SAC: Semantic Attention Composition for Text-Conditioned Image Retrieval

Surgan Jandial*†1, Pinkesh Badjatiya*‡2, Pranit Chawla*3, Ayush Chopra*‡4, Mausoom Sarkar1, and Balaji Krishnamurthy1

1Media and Data Science Research Lab, Adobe
2Microsoft, India
3Indian Institute of Technology, Kharagpur
4Media Lab, Massachusetts Institute of Technology

Abstract

The ability to efficiently search for images is essential for improving the user experiences across various products. Incorporating user feedback, via multi-modal inputs, to navigate visual search can help tailor retrieved results to specific user queries. We focus on the task of text-conditioned image retrieval that utilizes support text feedback alongside a reference image to retrieve images that concurrently satisfy constraints imposed by both inputs. The task is challenging since it requires learning composite image-text features by incorporating multiple cross-granular semantic edits from text feedback and then applying the same to visual features. To address this, we propose a novel framework SAC which resolves the above in two major steps: "where to see" (Semantic Feature Attention) and "how to change" (Semantic Feature Modification). We systematically show how our architecture streamlines the generation of text-aware image features by removing the need for various modules required by other state-of-art techniques. We present extensive quantitative, qualitative analysis, and ablation studies, to show that our architecture SAC outperforms existing techniques by achieving state-of-the-art performance on 3 benchmark datasets: FashionIQ, Shoes, and Birds-to-Words, while supporting natural language feedback of varying lengths.

1. Introduction

The ability to search for images over an indexed catalog is a fundamental task that serves as a cornerstone for several allied user experiences like smart, intuitive experiences for online commerce such as fine-grained tagging [45, 1], virtual try-on [20], product recommendations [25] and visual search [5]. The most ubiquitous frameworks in image search either take image or text as input query to search for relevant items [26, 14]. However, a key limitation of these frameworks is the in-feasibility to capture detailed user requirements, either with a single image or a combination of keywords. Correspondingly, several interactive paradigms are being explored, incorporating feedback to help tailor retrieved results to specific user intentions. These interactions involve refining a reference query image through feedback in form of spatial layouts [24], scene-graphs [21, 30] or relative attributes [18, 39]. More recently, text feedback via keywords [37] or short captions [7] are being explored to provide more expressive flexibility to the user during interactive image search [15]. This task is denoted as text-conditioned image retrieval. As shown in Figure 1, the task of text-conditioned image retrieval utilizes a support text feedback alongside a reference image with the objective of retrieving image results that can satisfy constraints imposed by both components of the multi-modal input.

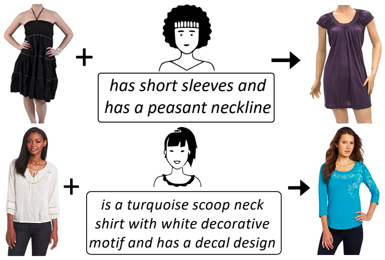
has short sleeves and has a peasant neckline
is a turquoise scoop neck shirt with white decorative motif and has a decal design

Figure 1: Given a *reference image* and a *support text*, we focus on the task of retrieving images that resemble the *reference image* while also satisfying constraints imposed by the *support text*.

*equal contribution
†Work done during the intership at Adobe.
‡Work done while at Media and Data Science Research Lab, Adobe

Broadly, this task requires learning composite image-text representations that transform only the image features relevant to the text modification while preserving the rest. Several works such as [37], [7], [3] have tried to address this issue. First in this domain, TIRG [37] proposed a simple method leveraging gating and residual modules on the average pooled features from the terminal image layers. Adding to this, ComposeAE [3], in their work suggests that TIRG assigns huge importance to image features than the text ones, and hence they propose a novel complex space to learn the composition respectively. Although working well to their capacity, the above methods fail to account for wide range of queries and visual concepts and hence are limited in their performance. Moving further, a recent method VAL [7] proposed to employ multiple composition modules at varying depths rather only the last. Unlike the previous methods that operate on the average pooled features from the last layer, VAL's composition module transforms the entire Image Volume. However, this not only requires complex series of steps but as VAL perturbs the entire Image Volume, it incurs an additional module just to preserve the features of the input image (as required by the task). Thus, posing challenges to performance and the leveraged compute. Moreover, even though VAL [7] composes the image and text at varying depths, it does not account for interactions among features across levels of conv layers.

In this work, we propose *SAC* that resolves the above issues in two major steps: ‘where to see’ and ‘how to change’, and subsequently propose two modules respectively. For example, in Row 1 of Figure 1, the support text requires that the modified image has **short sleeves** and a **peasant neckline**. This implies that the ‘where to see’ operation should focus on 'sleeves' and ‘neckline' whereas ‘how to change' operation should focus on the descriptive attributes of these regions such as ‘short' and ‘peasant'. For the first step (‘where to see'), we introduce a Semantic Feature Attention (SFA) module which effectively computes the salient regions in the image with respect to the text (i.e. regions which need to be modified). Since CNNs learn visual concepts with increasing abstraction ([42],[23]), we thus sample image features over two levels to capture the coarse and fine-grained features. For the second stage, we propose a Semantic Feature Modification (SFM) module that takes as input the two-stage image features along with the text vector to 1) aggregate inter-level features (the coarse and fine-grained features) while ensuring rich representation and 2) modify the resultant according to the text. Further, to focus on several nuances that arise while training composite (Image-Text) features, we propose a unique composition of loss functions.

Our contributions can be summarized as follows,

- We introduce two modules (SFA and SFM) to break down the task of TCIR into two simple steps 'where to see' and 'how to change'.
- We show how our SFA is able to capture salient image regions mentioned in the query and how our SFM module is able to modify these regions according to the query.
- We perform detailed quantitative and qualitative analysis on 3 benchmark datasets and outperform existing state-of-the-art methods.

