Attribute Manipulation Generative Adversarial Networks for Fashion Images

Kenan E. Ak[1,2] Joo Hwee Lim [2] Jo Yew Tham[3] Ashraf A. Kassim[1]
[1]National University of Singapore, Singapore
[2]Institute for Infocomm Research, A*STAR, Singapore
[3]ESP xMedia Pte. Ltd., Singapore
emir.ak@u.nus.edu, joohwee@i2r.a-star.edu.sg, thamjy@espxmedia.com, ashraf@nus.edu.sg

Abstract

Recent advances in Generative Adversarial Networks (GANs) have made it possible to conduct multi-domain image-to-image translation using a single generative network [7]. While recent methods such as Ganimation [24] and SaGAN [34] are able to conduct translations on attribute-relevant regions using attention, they do not perform well when the number of attributes increases as the training of attention masks mostly rely on classification losses. To address this and other limitations, we introduce Attribute Manipulation Generative Adversarial Networks (AMGAN) for fashion images. While AMGAN's generator network uses class activation maps (CAMs) to empower its attention mechanism, it also exploits perceptual losses by assigning reference (target) images based on attribute similarities. AMGAN incorporates an additional discriminator network that focuses on attribute-relevant regions to detect unrealistic translations. Additionally, AMGAN can be controlled to perform attribute manipulations on specific regions such as the sleeve or torso regions. Experiments show that AMGAN outperforms state-of-the-art methods using traditional evaluation metrics as well as an alternative one that is based on image retrieval.

1. Introduction

Attribute manipulation involves making translations/adjustments to images based on the target attributes. For fashion products, attributes of interest relate to visual qualities such as *sleeve length, color and pattern* while attribute values correspond to certain labels such as *long sleeve, red color and plain pattern*. Being able to manipulate attributes of images is especially useful in a variety of situations including a user not satisfied with some attributes. Recently, this task was studied from an image retrieval perspective which involved retrieving the target images in a dataset after conducting attribute manipulation [37, 2, 1, 4]. However, the image retrieval approach is limited by the dataset size and the increasing number of attributes.

Figure 1: Multi-domain image-to-image translation examples of the proposed AMGAN using images from the Deepfashion [20] and Shopping100k [3] datasets.

Since the introduction of Generative Adversarial Networks (GANs) [10], the task of image generation has received significant attention. Along with many computer vision tasks, GANs can be applied for the image-to-image translation problem [14, 39]. The StarGAN [7] architecture has shown to be able to perform multi-domain image-to-image translations with a single generative network. More recently, several approaches that incorporate an attention mechanism on the generative network have emerged such as Ganimation [24] and SaGAN [34]. Having an attention mechanism is especially useful when attribute manipulation needs to be performed on attribute-relevant regions while others remain the same. However, as the number of attributes increases, these attention-based methods become unstable as the attended regions are mostly learned through classification losses. Additionally, the discriminator network can also benefit from an attention mechanism and force the generative network to perform more realistic attribute manipulations.

In this paper, we propose Attribute Manipulation Generative Adversarial Networks (AMGAN) which focuses on

the multi-domain image-to-image translation problem for fashion images enabling users to conduct attribute manipulations. While the current image-to-image translation networks are mostly meant for face images, AMGAN achieves this for the less rigid objects such as fashion images. Figure 1 illustrates some examples of attribute manipulation on the Deepfashion [20] and Shopping100k [3] datasets. As shown, AMGAN has the ability to translate input images into new ones based on changes to target attributes while preserving the other attributes.

The proposed AMGAN incorporates an attention mechanism for attribute manipulation without leveraging on any information about the attribute location. In attribute manipulation, the objective is to locate those regions with the attribute of interest so that they can be translated into new ones. Therefore, it is crucial for the generative network to correctly localize regions based on the attributes to be manipulated. Class activation maps (CAMs) extracted from a Convolutional Neural Network (CNN) can be used to correctly localize the discriminative region of an attribute. By using CAMs as an attention loss, AMGAN's generator network is able to generate attention masks correctly which consequently improves its attribute manipulation ability. While different previous works [7, 24, 34] use a single discriminator for the whole image, AMGAN uses an additional discriminator network that focuses on attribute-relevant regions to detect unrealistic attribute manipulations to improve image translation performance.

