes denote the target complementary items.

	Polyvore Outfits-D					Polyvore Outfits				
	Retrieval			FITB	Compat.	Retrieval			FITB	Compat.
Method	Rec@10	Rec@30	Rec@50	Acc.	AUC	Rec@10	Rec@30	Rec@50	Acc.	AUC
Type-aware [44]	3.66	8.26	11.98	51.80	0.81	3.50	8.56	12.66	52.90	0.81
SCE-Net average [43]	4.41	9.85	13.87	53.67	0.82	5.10	11.20	15.93	59.07	0.88
CSA-Net [31]	5.93	12.31	17.85	59.26	0.87	**8.27**	15.67	20.91	63.73	0.91
ADDE-O	**6.18**	**13.79**	**18.60**	**60.53**	**0.88**	8.10	**16.02**	**21.57**	**65.16**	**0.93**

Table 5: Results for different tasks on the Polyvore-Outfit dataset. '-D' means the disjoint set. Complementary retrieval is measured with recall@k, FITB with accuracy, while compatibility prediction with AUC.

set is smaller with 16,995 training outfits and 15,154 test outfits. We use the same experimental setup of [31].

We measured the performance for outfit compatibility for three different tasks. 1) *Fill-In-The-Blank (FITB)*: the goal is to select the compatible item given a set of candidate items (four in case of the Polyvore dataset) and the remainder of the outfit, and the performance is evaluated by overall accuracy. 2) *compatibility prediction*: the task is to predict if the items in a candidate outfit are compatible with each other, and the area under the receiver operating characteristic curve (AUC) is used for evaluation. 3) *outfit complementary item retrieval*: the goal is to retrieve the complementary items from the target category, and recall@k is used for evaluation.

We compare our method (named ADDE-O) with three related approaches: Type-aware [44], SCE-Net [43], and the state-of-the-art CSA-Net [31]. For a fair comparison, ADDE-O uses the same backbone network as CSA-Net (ResNet18). ADDE is trained on Shopping100k, since Polyvore-Outfit does not contain attribute information. To ensure transferability of attributes between most apparel categories in Polyvore-Outfit, we selected 5 attributes out of the 12 available: category, color, fabric, fit and pattern.

Table 5 reports the results for the three outfit tasks. One can notice that ADDE-O significantly outperforms Type-aware [44] (+2.52% recall@10) and SCE-Net [43] (+1.77% recall@10) in all three tasks. CSA-Net performs similarly to ADDE-O for recall@10, but we have a significant improvement for recall@30 (+1.48%) and for the FITB task (+1.27%). Qualitative results for outfit complementary retrieval are shown in Figure 6.

5. Discussion and Conclusions

We demonstrated that learning attribute-driven disentangled representations improves controllability and effectiveness in interactive fashion retrieval. We tailored our solution to different tasks (attribute manipulation, conditional similarity and outfit complimentary item retrieval) while introducing novel components which enable the preservation of disengagement. Our experiments show that ADDE-$*$ outperforms the state-of-the-art for each task, confirming the importance of preserving disentanglement. One future direction is to explore the controllability given by disentanglement in the context of more flexible interactive applications, which involve language for example. Moreover, further investigation should focus on unsupervised or semi-supervised learning of disentangled representations, i.e., when attribute supervision is only partially available.

References

[1] Alessandro Achille and Stefano Soatto. Emergence of invariance and disentanglement in deep representations. *The Journal of Machine Learning Research*, 19(1):1947–1980, 2018.

[2] Kenan E Ak, Ashraf A Kassim, Joo Hwee Lim, and Jo Yew Tham. FashionSearchNet: Fashion search with attribute manipulation. In *European Conference on Computer Vision (ECCV) Workshops*, 2018.

[3] Kenan E Ak, Ashraf A Kassim, Joo Hwee Lim, and Jo Yew Tham. Learning attribute representations with localization for flexible fashion search. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7708–7717, 2018.

[4] Kenan E Ak, Joo Hwee Lim, Jo Yew Tham, and Ashraf A Kassim. Efficient multi-attribute similarity learning towards attribute-based fashion search. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 1671–1679, 2018.

[5] Kenan E Ak, Joo Hwee Lim, Jo Yew Tham, and Ashraf A Kassim. Attribute manipulation generative adversarial networks for fashion images. In *IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10541–10550, 2019.

[6] Yoshua Bengio, Aaron Courville, and Pascal Vincent. Representation learning: A review and new perspectives. *IEEE transactions on pattern analysis and machine intelligence*, 35(8):1798–1828, 2013.