2. Related Work

Product Search and Image Retrieval attracts significant research interest due to the diverse practical applicability [17]. Conventional works have utilized uni-modal (image or text) queries to retrieve similar [9] or compatible [34] images. More recently, we have witnessed a surge in interactive multi-modal techniques that incorporate user feedback to navigate visual search. The user interactions can manifest in form of attributes [1, 44], spatial layouts [24, 4], sketches [40] and text descriptions [37, 7, 15]. Owing to the ubiquity in existing search engines and flexibility of articulation, using textual support can facilitate fine-grained specificity in user queries. In this work, we pursue the problem of visual search with textual feedback and propose a framework to efficiently handle unconstrained natural language descriptions of varying lengths.

Learning Composite Image-Text Representations involves jointly processing image and text inputs to capture both these contexts effectively and in a way specific to each task. To review a few, we have 1). Visual Question Answering [2, 27] which uses text semantics to localise the image and further generate the answer, 2). Language Grounding [36, 35] which requires spatial localization subject to the input text/phrases , 3). Image-Text matching [32, 38] which searches for an Image given the natural language phrase and vice-versa, among many other tasks digesting both the modalities. To this contrast, learning representation in our task involves incorporating text inputs to selectively modify the relevant image features in a way that ensures the preservation of the unaltered features.

Text Conditioned Image Retrieval has been well explored in several recent works [37, 7, 3]. First in this domain, TIRG [37] introduced a residual gating operation to fuse latent image and text embeddings. Citing drawbacks of TIRG, ComposeAE [3] proposed their novel complex space for robust composition of Image and Text Concepts. Moving one step further, a recent state of the art, VAL [7] proposed to use multiple compositional modules over varying convolutional depths. Briefly, VAL first broadcasts and then fuses the text with image feature to obtain a visiolinguistic representation and then performs self-attention to improve visiolinguistic cues and performs Joint-Attentional Preservation (JAP) to preserve image features. Although effective, VAL in its formulation poses several key limitations: need for complex steps and additional modules affecting performance gains and compute, and in-feasibility to model the interactions/relationship among the features obtained over multiple levels. Addressing the aforementioned, we introduce *SAC*, which tackles the task of TCIR in two steps, first attending to salient regions in a more structured and simplified way and second, by taking into account both inter-level relationships among features across hierarchy and inter-modal relationship between image and text.

3. Approach

Given a query image (I_q) and the modification text description (D_s), the training objective of our task is to learn a Image-Text composite representation that uniquely aligns with the visual representation of the target ground-truth image (I_t). To debrief our overall approach and the components underlying, we divide this section as: Section 3.1 presents an overview and motivation behind our approach, which is then followed by each part of our methodology. Section 3.2 provides our strategy to independently encode the image (I_q, I_t) and text (D_s) inputs. Further, Sections 3.3 to 3.4 delineates different phases of ***SAC***, thus generating visual representation for I_t and the composite Image-Text representation for I_q and D_s. In the last part of this section, we present our unique composition of loss functions (in Sec. 3.5) which are designed to regularize the visual and linguistic features in the composite representations. An overview of the proposed approach is provided in Fig. 2.

3.1. Learning in Two Stages

We intuitively break down the learning of *SAC* in two stages: *'where to look'* and *'how to change'*.

In the first stage, we utilize the **Semantic Feature Attention (SFA)** module to find the salient image regions with respect to the text. On the other hand, VAL [7] in their very first step fuses the image and text features to obtain visiolinguistic representations and further introduces two set of modules: self-attention to improve the obtained visiolinguistic cues and "Joint-Attentional Preservation" (JAP) to preserve the image features which do not have to be modified. Intuitively, they first learn the coarse level visiolinguistic Image Text Relationships which further undergo transformation and preservation to obtain final features required for retrieval.

In contrast to learning any complex visiolinguistic transformation, the only task for our Semantic Feature Attention (SFA) is *given an Image, generates the 2-d probability map that describes the importance of each pixel with respect to text*. Therefore, we use the text vector as a kernel and convolve the entire image volume to obtain a probability matrix. The probability matrix is then applied back to the image volume to reweigh the features in accordance with the text importance, hence *'where to look'*. Intuitively, our method keeps the image volume intact and only alters the regions of interest from the original image volume, thus eliminating the need for a separate preservation module (as used in VAL).

To handle the 'how to change', we propose a Semantic Feature Modification (SFM) Module. Since, we sample two levels of image features, the input to SFM includes two reweighted image features and the text feature vector, which are encapsulated in our novel way to capture relationships across feature levels (coarse and fine) and across modalities (image and text).

3.2. Representing Image and Text

Image Encoder: CNNs are well known to encode visual concepts with increasing abstraction, generally, becoming finer as we progress over levels. Following the similar idea, in our method, we propose to sample out two granularities of embeddings: Low-Level features and High-Level features. Furthermore, in Section 4.4 we provide an analysis on levels as used in our method and related efforts. Concretely, the resultant visual features $\mathcal{F}_q$ and $\mathcal{F}_t$ for the query (I_q) and the target image (I_t) respectively, are computed as,

$$\begin{aligned} \mathcal{F}_q &= \{V_q^1, V_q^2\} = \phi_{\text{CNN}}(I_q) \\ \mathcal{F}_t &= \{V_t^1, V_t^2\} = \phi_{\text{CNN}}(I_t) \end{aligned} \tag{1}$$

Text Encoder: To generate text embeddings corresponding to the visual features at the two granularities we use a GRU [8] followed by *2* parallel fully connected layers. Given the support text D_s (max N words), we obtain a sequence of word-level embedding features $F_{word} \in \mathbb{R}^{1\times768}$ which are then passed through a GRU to obtain the support text feature $\mathcal{F}_{sent} \in \mathbb{R}^{1\times1024}$ as

$$\mathcal{F}_{sent} = \text{GRU}([F^1_{word},\ F^2_{word},\ \cdots\ ,\ F^N_{word}]) \tag{2}$$

We then transform the $\mathcal{F}_{sent}$ through two separate linear projection layers as,

$$\mathcal{T}^1,\ \mathcal{T}^2 = \Omega_1(\mathcal{F}_{sent}),\ \ \Omega_2(\mathcal{F}_{sent}) \tag{3}$$