For unpaired image-to-image translation, there is no reference (target) image according to the input image and attribute manipulation, making it infeasible to use perceptual losses [8, 9, 16]. We fix this issue by assigning a reference image based on attribute similarities. Consequently, AMGAN benefits from a perceptual loss function which is based on features from the same CNN that extracts CAMs. With the perceptual loss, AMGAN is able to generate more realistic images while the ability to match the attribute manipulation is boosted. In addition to conventional image-to-image translation, AMGAN can be adapted to conduct attribute manipulations on specific regions by intervening with attention masks. For example, AMGAN can be adjusted to perform "red color" attribute manipulation on the sleeve region by replacing its attention mask with a mask for "sleeveless" attribute manipulation. This ability is useful to automate the region-specific attribute manipulation.

The key contributions of AMGAN are:

- Empowering attention mechanism of the generative network with CAMs extracted from the same CNN that is used to enable perceptual losses based on attribute similarities.
- Incorporating an additional discriminator that focuses on attribute-relevant regions.
- Enabling attribute manipulations on specific regions.
- Detailed experiments on two fashion datasets are presented to show the superior performance of AMGAN over state-of-the-art methods. We also introduce a new method based on image retrieval to test the success of attribute manipulation.

2. Related Work

Generative Adversarial Networks (GANs). GANs introduced by Goodfellow *et al*. [10] have demonstrated remarkable success in many computer vision problems including image generation [25, 28], image-to-image translation [14, 7], image inpainting [18, 23]. GANs consist of generator and discriminator networks where they compete with each other in a minimax game. While the generator tries to produce realistic samples, the discriminator attempts to distinguish the fake samples for the real ones. Networks are trained jointly with an adversarial loss.

Conditional GANs (cGANs). GANs can be modified to generate images based on several conditions. The conditional generation of samples can be from the class information [21, 22], text descriptions [27, 35, 32], etc. Using encoder-decoder architecture, the conditions can be applied to conduct domain changes on images such as image inpainting [23], image editing [6]. AMGAN uses conditions to signal attribute manipulations and performs multi-domain image-to-image translations.

Image-to-Image Translation. The aim of this task is to apply certain changes to the input image. Based on cGANs, pix2pix [14] used paired data to train the generative network based on pixel similarity and adversarial loss. CycleGAN [39] removed the obligation of the paired data and introduced a novel cycle consistency loss function for image-to-image translation. The main drawback of CycleGAN [39] is that it can only operate between two domains at a time. This issue is addressed by the StarGAN architecture [7] which includes auxiliary classification losses and trains a single generator network for multi-domain translations. Following StarGAN [7], several architectures that involve attention mechanism emerged [24, 34, 36]. However, these methods mostly rely on classification losses in order to produce an attention mask and only benefit from an attention mechanism on the generator network.

GANs in Fashion. Generative networks have also been widely applied on various fashion-related tasks such as virtual try-on [15, 12, 30, 26] and fashion design/generation [19, 31, 29]. Similar to our task, FashionGan [40] is introduced to conduct text-based image manipulation *e.g*., short sleeve to long sleeve while preserving the person wearing the clothing. In contrast, AMGAN focuses on attributes which are more accessible in many datasets and proposes