[7] Sarthak Bhagat, Vishaal Udandarao, Shagun Uppal, and Saket Anand. Discont: Self-supervised visual attribute disentanglement using context vectors. In *European Conference on Computer Vision (ECCV)*, pages 549–553, 2020.

[8] Sarath Chandar, Mitesh M Khapra, Hugo Larochelle, and Balaraman Ravindran. Correlational neural networks. *Neural computation*, 28(2):257–285, 2016.

[9] Ricky TQ Chen, Xuechen Li, Roger Grosse, and David Duvenaud. Isolating sources of disentanglement in variational autoencoders. *arXiv preprint arXiv:1802.04942*, 2018.

[10] Yanbei Chen and Loris Bazzani. Learning joint visual semantic matching embeddings for language-guided retrieval. In *European Conference on Computer Vision (ECCV)*, 2020.

[11] Yanbei Chen, Shaogang Gong, and Loris Bazzani. Image search with text feedback by visiolinguistic attention learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3001–3011, 2020.

[12] Sumit Chopra, Raia Hadsell, and Yann LeCun. Learning a similarity metric discriminatively, with application to face verification. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 539–546, 2005.

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

[14] Zunlei Feng, Zhenyun Yu, Yongcheng Jing, Sai Wu, Mingli Song, Yezhou Yang, and Junxiao Jiang. Interpretable partitioned embedding for intelligent multi-item fashion outfit composition. *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, 15(2s):1–20, 2019.

[15] Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, Gerald Tesauro, and Rogerio Feris. Dialog-based interactive image retrieval. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 676–686, 2018.

[16] 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 *IEEE International Conference on Computer Vision (ICCV)*, pages 3343–3351, 2015.

[17] Xianjing Han, Xuemeng Song, Jianhua Yin, Yinglong Wang, and Liqiang Nie. Prototype-guided attribute-wise interpretable scheme for clothing matching. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 785–794, 2019.

[18] Xintong Han, Zuxuan Wu, Yu-Gang Jiang, and Larry S Davis. Learning fashion compatibility with bidirectional lstms. In *Proceedings of the 25th ACM international conference on Multimedia*, pages 1078–1086, 2017.

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 770–778, 2016.

[20] Irina Higgins, Loic Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *International Conference on Learning Representations (ICLR)*, 2017.

[21] Qiyang Hu, Attila Szabó, Tiziano Portenier, Paolo Favaro, and Matthias Zwicker. Disentangling factors of variation by mixing them. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3399–3407, 2018.

[22] Junshi Huang, Rogerio S Feris, Qiang Chen, and Shuicheng Yan. Cross-domain image retrieval with a dual attribute-aware ranking network. In *IEEE International Conference on Computer Vision (ICCV)*, pages 1062–1070, 2015.

[23] Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Transactions on Information Systems (TOIS)*, 20(4):422–446, 2002.

[24] E Ak Kenan, Ying Sun, and Joo Hwee Lim. Learning cross-modal representations for language-based image manipulation. In *2020 IEEE International Conference on Image Processing (ICIP)*, pages 1601–1605. IEEE, 2020.

[25] Minyoung Kim, Yuting Wang, Pritish Sahu, and Vladimir Pavlovic. Relevance factor vae: Learning and identifying disentangled factors. *arXiv preprint arXiv:1902.01568*, 2019.

[26] Adriana Kovashka, Devi Parikh, and Kristen Grauman. Whittlesearch: Image search with relative attribute feedback. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2973–2980, 2012.

[27] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In *Advances in Neural Information Processing Systems (NIPS)*, pages 1097–1105, 2012.

[28] Jeong-gi Kwak, David K Han, and Hanseok Ko. Cafe-gan: Arbitrary face attribute editing with complementary attention feature. In *European Conference on Computer Vision (ECCV)*, 2020.

[29] Hanbit Lee, Jinseok Seol, and Sang-goo Lee. Style2vec: Representation learning for fashion items from style sets. *arXiv preprint arXiv:1708.04014*, 2017.

[30] Bowen Li, Xiaojuan Qi, Thomas Lukasiewicz, and Philip HS Torr. ManiGAN: Text-guided image manipulation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7880–7889, 2020.

[31] Yen-Liang Lin, Son Tran, and Larry S Davis. Fashion outfit complementary item retrieval. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3311–3319, 2020.

[32] Yang Liu, Zhaowen Wang, Hailin Jin, and Ian Wassell. Multi-task adversarial network for disentangled feature learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3743–3751, 2018.

[33] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *IEEE conference on computer vision and pattern recognition (CVPR)*, pages 1096–1104, 2016.