3.3. Semantic Feature Attention (SFA)

As previously mentioned, the goal of our SFA module is to highlight salient regions in the image which need to be modified according to the text. Our SFA Module is made of two major sub-parts: (1) Attentional Visual Transformation (2) Semantic Pooling. Intuitively, the first one captures the importance of a pixel, subject to the other positional locations within the Image, while the second one captures the importance with respect to text. Formally, we define both these operations below. Since SFA at both levels follows

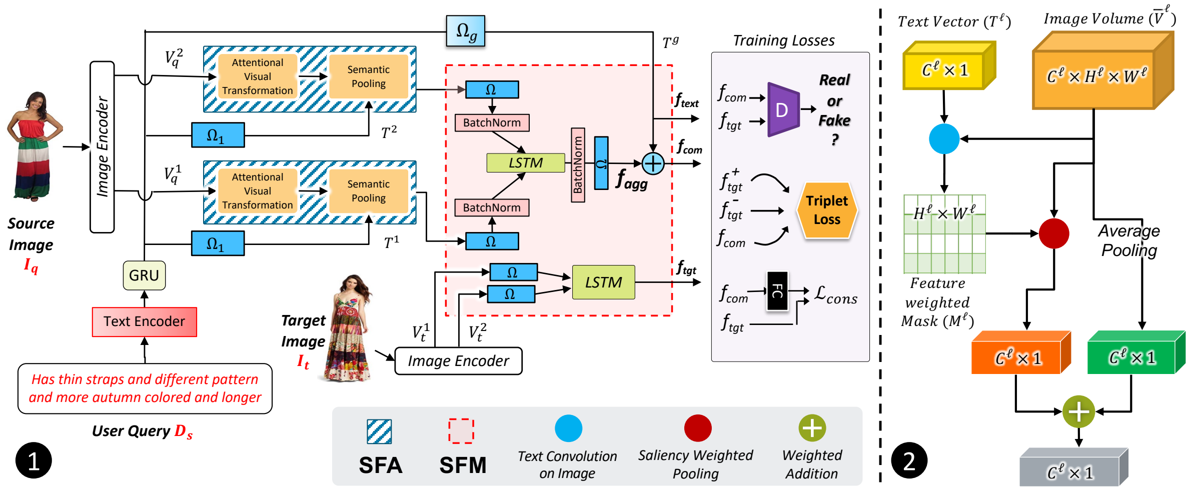

Figure 2: **(1)** Outline of our proposed *SAC* framework. We highlight the 2 main components (shaded): the **Semantic Feature Attention** module, the **Semantic Feature Modification (SFM)** module. **(2)** Schematic representation of the Semantic Pooling component. Some of the operations are denoted by symbols with their description provided in the legend.

the same operations, we use level ℓ in the discussion below for brevity.

Attentional Visual Transformation: Here, we capture apriori long-range contextual relationships within the visual embedding (V_q^ℓ) to help enhancing the representational capabilities. Therefore, we leverage a positional attention mechanism to aggregate the spatial context, [43, 13] to transform V_q^ℓ into volumetric representation $\overline{V}_q^\ell \in \mathbb{R}^{C_\ell \times H_\ell \times W_\ell}$. For this, V_q^ℓ is passed through parallel convolutional layers (denoted by Θ_q, Θ_k, Θ_v) and the obtained volume is reshaped to obtain new query and key feature maps denoted by $(Q^\ell, K^\ell) \in \mathbb{R}^{C_\ell \times N_\ell}$, $N_\ell = H_\ell \times W_\ell$ which are then used to obtain a spatial attention map $\mathcal{A}_{\textbf{self}}^\ell$.

$$\mathcal{Q}^\ell = \Theta_q(V_q^\ell), \quad \mathcal{K}^\ell = \Theta_k(V_q^\ell), \quad \mathcal{V}^\ell = \Theta_v(V_q^\ell)$$
$$\mathcal{A}_{\textbf{self}}^\ell = \text{softmax}((\mathcal{Q}^\ell)^T \mathcal{K}^\ell)$$

We generate an intermediate feature E^ℓ to compute the transformed attentive visual feature map $\overline{V}_q^\ell$ as,

$$E^\ell = \mathcal{V}^\ell (\mathcal{A}_{\textbf{self}}^\ell)^T \quad \text{and} \quad \overline{V}_q^\ell = \beta E^\ell + V_q^\ell \tag{4}$$

The feature vector $\overline{V}_q^\ell$ encodes global visual information along with selectively aggregated spatial context which improves the semantic consistency in the representation.

Semantic Pooling: Further, the learnt attentive visual representation $\overline{V}_q^\ell$ is now convolved with the corresponding text representation T^ℓ to obtain a *2-D saliency map)* $\mathcal{A}_{\textbf{sal}}^\ell \in \mathbb{R}^{H_i \times W_i}$. that essentially gives the importance of each pixel with respect to Text.

$$\mathcal{A}_{\textbf{sal}}^\ell = \overline{V}_q^\ell \circledast T^\ell \tag{5}$$

$\mathcal{A}_{sal}^\ell$ is then passed through softmax, with temperature $\mathcal{T}$, to obtain feature-weightage map (probability map) M^ℓ.

$$M^\ell = \text{softmax}(\mathcal{A}_{\textbf{sal}}^\ell / \mathcal{T}) \tag{6}$$

We provide a clear representation of the operations performed in Figure 2.

We then use the obtained feature-weighted map M^ℓ to pool each channel in the attentional visual feature map $\overline{V}_q^\ell$ to select image features salient to text features, thus generating $S^\ell \in \mathbb{R}^{C_\ell \times 1}$ given as,

$$S^\ell(c) = \sum_{h=1}^{H_\ell} \sum_{w=1}^{W_\ell} M^\ell(h, w) \circledast \overline{V}_q^\ell(c, h, w) \tag{7}$$

where $1 \le c \le \mathcal{C}^\ell$ and $\mathcal{C}^\ell$ denotes the number of channels.