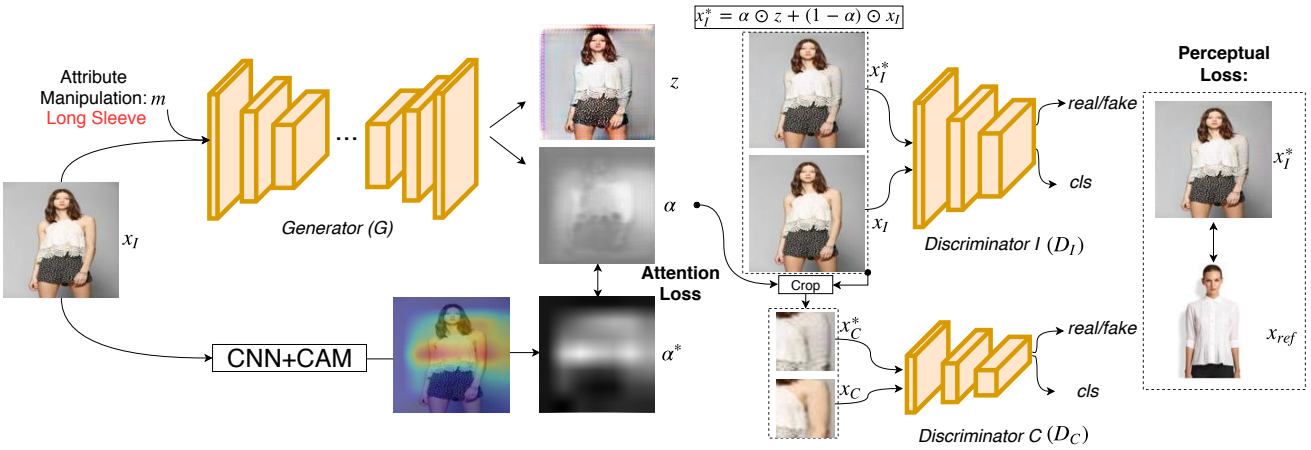


Figure 2: Overview of the proposed AMGAN architecture. Given input image x_I and attribute manipulation m, the generator G produces two outputs: generated image z and attention mask α where the final output x_I^* is generated through a blending operation. Class activation mapping (CAM) technique is used to assist the attention mask generation. For illustration purposes, the nested form of x_I and α^* is shown. Both discriminators D_I, D_C serve as real/fake and attribute classifiers. While the inputs of D_I are from the whole images, D_C's inputs are the attribute localized images estimated from α. Based on attribute similarities, a reference image x_{ref} is used for the perceptual loss.

several innovations. Additionally, AMGAN does not require any segmentation/annotation maps.

3. AMGAN

In this section, we describe AMGAN which is able to carry out the multi-domain image-to-image translation (attribute manipulation). Next, we show how AMGAN can be adjusted to perform region-specific attribute manipulations.

Problem definition. The proposed AMGAN architecture consists of a generator G and two discriminator networks D_I, D_C as shown in Figure 2. The aim of G which is based on an encoder-decoder structure is to translate an input image x_I by applying an attribute manipulation m to an output image x_I^*, $G(x_I, m) \rightarrow x_I^*$. All possible attribute manipulation operations can be encoded into $m = \{m_1, ...m_N, r\}$ where N is the number of attribute values (*e.g.*, long sleeve, red color, etc.) and r indicates the attribute that is being manipulated (*e.g.*, sleeve, color, etc.) so that the generator focuses on a specific attribute for each training iteration. Both r and m are represented by one-hot encodings.

3.1. Network Construction

The input image x_I is fed together with the attribute manipulation m into G which has two outputs: generated image z and attention mask α. The attention mask is combined with input and output images as in [12, 24, 33] where only specific regions are subject to attribute manipulations while the other regions are not changed. The final output x_I^* is obtained as follows:

$$x_I^* = \alpha \odot z + (1 - \alpha) \odot x_I \tag{1}$$

By feeding x_I into the CNN that is pre-trained with attributes of interest, a guidance mask α^* is calculated using class activation mapping (CAM) method [38] in order to assist G on which regions to focus with the attention loss. Additionally, based on attribute similarities, a reference image x_{ref} which demonstrates the expected attributes after attribute manipulation is assigned for the perceptual loss. Discriminator networks are used to distinguish the real samples from the fake ones and provide the classification loss. While D_I focuses on the whole image, the aim of D_C is to focus on attribute manipulated regions. Inputs of D_C denoted as x_C^*, x_C are estimated from α. First, pixel values of α that are above 50% of its maximum value are segmented followed by estimating a bounding box that covers the largest connected region. Using bounding boxes, x_C^*, x_C are cropped from x_I^*, x_I as shown in Figure 2.

3.2. Discriminators

Both discriminators D_I, D_C are based on a deep convolutional network and have two outputs for adversarial and classification losses.