[34] Francesco Locatello, Stefan Bauer, Mario Lucic, Gunnar Raetsch, Sylvain Gelly, Bernhard Schölkopf, and Olivier Bachem. Challenging common assumptions in the unsupervised learning of disentangled representations. In *International Conference on Machine Learning (ICML)*, pages 4114–4124. PMLR, 2019.

[35] Zhe Ma, Jianfeng Dong, Zhongzi Long, Yao Zhang, Yuan He, Hui Xue, and Shouling Ji. Fine-grained fashion similarity learning by attribute-specific embedding network. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 11741–11748, 2020.

[36] Long Mai, Hailin Jin, Zhe Lin, Chen Fang, Jonathan Brandt, and Feng Liu. Spatial-semantic image search by visual feature synthesis. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4718–4727, 2017.

[37] Kevin Matzen, Kavita Bala, and Noah Snavely. Streetstyle: Exploring world-wide clothing styles from millions of photos. *arXiv preprint arXiv:1706.01869*, 2017.

[38] Amrita Saha, Megha Nawhal, Mitesh M Khapra, and Vikas C Raykar. Learning disentangled multimodal representations for the fashion domain. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 557–566, 2018.

[39] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 815–823, 2015.

[40] Hinrich Schütze, Christopher D Manning, and Prabhakar Raghavan. *Introduction to information retrieval*, volume 39. Cambridge University Press Cambridge, 2008.

[41] Minchul Shin, Sanghyuk Park, and Taeksoo Kim. Semi-supervised feature-level attribute manipulation for fashion image retrieval. 2019.

[42] Edgar Simo-Serra and Hiroshi Ishikawa. Fashion style in 128 floats: Joint ranking and classification using weak data for feature extraction. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 298–307, 2016.

[43] Reuben Tan, Mariya I Vasileva, Kate Saenko, and Bryan A Plummer. Learning similarity conditions without explicit supervision. In *IEEE International Conference on Computer Vision (ICCV)*, pages 10373–10382, 2019.

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

[45] Andreas Veit, Serge Belongie, and Theofanis Karaletsos. Conditional similarity networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 830–838, 2017.

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

[47] Sirion Vittayakorn, Kota Yamaguchi, Alexander C Berg, and Tamara L Berg. Runway to realway: Visual analysis of fashion. In *IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 951–958. IEEE, 2015.

[48] Nam Vo, Lu Jiang, Chen Sun, Kevin Murphy, Li-Jia Li, Li Fei-Fei, and James Hays. Composing text and image for image retrieval-an empirical odyssey. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6439–6448, 2019.

[49] Taihong Xiao, Jiapeng Hong, and Jinwen Ma. DNA-GAN: Learning disentangled representations from multi-attribute images. *International Conference on Learning Representations (ICLR), Workshop*, 2018.

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

[51] Xun Yang, Yunshan Ma, Lizi Liao, Meng Wang, and Tat-Seng Chua. TransNFCM: Translation-based neural fashion compatibility modeling. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 403–410, 2019.

[52] Xin Yang, Xuemeng Song, Xianjing Han, Haokun Wen, Jie Nie, and Liqiang Nie. Generative attribute manipulation scheme for flexible fashion search. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 941–950, 2020.

[53] Gökhan Yildirim, Calvin Seward, and Urs Bergmann. Disentangling multiple conditional inputs in gans. *arXiv preprint arXiv:1806.07819*, 2018.

[54] A. Yu and K. Grauman. Fine-grained visual comparisons with local learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014.

[55] Feifei Zhang, Mingliang Xu, Qirong Mao, and Changsheng Xu. Joint attribute manipulation and modality alignment

learning for composing text and image to image retrieval. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 3367–3376, 2020.

[56] Jianfu Zhang, Yuanyuan Huang, Yaoyi Li, Weijie Zhao, and Liqing Zhang. Multi-attribute transfer via disentangled representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 9195–9202, 2019.

[57] Bo Zhao, Jiashi Feng, Xiao Wu, and Shuicheng Yan. Memory-augmented attribute manipulation networks for interactive fashion search. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1520–1528, 2017.

[58] Yiru Zhao, Xu Shen, Zhongming Jin, Hongtao Lu, and Xian-sheng Hua. Attribute-driven feature disentangling and temporal aggregation for video person re-identification. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4913–4922, 2019.

[59] Zhenyao Zhu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Multi-view perceptron: a deep model for learning face identity and view representations. *Advances in Neural Information Processing Systems 27 (NIPS)*, 2014.