Finally, the *granular* text-conditioned visual embedding O_q^ℓ is obtained by a weighted addition of the Text Conditioned Image feature S^ℓ with pooled attentive visual feature map (we use generalized-mean pooling technique **GeM** [29]). The pooled visual embedding for the target image is also obtained as,

$$O_q^\ell = Pool(\overline{V}_q^\ell) + \gamma S^\ell \quad \text{and} \quad O_t^\ell = Pool(V_t^\ell) \tag{8}$$

The obtained embeddings for the query image O_q^ℓ and the target image O_t^ℓ, across the *two* levels combined, form the resultant salient feature set $\mathcal{F}_q^{img}$ and $\mathcal{F}_t^{img}$ which is passed on to SFM.

$$\mathcal{F}_q^{img} = \{O_q^1, O_q^2\} \quad \text{and} \quad \mathcal{F}_t^{img} = \{O_t^1, O_t^2\} \tag{9}$$

3.4. Semantic Feature Modification (SFM)

Here, we address the task of "how to change" by compositing the transformed image features with text features. Inputs to this are salient feature set $\mathcal{F}_q^{img}$ and text feature $\mathcal{F}_{sent}$. Briefly, in this step, we first perform a gating operation over feature levels and then subsequently modify the resultant with text. Formally, we take the feature set $\mathcal{F}_q^{img}$ and pass it through two independent linear projections. As features across levels encode different properties and consequently exhibit different output sizes, projecting them to a common space before modeling their interactions is helpful,

$$G_q^1, G_q^2 = \Omega_1([\mathcal{F}_q^{img}]_1), \Omega_2([\mathcal{F}_q^{img}]_2) \quad (10)$$

$$G_q^1, G_q^2 = BatchNorm(G_q^1), BatchNorm(G_q^2) \quad (11)$$

To selectively pass on features from the lower-level (G_q^1) to higher-level of hierarchy (G_q^2), we further use our gating operation, that uses an LSTM followed by a $BatchNorm$ to obtain aggregated feature vector f_{agg}:

$$\begin{aligned} H &= LSTM([G_q^1, G_q^2]) \\ f_{agg} &= \Omega(BatchNorm(H)) \end{aligned} \quad (12)$$

To obtain embedding f_{tgt} for the target feature set $\mathcal{F}_t^{img}$, we follow the same pipeline of projection up until gating across levels. (G_t^1, G_t^2 below are obtained using Eq. 10 and 11 for $\mathcal{F}_t^{img}$)

$$f_{tgt} = LSTM([G_t^1, G_t^2]) \quad (13)$$

Next, we take the aggregated feature vector (f_{agg}) and the global text vector f_{text} obtained by taking the linear projection $f_{text} = \Omega_g(\mathcal{F}_{sent})$. The final composed image-text representation which includes the modifications is then obtained by Residual Offsetting of f_{agg} with f_{text} followed by vector normalization as:

$$f_{com} = \delta \frac{f_{agg} + f_{text}}{\|f_{agg} + f_{text}\|_2} \quad (14)$$

where δ parameter denotes the learnable normalization scale and $\|.\|_2$ denotes the L_2 norm. We discuss the impact of this residual composition strategy in Section 4.4.

f_{com}, f_{tgt} and f_{text} are used as inputs to the loss functions detailed in the next section.

3.5. Loss Functions

The training dataset (I_{train}) is characterised by 3-tuples consisting of (I_q, D_s, I_t). Correspondingly, f_{com} represents the composed text-conditioned image embedding for (I_q, D_s), f_{text} represents the latent embedding for D_s and f_{tgt}^+ represents the latent embedding for I_t. Consider another image I_n sampled from I_{train}, s.t. $I_n \notin \{I_q \cup I_t\}$ where f_{tgt}^- represents its latent visual embedding which is generated using the same pipeline as for f_{tgt}. We next explain the different loss functions used to train *SAC*.

Triplet Loss is the primary training objective which seeks to constrain the *anchor* f_{com} to align with the *target* f_{tgt}^+ by simultaneously contrasting with the embedding for a *negative* image f_{tgt}^-. The loss function is defined as

$$\mathcal{L}_{triplet} = \log(1 + e^{\|f_{com} - f_{tgt}^+\|_2 \, - \, \|f_{com} - f_{tgt}^-\|_2}) \quad (15)$$

where $\|.\|_2$ operator denotes the L_2 norm.

To help learn discriminative representations, we employ a hard negative strategy that interleaves the random selection of I_n with an online distance-based sampling technique. This sampling weighs each $I_n \in I_{train}$ using the L_2-distance of the corresponding embedding (f_{tgt}^-) with f_{com} with smaller distances weighted higher.

Discriminator Loss helps improve the alignment of f_{com} with f_{tgt} by utilizing a discriminator that penalizes distributional divergence of linear projections of these embeddings.

$$\mathcal{L}_{disc} = -\mathbb{E}\big[log(\mathcal{D}(f_{tgt})\big] - \mathbb{E}\big[log(1 - \mathcal{D}(f_{com}))\big] \quad (16)$$

where $\mathcal{D}$ is the discriminator network which has three fully-connected layers and is trained end-to-end along with the entire model. Details about the architecture of the discriminator is provided in Appendix A. We also discuss the particular impact of using this discriminator loss in Section 4.4.

Consistency Loss constraints visual and linguistic projections of f_{com}, denoted by f_{gen}^{img} and f_{gen}^{text}, to align with latent embeddings f_{tgt} and f_{text} respectively. This objective by reconstruction regularizes and reinforces the balanced utilization of both text and image in composed embedding f_{com}.

$$\mathcal{L}_{cons} = \alpha_t \|f_{gen}^{text} - f_{text}\|_2 \; + \; \alpha_i \|f_{gen}^{img} - f_{tgt}\|_2 \quad (17)$$

where, $\|.\|_2$ is the L_2 norm.

In the above equation, we project the vector f_{com} using learnable transformations to obtain f_{gen}^{img} and f_{gen}^{text} as,

$$f_{gen}^{img} = \Omega_{img}^c(f_{com}) \quad f_{gen}^{text} = \Omega_{text}^c(f_{com}) \quad (18)$$

where Ω_{img}^c and Ω_{text}^c are learnable transformations and are trained end-to-end alongside the model. We discuss the particular impact of using this consistency loss in Section 4.4.