Adversarial Loss. We denote the ouput of image x_d as $D_{d_{src}}(x_d)$ where $d \in \{I, C\}$ indicates the discriminator on whole image x_I or cropped image x_C. The purpose of discriminators is to maximize $D_{d_{src}}(x_d)$ and minimize $D_{d_{src}}(x_d^*)$. Accordingly, the overall adversarial loss for D is defined as:

$$L_{adv}^D = \sum_{d \in \{I,C\}} \Big(-\mathbb{E}_{x_d}[D_{d_{src}}(x_d)] + \mathbb{E}_{x_d^*}[D_{d_{src}}(x_d^*)] + \lambda_{gp} \mathbb{E}_{\tilde{x}_d}[(|| \bigtriangledown_{\tilde{x}_d} D_{d_{src}}(\tilde{x}_d)||_2 - 1)^2] \Big) \tag{2}$$

The final term in Eq. (2) above is the Wasserstein GAN

objective [5, 11] with gradient penalty λ_{gp} where $\tilde{x}_d$ is sampled uniformly along a straight line between real and generated images.

Classification Loss. In addition to recognizing real/fake samples with an adversarial loss, it is also crucial that discriminators can classify the attributes of real/fake images. Therefore, D_d has another output denoted as $D_{d_{cls}}(m'|x_d)$ where m' corresponds to the original attribute value before the attribute manipulation. Using the cross-entropy loss function, the overall classification loss for D is defined as:

$$L_{cls}^{D} = \sum_{d\in\{I,C\}} E_{x_d}[-logD_{d_{cls}}(m'|x_d)] \tag{3}$$

Combining both losses, the objective function to optimize both discriminators can be written as:

$$L_D = L_{adv}^{D} + \lambda_{cls} L_{cls}^{D} \tag{4}$$

3.3. Generator

AMGAN's generator G aims to generate a new image according to the attribute manipulation and consists of the following loss functions:

Adversarial Loss. As it is crucial for G to generate realistic samples, the following adversarial loss is used:

$$L_{adv}^{G} = \sum_{d\in\{I,C\}} -E_{x_d^*}[D_{d_{src}}(x_d^*)] \tag{5}$$

Classification Loss. In order to generate images with respect to m, the generated images are fed into the discriminators to estimate $D_{d_{cls}}(m|x_d^*)$ and the classification loss for G is defined as:

$$L_{cls}^{G} = \sum_{d\in\{I,C\}} E_{x_d^*}[-logD_{d_{cls}}(m|x_d^*)] \tag{6}$$

when $d = C$, attribute localized images are fed into D_C which forces G to generate more realistic samples with the correct attribute value on the attended region.

Cycle Consistency Loss. We use cycle consistency loss [39] to make sure that the contents of inputs are preserved while "irrelevant regions" remain unchanged. When attribute manipulations m and m^* are back-to-back performed on x_I, the generated image is expected to be the same as x_I. Therefore, cycle consistency loss is defined as:

$$L_{cyc}^{G} = E_{x_I}[||x_I - G(G(x_I, m), m^*)||_1] \tag{7}$$

Attention Loss. It is possible to have plausible attention masks with the loss functions defined above where the classification loss would drive attention masks toward attribute relevant regions. However, as the number of attributes increases or when images exhibit challenging pose variations, attention masks may become unstable, hence affect attribute manipulation results. As it is not possible to have a ground truth mask for every attribute, we propose to use class activation mapping (CAM) [38] technique to guide the generator network on the attribute location. By using CAM, the attention loss is included in G as follows:

Class Activation Mapping (CAM). First, the input image x_I is passed to the CNN which produces convolutional features f_k. Class activation map of x_I for the original attribute m' at spatial location (i, j) is estimated as follows:

$$M_{m'}(x_I, i, j) = \sum_k w_k^{m'} f_k(x_I, i, j) \tag{8}$$

where $w_k^{m'}$ is the weight variable of attribute m' associated with $k'th$ feature map. The values of $M_{m'}(x_I, i, j)$ are then normalized to the range of $(0, 1)$ which correspond to a guidance mask denoted as α^*. We adopt the L1 norm and define the attention loss between the attention mask of G and of CNN:

$$L_a^G = ||\alpha - \alpha^*||_1 \tag{9}$$

After α^* is calculated, one can choose to use it without having an extra attention mask output on the generator. This is troublesome because of two reasons: (1) while CAMs are used to find where classification scores come from, the task in AMGAN is to perform attribute manipulation where it can benefit from the combination of all losses; (2) CAMs sometimes may correspond to small regions which is problematic when the attribute to be manipulated involves the entire clothing item (*e.g.*, color). We use the attention loss to contribute to AMGAN's localization ability while not directly "mimic" CAMs.

Perceptual Loss. We use the perceptual loss that is based on differences between feature representations of a CNN which is defined as follows:

$$L_p^G = \sum_{j=1}^{n} ||CNN_j(x_{ref}) - CNN_j(x_I^*)||_1 \tag{10}$$

where j represent features extracted from the j'th layer of the CNN. In the unpaired image-to-image translation task, using this loss function may be confusing as it is unclear how to choose the reference image x_{ref} without having a paired match. In AMGAN, we propose to choose the reference image as the one which corresponds to attributes after the attribute manipulation. Therefore, the rule of picking x_{ref} is that all attributes should match with x_I^* as shown in the example provided in Figure 2. Even though wearers and their poses are different, x_I^* and x_{ref} are close to each other in the feature space. While improving the quality of the generated image z, the perceptual loss can also contribute to the attention mechanism.

Finally, the objective function to optimize G can be jointly written as:

$$L_G = L_{adv}^{G} + \lambda_{cls} L_{cls}^{G} + \lambda_{cyc} L_{cyc}^{G} + \lambda_a L_a^G + \lambda_p L_p^G \tag{11}$$

Hyper-Parameters. λ_{cls}, λ_{cyc}, λ_a, λ_p are hyper-parameters that control the importance of different terms. In our experiments, we use the following setup; $\lambda_{cls} = 1, \lambda_{cyc} = 10, \lambda_a = 10, \lambda_p = 20$.

4. Region-specific Attribute Manipulation

AMGAN's ability to perform attribute manipulation towards specific regions is limited as it does not use any segmentation ground truths. This can be overcome by allowing the user to manually edit AMGAN's attention mask which is time intensive. In order to automate this process, we propose a method that enables attribute manipulations on specific regions such as the torso, sleeve, etc.

First, attribute manipulation is performed using the generator network. If say, the user wants to manipulate only the sleeve or torso regions, an intervention must be made on the attention masks as shown in Figure 3. In order to generate region-specific attention masks, "sleeveless" attribute manipulation applied which would highlight the sleeve regions denoted as α_1. Before directly applying α_1, we use a threshold function to get rid of the pixel values that are smaller than 0.9 to clear out the noisy values. The attention mask α_1^* can now be applied to perform "orange color" attribute manipulation on the sleeve or torso region as:

$$x_a^* = \alpha_1^* \odot z + (1 - \alpha_1^*) \odot x \tag{12}$$

$$x_b^* = (1 - \alpha_1^*) \odot z + \alpha_1^* \odot x \tag{13}$$

This method can also be used to show attention masks are correlated with attribute manipulations. More variations of this method are investigated in experiments.

5. Implementation Details

Network Architecture: For the generator network in AMGAN, we use a structure similar to [39] and add an additional convolutional layer with a sigmoid activation function which outputs a single channel attention mask. The input of the generator is a tensor with "3+N+M" dimensions where N is the number of attribute values and M corresponds to the number of attributes. We use the masking vector from [7] to perform the alternating training strategy between the attributes. The PatchGAN architecture [14] is used for both discriminator networks. For the second discriminator network D_C, the size of input images are halved and two less convolutional layers are used.

For the CNN architecture, we use ResNet-50 [13] to extract CAMs and features. For each dataset, transfer learning (fixed for AMGAN) is performed for attribute prediction. The same network is employed for the feature extraction using $conv_5$ and avg_pool layers.

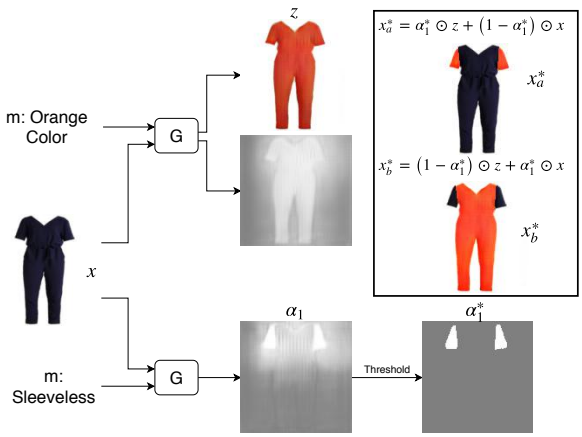


Figure 3: Region-specific attribute manipulation is possible by using attention masks from the sleeve attribute. Note that we omitted the generated image output from the sleeveless attribute manipulation.