Total Loss used for training is computed as

$$\mathcal{L}_{total} = \lambda_1 \mathcal{L}_{triplet} + \lambda_2 \mathcal{L}_{disc} + \lambda_3 \mathcal{L}_{cons} \quad (19)$$

α_t, α_i, β, γ, λ_1 to λ_3 are learnable scalar hyperparameters.

4. Experiments

In this section, we formalize the datasets, baselines, implementation and evaluation details for our experiments. We use the same experimental and evaluation settings as used by the previous techniques to ensure consistency.

Dataset Method	FashionIQ								
	Dress		Toptee		Shirt		**Average**		**Average**
	R@10	R@50	R@10	R@50	R@10	R@50	R@10	R@50	
Image Only	2.92	10.10	4.53	11.63	5.34	14.62	4.26	12.12	8.19
Text Only	8.67	25.08	9.68	28.25	8.30	25.02	8.88	26.11	17.50
Concat	9.06	27.27	10.45	29.83	9.66	28.06	9.72	28.39	19.56
FiLM [28]	14.23	33.34	17.30	37.68	15.04	34.09	15.52	35.04	25.28
TIRG [37]	14.87	34.66	19.08	39.62	18.26	37.89	17.40	37.39	27.40
Relationship [31]	15.44	38.08	21.10	44.77	18.33	38.63	18.29	40.49	29.39
VAL [7]	21.47	43.83	26.71	51.81	21.03	42.75	23.07	46.13	34.60
VAL w/ GloVe [7]	22.53	44.00	27.53	51.68	22.38	44.15	24.14	46.61	35.38
CurlingNet [41] (FashionIQ-W 2019)	24.44	47.69	25.19	49.66	18.59	40.57	22.74	45.97	34.36
RTIC [33] (FashionIQ-W 2020)	**28.21**	51.41	28.00	55.58	21.30	44.80	25.83	50.59	38.22
ComposeAE w/ Random Emb. [3] (WACV 2021)	11.99	31.38	11.01	27.48	11.04	26.49	11.34	28.45	19.89
ComposeAE w/ BERT. [3] (WACV 2021)	14.03	35.1	15.8	39.26	13.88	34.59	19.89	36.31	25.44
SAC w/ *Random Emb.*	26.13	**52.10**	31.16	59.05	26.20	50.93	27.83	54.03	40.93
SAC w/ BERT	26.52	51.01	**32.70**	**61.23**	**28.02**	**51.86**	**29.08**	**54.70**	**41.89**

Table 1: Quantitative comparison on FashionIQ dataset. *SAC* outperforms existing methods using both randomly and BERT-pretrained initialized text embedding. Best numbers are highlighted in **bold**.

Dataset Method	Birds-to-Words		
	R@10	R@50	**Average**
Text Only	1.69	8.34	5.01
Image Only	15.45	32.14	23.80
Concat	12.05	34.27	23.16
TIRG [37]	15.8	38.65	27.22
VAL [7]	-	-	-
ComposeAE w/ Random Emb. [3]	10.94	29.35	20.14
ComposeAE w/ BERT. [3]	10.66	34.84	22.75
SAC w/ *Random Emb.*	**20.34**	44.94	**32.64**
SAC w/ BERT	19.56	**45.24**	32.40

Table 2: Quantitative comparison on Birds-to-Words dataset. *SAC* outperform existing methods using both randomly and BERT-pretrained initialized text embedding.

Dataset Method	Shoes			
	R@1	R@10	R@50	**Average**
Text Only	0.60	6.20	19.42	8.74
Image Only	6.07	25.6	47.87	26.51
Concat	5.70	20.32	39.97	22.00
FiLM [28]	10.19	38.39	68.30	38.96
TIRG [37]	12.60	45.45	69.39	42.48
Relationship [31]	12.31	45.10	71.45	42.95
VAL (2 level) [7]	14.98	47.25	-	-
VAL [7]	16.98	49.83	73.91	46.91
VAL w/ GloVe [7]	17.18	51.52	75.83	48.18
ComposeAE w/ Random Emb. [3]	3.46	20.84	52.58	25.62
ComposeAE w/ BERT [3]	4.37	19.36	47.58	23.77
SAC w/ *Random Emb.*	18.11	**52.41**	75.42	48.64
SAC w/ BERT	**18.5**	51.73	**77.28**	**49.17**

Table 3: Quantitative comparison on Shoes datasets. *SAC* outperform existing methods using both randomly and BERT-pretrained initialized text embedding.

4.1. Datasets

We conduct experiments on multiple benchmark datasets that are selected to maximize diversity in length of the natural language descriptions. Figure **??** (in Appendix) shows the average number of words in the support text vary from 5 to 31 across the different datasets. **Shoes** [6] contains 14,658 images of footwear tagged with relative captions for dialog-based interactive retrieval. The dataset is split into 10,000 training and 4,658 test images with *short* support text descriptions that have an average length of 5.32 words.

FashionIQ [16] contains 77,684 images of fashion products over 3 categories: Dress, Toptee, and Shirt, with 46,609 images in the training and 31,075 images in the validation set. The dataset is characterized by *medium* support text descriptions with an average length of 10.69 words per sample. Since the ground-truth is not publicly available, so we follow VAL and report performance on the validation set.

Birds-to-Words (B2W) [12] contains 15,931 images (12,770 training and 3,151 testing) tagged with descriptions of fine-grained differences between pairwise bird images. The natural language queries here are *long* with an average length of 31.38 words.

4.2. Experimental Setup

Baselines: We compare *SAC* with a wide range of baselines including early works and recent State of the Art models on this task. **Image Only** uses only image representation as composed embedding. **Text Only** uses only text representation as composed embedding. **Concat Only** concatenates (denoted by $+\!\!+$) and linear transforms the image and text representations (following details from [37]) to obtain the composed embedding. **Relationship [31]** takes the feature maps from final CNN layer alongwith the text feature from RNN and performs concatenation followed by MLP to learn cross-modal relationships. **FiLM** [28] is a Feature-wise Linear Modulation wherein the text information added to the CNN output to modulate each feature map by affine transformation. **TIRG** [37] concatenates visual and textual representations followed by learning a gating and a residual connection to obtain a composed embedding. **VAL** [7] composes the textual representation with the visual representations at multiple CNN layers using a composite transformer (more details are mentioned in Section 2).