Training: We train AMGAN from scratch using Adam optimizer [17] with $\beta = 0.5$, $\beta = 0.999$, set the learning rate to 0.0001 and use the mini-batch size of 16. For each generator update, the discriminator is updated 5 times. For the DeepFashion and Shopping100k datasets, we train AMGAN for 80k and 50k iterations for each attribute which takes about 1,5 and 2 days respectively with a GeForce GTX TITAN X GPU. After the first half of the training is finished, the learning rate is linearly decreased to zero.

6. Experiments

In this section, AMGAN is compared with several recent methods using quantitative and qualitative experiments. We also perform ablation experiments to investigate the effect of each novel component.

6.1. Competing Methods

Following state-of-the-art architectures that have been shown to successfully conduct multi-domain image-to-image translations are chosen as competing methods:
StarGAN [7] uses a single generator network which translates an input image to the target attribute and is able to perform the multi-domain image-to-image translations.
Ganimation [24] has a similar architecture to StarGAN but contains an attention mechanism on the generator. We replace the regression loss with the classification loss.
SaGAN [34] also incorporates an attention mechanism but the main difference with Ganimation is that it consists of two generative networks to generate images and attention masks. For multi-domain translations, we add attribute manipulation as a condition in order to have a single model for all attributes.

6.2. Datasets

Two fashion datasets which are rich in terms of the number of attributes are used for the experiments:

DeepFashion-Synthesis [40] dataset includes 78,979 images extracted from the DeepFashion dataset [20] and consists of upper clothing images. This subset is a much more clean version of the DeepFashion dataset and we choose to use the following attributes: color (17), sleeve (4) which corresponds to 21 attribute values.

Shopping100k dataset [3] includes 101,021 clothing images and we choose to use the following 6 attributes: collar (17), color (19), fastening (9), pattern (16), sleeve length (9) corresponding to 70 attribute values. We only use avg_pool layer to extract features with $\lambda_p = 10$ for this dataset.

All images are resized to 128x128 and 2,000 images are randomly sampled for the test set, the rest is used in the training. We choose to use mostly sensible attributes for attribute manipulations from both datasets. While choosing reference images, we additionally include category and gender attributes to have more correct matches.

6.3. Evaluation Metrics

Classification Accuracy. In order to test if an attribute manipulation is successfully applied, we check the classification accuracy for the attribute that is being manipulated. We choose to train a ResNet-50 architecture [13] for attribute classification with cross-entropy loss. All competing methods are tested with the same architecture and the higher accuracy rates indicate that attribute manipulation is more successful according to the network.

Top-k Retrieval Accuracy. We propose an image retrieval based method to evaluate the success of the generated images. Top-k retrieval accuracy considers whether the search algorithm finds a correct image in Top-k results or not. If the retrieved image consists of the attributes demanded by the input and attribute manipulation it will be a hit "1", otherwise a miss "0". This metric can be applied for attribute manipulation, as we directly generate the desired image. More specifically, we use ResNet-50 network and extract features from avg_pool layer for both generated images (Query) and real images (Retrieval Gallery) and conduct comparisons between Query and Retrieval Gallery.

User Study. In order to assess the generated images from each competing method, we perform a user study consisting of 20 participants. Before the study, each participant is instructed on each attribute value. Given an input image and the attribute manipulation, participants were asked to pick the best-generated image based on perceptual realism, quality of attribute manipulation, and preservation of an image's original identity from the four competing methods.

	DeepFashion			Shopping100k					
	Color	Sleeve	**Avg.**	Collar	Color	Fasten	Pattern	Sleeve	**Avg.**
StarGAN	69.19	61.76	65.47	26.37	46.76	23.36	22.82	59.58	35.76
Ganimation	67.77	67.17	67.47	21.73	17.31	18.47	11.75	45.87	23.02
SaGAN	68.59	63.16	65.87	18.21	30.03	17.56	10.51	47.07	24.68
AMGAN	**77.01**	**81.94**	**79.48**	**45.84**	**64.52**	**29.07**	**43.59**	**64.42**	**49.49**
AMGAN w/o D_C	74.23	73.86	74.05	40.51	59.66	21.60	35.53	58.46	43.15
AMGAN w/o L_p^G, D_C	70.19	69.49	69.84	30.70	50.84	21.61	21.95	52.25	35.47
AMGAN w/o L_a^G, L_p^G, D_C	68.28	66.40	67.34	24.19	25.98	21.53	11.91	43.91	25.50
AMGAN, CAMs as α	59.15	66.60	62.88	24.47	35.65	19.75	21.82	28.97	26.13

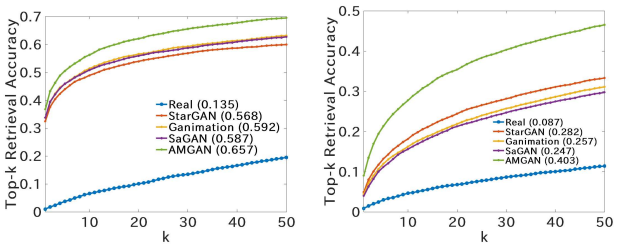
Table 1: Classification accuracy of manipulated attributes for competing methods and ablation experiments.

6.4. Quantitative Experiments

Classification accuracy results with respect to attribute manipulations are reported in Table 1 where AMGAN achieves the best performance with 79.48%, 49.49% on average for DeepFashion and Shopping100k datasets respectively. While all three competing perform close to each other in DeepFashion dataset, StarGAN has a much better performance on Shopping100k dataset and the reason for this is mostly due to the higher number of attributes (21 vs 70). This points out the scaling issue of attention-based methods with the increasing number of attributes and having an attention mechanism does not necessarily bring extra success for this evaluation metric. On the other hand, AMGAN is much more stable due to its novel components. A thorough investigation of AMGAN's components is made on the ablation experiments.

According to Figure 4, AMGAN performs better than the competing models with 0.657 and 0.403 average Top-30 retrieval accuracy for DeepFashion and Shopping100k datasets respectively. We also report the Top-30 retrieval accuracy of each attribute in Table 2 for a more detailed investigation. These experiments show that AMGAN is not only superior for the attribute manipulation but for keeping the untouched attributes the same. We also report results of real images which means that features are extracted from the input images without any attribute manipulation. The fact that real images achieve the worst result by a large margin proves that the ability to perform attribute manipulation is very important for this metric. Both Ganimation and SaGAN perform worse than StarGAN for Shopping100k dataset which was the case in Table 1 showing the correlation between two evaluation metrics. We believe that Top-k retrieval accuracy is a good metric to observe the balance of "keeping the untouched attributes" and "enabling attribute manipulations" for the generated images. These experiments also suggest that AMGAN can be a good model for image retrieval as well which is worth investigating for future studies.

In Table 3, we show the results of the user study which



(a) DeepFashion (b) Shopping100k

Figure 4: Average Top-K retrieval accuracy of attribute manipulations. The number in the parentheses corresponds to the Top-30 retrieval accuracy.

	DeepFashion			Shopping100k					
	Color	Sleeve	**Avg.**	Collar	Color	Fasten	Pattern	Sleeve	**Avg.**
Real	0.209	0.061	0.135	0.061	0.092	0.138	0.078	0.064	0.087
StarGAN	0.724	0.413	0.568	0.263	0.453	0.153	0.205	0.333	0.282
Ganimation	0.720	0.465	0.592	0.303	0.218	0.220	0.143	0.398	0.257
SaGAN	0.725	0.449	0.587	0.260	0.300	0.188	0.126	0.358	0.247
AMGAN	**0.766**	**0.550**	**0.657**	**0.439**	**0.547**	**0.276**	**0.305**	**0.447**	**0.403**

Table 2: Top-30 retrieval accuracy of each attribute manipulation using the generated images.

is based on preferences. AMGAN again achieves the best performance for all attributes where other competing methods perform similarly to each other in both datasets. For the Shopping100k dataset, StarGAN performs not as good as shown in Table 1 especially with the sleeve attribute where it had a good performance. The user study proves that having high classification accuracy does not mean that a model is performing visually good translations. An interesting finding in Shopping100k dataset is that StarGAN has better ability to perform attribute manipulations which involves the whole image (color, pattern) compared to those which correspond to a specific region (collar, fasten, sleeve). The user study suggests that AMGAN's attention mechanism is more stable than Ganimation and SaGAN while performing more realistic translations. For the fastening attribute, AMGAN has slightly lower score indicating this attribute manipulation is more difficult than the others.