VAL is the most recent state-of-the-art technique and the

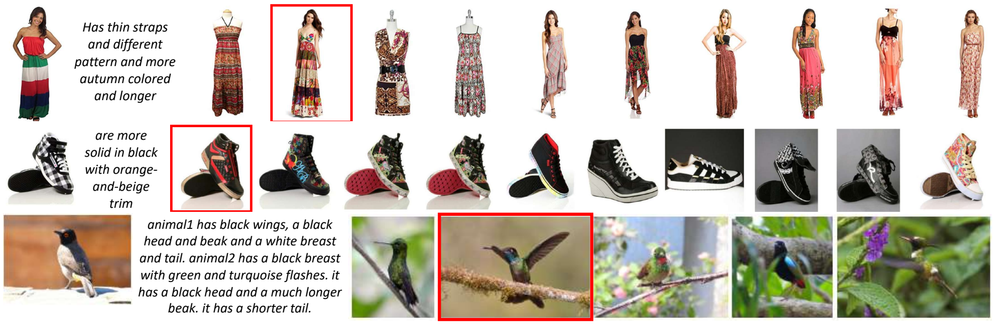
Has thin straps and different pattern and more autumn colored and longer
are more solid in black with orange-and-beige trim
animal1 has black wings, a black head and beak and a white breast and tail. animal2 has a black breast with green and turquoise flashes. it has a black head and a much longer beak. it has a shorter tail.

Figure 3: Qualitative results (one for each dataset) from our approach *SAC*. Images in the first column are the reference images followed with the query text. Retrieved results are ranked from left-to-right. Red boxes highlight the target image.

strongest baseline for our experimental study. For both VAL and TIRG, we refer to the author-provided code implementations with the recommended hyper-parameter settings.

Comparison with Workshop: For comparison with the FashionIQ 2020/2019 Workshop (FashionIQ-W), we compare with the workshop winners who follow the standard experimental settings i.e. who do not perform pre-training of feature network on attribute prediction tasks or any other external data/tasks, and do not use an ensemble of models. We compare *SAC* with RTIC [33] and CurlingNet [41]. However, RTIC uses the ResNet101 backbone while the widely used encoder for the task is ResNet50.

Implementation & Evaluation Details: We use Resnet-50 [19] pre-trained on ImageNet [10] as the backbone for image encoder and set $L = 2$ for our experiments. For our experiments with pretrained text embeddings, we use BERT [11] pretrained on QA task. Performance is evaluated using the Recall@K (R@K) $\{K = 1, 10, 50\}$ metric which computes the percentage of evaluation queries where the target image is found within the top-K retrieved images. We use a batch-size of 32 and Adam [22] (initial learning-rate of $1e^{-3}$) optimizer for image & text encoders and the SGD optimizer (initial learning-rate of $2e^{-4}$) for the discriminator. The learning rate was divided by 2 for the Adam optimizer and divided by 10 for the SGD optimizer when the loss plateaued on the validation set until it reached $1e^{-6}$. For the image encoder, we allow for the gradual fine-tuning by unfreezing it's weights only after first few epochs (10 in our case) of training. We use a temperature of $\mathcal{T} = 1, 8$. We choose the values λ_1 = 1, λ_2 = 0.6, λ_3 = 0.1 as the hyper-parameters of our loss functions. We take α_t = 1 and α_i = 0.1 in the consistency loss.

For the discriminator $\mathcal{D}$, we use a simple neural network with three fully-connected layers that reduce the feature vectors embedding size to a scalar value that is then passed through the loss function as described in Section 3.5. The architecture for the Discriminator is provided in Appendix.

4.3. Results

We present quantitative and qualitative comparison of *SAC* with our baselines on each of the three datasets. Due to limited space, additional results are included in the appendix which are cited where pertinent.

Quantitative Results: The quantitative results for all three datasets FashionIQ, Birds-2-Words and the Shoes dataset are summarized in Tables 1, 2, and 3 respectively. We also highlight the best number in **bold** in all the tables for convenience. We report the performance of *SAC* using both pre-trained BERT and random embeddings. Overall, we can see that *SAC* outperforms the strongest baseline on all three datasets by 3-4% on average on the R@10 and R@50 metrics. Moreover, our model also outperforms the baselines on the challenging Birds to Words dataset which has much longer and more complex sentences. [1]

Qualitative Results: To corroborate our quantitative observations, we also present a qualitative analysis for *SAC* and present the results for the same in Figure 3. We observe that the *SAC* is able to concurrently incorporate multiple semantic transformations in visual representations from text descriptions when retrieving images. We observe that *SAC* is able to – **(A)** retrieve new images while changing certain attributes conditioned on text feedback eg. color, material (from row 1, *SAC* captures the “autumn-colored” while preserving the “longer” property) **(B)** ingest multiple visual attributes and properties in the natural language text feedback (from row 2, “solid”, “black” and “orange-and-beige trim” all focus on different semantics of the image. *SAC* captures all of the semantics in the retrieved image) **(C)** can jointly comprehend global appearance and local details for image search (from row 2, *SAC* captures the overall “black” look across the retrieved results and attempts to find the appropriate local variations in the design) **(D)** aggregate multiple fine-grained semantic concepts within query sentence for image search (from row 3, *SAC* captures the fine-grained

[1] Results on B2W dataset for VAL were not available and experimenting using their code was prohibitive even with 16-GB GPUs

Composition	R@10	R@50
Concatenation	22.98	47.73
Residual Gating	24.00	46.72
Hadamard	22.74	46.35
Residual Offsetting	**27.83**	**54.02**

Table 4: Results on ablations for effect of composition

Method	# of Parameters
TIRG	16.96M
ComposeAE	19.07M
VAL	61.76M
Ours	21.74M

Table 5: Number of params for different methods

Loss Functions	Average	
	R@10	R@50
$\mathcal{L}_{triplet}$	23.47	48.48
$\mathcal{L}_{triplet} + \mathcal{L}_{cons}$	24.82	50.25
$\mathcal{L}_{triplet} + \mathcal{L}_{disc}$	22.84	48.87
$\mathcal{L}_{triplet} + \mathcal{L}_{disc} + \mathcal{L}_{cons}$	**27.83**	**54.02**

Table 6: Results from our ablation study showcasing the impact of individual losses on FashionIQ dataset.

Level-1	Level-2	R@10	R@50
✓		8.95	24.13
	✓	14.24	31.05
✓	✓	27.83	54.02

Table 7: Ablations for aggregating both levels

changes like "black breast", "green flashes" and "longer beak" in a single query and aggregates these concepts effectively). Due to limited space, we have included comparative qualitative analysis with VAL and additional qualitative results in Appendix C.

4.4. Ablation Studies

In this section, we conduct ablation studies to investigate the impact of different design choices in *SAC*. For all our ablations, we restrict our scope to the FashionIQ dataset for ease of exposition and analysis.

Importance of SFA and Attention Maps We provide the attention map from the last level ($\mathcal{A}^{\ell}_{\textbf{cross}}$) from *SAC* in Figure 4. From the figure, the network focuses on the region of the sleeve in the image since the text has "shorter sleeves", and the neck region as the text said "deeper neck". We also provide additional attention maps in Figure 5 in Appendix B. Besides, we also show the importance of the SFA module on the right side of Figure 4 and it can be seen that adding the SFA module improves the R@10 and R@50 metrics by around 3%. Since, for this problem at hand and our method, it wouldn't be logical to run analysis by removing SFM ('how to change'), which would make it equivalent to Image Only baseline as discussed in Section 4.2, hence, we omit the same in our study.

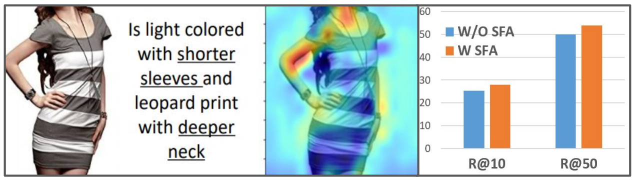

Figure 4: On the left, Attention maps for a pair of input image and text with specific keywords *underlined* corresponding to the attention heat-maps on the image. On the right, the effect of SFA is shown on the Fashion-IQ dataset

Effect of Residual Offsetting in SFM: Here, we study our idea to utilize text to only "modify" the image feature based on the text feature, rather than create an entirely. Correspondingly, we validate this design choice by contrasting with the following operators: *Concatenation*, *Hadamard Product*, *Residual Gating* (used in TIRG) and *Residual Offsetting* (defined in Eq. 14). The results are summarized in Table 4 which highlights that our operator significantly outperforms the alternate choices.

Importance of Aggregating both Levels: Here, we study the effect of aggregating both the levels in contrast to using one of them. Table 7 shows how taking coverage of concepts over both the levels results in better performance.

Effect of Discriminator and Consistency Loss: Discriminator loss is defined to provide a weaker supervision to further knit the two distributions (f_{com} and f_{tgt}) together while **Consistency Loss** is designed to regularize the learned composite multi-modal representations (see Section 3.5) Table 6 shows the efficacy of the two loss functions.

Comparison of Number of parameters: Proposing a simple yet efficient approach, we compare the number of parameters against existing SOTA (VAL [7], TIRG [37] and ComposeAE [3]) in Table 5. Table 1 shows how our model outperforms VAL significantly, while using just one-third number of parameters. Moreover, by adding just 14% parameters to ComposeAE, our model achieves a gain of 10% on R@10 metric.

5. Conclusion and Future Work

In this work, we focus on the task of text conditioned image retrieval and introduce *SAC*, which resolves the given task into 2 major steps, SFA (where to see) and SFM (how to change) which systematically streamlines the generation of text aware image features. We conduct extensive experiments on diverse benchmark datasets and consistently achieve state-of-the-art performance.

There are some cases when all the predictions are qualitatively coherent but this is not captured by metric since the specific target image is not a part of the retrieved set. Thus, exploring adaptive evaluation metrics is an interesting direction for future work.

References

[1] K. E. Ak, A. A. Kassim, J. H. Lim, and J. Y. Tham. Learning attribute representations with localization for flexible fashion search. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7708–7717, 2018. 1, 2

[2] S. Antol, A. Agrawal, J. Lu, M. Mitchell, D. Batra, C. L. Zitnick, and D. Parikh. Vqa: Visual question answering. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 2425–2433, 2015. 2

[3] Muhammad Umer Anwaar, Egor Labintcev, and Martin Kleinsteuber. Compositional learning of image-text query for image retrieval. *arXiv preprint arXiv:2006.11149*, 2020. 2, 6, 8

[4] A. Barman and S. K. Shah. A graph-based approach for making consensus-based decisions in image search and person re-identification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 1–1, 2019. 2

[5] Sean Bell and Kavita Bala. Learning visual similarity for product design with convolutional neural networks. *ACM Trans. Graph.*, 34(4), July 2015. 1

[6] Tamara L Berg, Alexander C Berg, and Jonathan Shih. Automatic attribute discovery and characterization from noisy web data. In *European Conference on Computer Vision*, pages 663–676. Springer, 2010. 6

[7] Yanbei Chen, S. Gong, and L Bazzani. Image search with text feedback by visiolinguistic attention learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3001–3011, 2020. 1, 2, 3, 6, 8

[8] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1724–1734, Doha, Qatar, Oct. 2014. Association for Computational Linguistics. 3

[9] Ayush Chopra, Abhishek Sinha, Hiresh Gupta, Mausoom Sarkar, Kumar Ayush, and Balaji Krishnamurthy. Powering robust fashion retrieval with information rich feature embeddings. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2019. 2

[10] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 7

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. 7

[12] Maxwell Forbes, Christine Kaeser-Chen, Piyush Sharma, and Serge Belongie. Neural naturalist: Generating fine-grained image comparisons. In *Conference on Empirical Methods in Natural Language Processing (EMNLP)*, Hong Kong, 2019. 6

[13] Jun Fu, Jing Liu, Haijie Tian, Yong Li, Yongjun Bao, Zhiwei Fang, and Hanqing Lu. Dual attention network for scene segmentation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3146–3154, 2019. 4

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

[15] Xiaoxiao Guo, Hui Wu, Yu Cheng, Steven Rennie, Gerald Tesauro, and Rogerio Feris. Dialog-based interactive image retrieval. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 678–688. Curran Associates, Inc., 2018. 1, 2

[16] Xiaoxiao Guo, Hui Wu, Yupeng Gao, Steven Rennie, and Rogerio Feris. Fashion iq: A new dataset towards retrieving images by natural language feedback. *arXiv preprint arXiv:1905.12794*, 2019. 6

[17] Alaa Halawani, Alexandra Teynor, Lokesh Setia, Gerd Brunner, and Hans Burkhardt. Fundamentals and applications of image retrieval: An overview. *Datenbank-Spektrum*, 18:14–23, 01 2006. 2

[18] Xintong Han, Zuxuan Wu, Phoenix X Huang, Xiao Zhang, Menglong Zhu, Yuan Li, Yang Zhao, and Larry S Davis. Automatic spatially-aware fashion concept discovery. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1463–1471, 2017. 1

[19] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2016. 7

[20] Surgan Jandial, Ayush Chopra, Kumar Ayush, Mayur Hemani, Balaji Krishnamurthy, and Abhijeet Halwai. Sievenet: A unified framework for robust image-based virtual try-on. In *The IEEE Winter Conference on Applications of Computer Vision*, pages 2182–2190, 2020. 1

[21] J. Johnson, R. Krishna, M. Stark, L. Li, D. A. Shamma, M. S. Bernstein, and L. Fei-Fei. Image retrieval using scene graphs. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3668–3678, 2015. 1

[22] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014. 7

[23] Yann LeCun, Y. Bengio, and Geoffrey Hinton. Deep learning. *Nature*, 521:436–44, 05 2015. 2

[24] L. Mai, H. Jin, Z. Lin, C. Fang, J. Brandt, and F. Liu. Spatial-semantic image search by visual feature synthesis. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1121–1130, 2017. 1, 2

[25] Julian McAuley, Christopher Targett, Qinfeng Shi, and Anton Van Den Hengel. Image-based recommendations on styles and substitutes. In *Proceedings of the 38th international ACM SIGIR conference on research and development in information retrieval*, pages 43–52, 2015. 1

[26] Hyeonwoo Noh, Andre Araujo, Jack Sim, Tobias Weyand, and Bohyung Han. Large-scale image retrieval with attentive deep local features. In *Proceedings of the IEEE international conference on computer vision*, pages 3456–3465, 2017. 1

[27] Badri Patro and Vinay P. Namboodiri. Differential attention for visual question answering. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018. 2

[28] Ethan Perez, Florian Strub, Harm De Vries, Vincent Dumoulin, and Aaron Courville. Film: Visual reasoning with a general conditioning layer. In *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018. 6

[29] F. Radenović, G. Tolias, and O. Chum. Fine-tuning cnn image retrieval with no human annotation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1655–1668, 2019. 4

[30] Sahana Ramnath, Amrita Saha, Soumen Chakrabarti, and Mitesh M. Khapra. Scene graph based image retrieval – a case study on the clevr dataset, 2019. 1

[31] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. A simple neural network module for relational reasoning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4967–4976. Curran Associates, Inc., 2017. 6

[32] Nikolaos Sarafianos, Xiang Xu, and Ioannis A. Kakadiaris. Adversarial representation learning for text-to-image matching. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019. 2

[33] Minchul Shin, Yoonjae Cho, and Seongwuk Hong. Fashion-iq 2020 challenge 2nd place team's solution. *arXiv e-prints*, pages arXiv–2007, 2020. 6, 7

[34] Anirudh Singhal, Ayush Chopra, Kumar Ayush, Utkarsh Patel Govind, and Balaji Krishnamurthy. Towards a unified framework for visual compatibility prediction. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020. 2

[35] A. Sinha, B. Akilesh, M. Sarkar, and B. Krishnamurthy. Attention based natural language grounding by navigating virtual environment. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 236–244, 2019. 2

[36] Stefanie Tellex, Thomas Kollar, Steven Dickerson, Matthew R Walter, Ashis Gopal Banerjee, Seth Teller, and Nicholas Roy. Understanding natural language commands for robotic navigation and mobile manipulation. In *Twenty-fifth AAAI conference on artificial intelligence*, 2011. 2

[37] 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6439–6448, 2019. 1, 2, 6, 8

[38] X. Xu, T. Wang, Y. Yang, L. Zuo, F. Shen, and H. T. Shen. Cross-modal attention with semantic consistence for image-text matching. *IEEE Transactions on Neural Networks and Learning Systems*, pages 1–14, 2020. 2

[39] Aron Yu and Kristen Grauman. Thinking outside the pool: Active training image creation for relative attributes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 708–718, 2019. 1

[40] Qian Yu, Feng Liu, Yi-Zhe Song, Tao Xiang, Timothy M Hospedales, and Chen-Change Loy. Sketch me that shoe. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 799–807, 2016. 2

[41] Youngjae Yu, Seunghwan Lee, Yuncheol Choi, and Gunhee Kim. Curlingnet: Compositional learning between images and text for fashion iq data. *arXiv e-prints*, pages arXiv–2003, 2020. 6, 7

[42] Matthew D. Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars, editors, *Computer Vision – ECCV 2014*, pages 818–833, Cham, 2014. Springer International Publishing. 2

[43] Han Zhang, Ian Goodfellow, Dimitris Metaxas, and Augustus Odena. Self-attention generative adversarial networks. In *International Conference on Machine Learning*, pages 7354–7363, 2019. 4

[44] B. Zhao, J. Feng, X. Wu, and S. Yan. Memory-augmented attribute manipulation networks for interactive fashion search. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6156–6164, 2017. 2

[45] Feng Zhou and Yuanqing Lin. Fine-grained image classification by exploring bipartite-graph labels. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1124–1133, 2016. 1