Ablation Experiments. We analyzed the performance of AMGAN without the following: D_C network, perceptual loss, and attention loss as well as using CAMs directly as α by using the classification accuracy results presented on the lower side of Table 1.

AMGAN w/o D_C: Removing D_C from the AMGAN architecture results in 5.43% and 6.34% average accuracy drop for the Deepfashion and Shopping100k datasets respectively. Therefore, the effect of D_C cannot be ignored as it directly forces the generator to produce more realistic images with correct attributes on attended regions.

AMGAN w/o L_p^G, D_C: Disabling the perceptual loss (Eq. 10) in addition to removing D_C reduces the average accuracy by 4.21% and 7.68% for the Deepfashion and Shopping100k datasets compared to "AMGAN w/o D_C". This shows that enabling the perceptual loss to guide AMGAN on the expected output image has a positive effect on the ability to perform attribute manipulations.

	DeepFashion			Shopping100k					
	Color	Sleeve	**Avg.**	Collar	Color	Fasten	Pattern	Sleeve	**Avg.**
StarGAN	12.7	10.9	11.8	10.3	25.0	12.2	13.0	6.3	13.36
Ganimation	12.7	12.6	12.65	20.5	1.2	17.1	11.6	12.6	12.6
SaGAN	18.6	8.4	13.5	5.1	19.3	24.4	4.4	14.6	13.56
AMGAN	**56.0**	**68.1**	**62.05**	**64.1**	**54.5**	**46.3**	**71.0**	**66.5**	**60.48**

Table 3: User study results. Each column sums to 100.

AMGAN w/o L_a^G, L_p^G, D_C: Additionally, we disable the attention loss (Eq. 9) compared to "AMGAN w/o L_p^G, D_C" which reduces the average accuracy by 2.50% and 9.97%. In this version, the generator network has problems on correctly localizing towards correct regions by not having the extra assistance from CAMs. The difference is evident especially for the Shopping100k dataset where the number of attributes is much higher. As expected, AMGAN without its novel components has a similar performance to Ganimation [24] and SaGAN [34].

AMGAN, CAMs as α: For this model, we exclude the attention mask output of AMGAN and directly use CAMs (α^*) to compute Eq. 1. In order to compare "using CAMs as an attention loss" vs "directly using CAMs", D_C network and the perceptual loss are not included in the training. As shown in Table 1, "AMGAN, CAMs as α" performs 6.97% and 9.34% worse than "AMGAN w/o L_p^G, D_C" for the Deepfashion and Shopping100k datasets. This finding confirms our intuitive of using CAMs as a tool in the training rather than directly utilizing them. The aim of attention loss is to contribute to AMGAN's localization ability not directly replicate CAMs.

6.5. Qualitative Evaluation

From Figure 5 which presents several attribute manipulation examples, it is evident that AMGAN performs decent translations in terms of performing attribute manipulation and keeping contents of the original image. For the DeepFashion dataset, it is evident that the competing methods which apply color attribute manipulation have trouble focusing on the correct regions. With the "extra help" from a deep neural network, AMGAN is able to generate more accurate attention masks which result in having more realistic translations. Looking at the sleeve attribute, all methods seem to be applying translations on the correct regions; however, AMGAN is able to perform more realistically as the generated sleeve regions are more correlated with the input image in terms of color and pattern similarity.

For the Shopping100k dataset, several examples are provided on the right side of Figure 5. Attribute manipulation results by AMGAN are again more consistent and accu-

Figure 5: Attribute manipulation examples on the DeepFashion and Shopping100k datasets. The first two columns show input image and attribute manipulations while the other columns are images generated from each competing method. As can be seen, AMGAN consistently generates better images. More examples can be found in the supplementary material along with attention mask outputs of G.

Figure 6: Region-specific attribute manipulation examples for the Shopping100k and DeepFashion datasets. Each attribute manipulation provides the desired attribute of the sleeve region.

rate. This can easily be seen from the long sleeve attribute manipulation in the third row where the competing methods only provide a silhouette of the desired attribute. AMGAN's ability to perform more accurate realistic translations is mostly due to using an extra discriminator to attend attribute specific regions and perceptual loss which guides the generator on the expected outputs.

Region-specific Attribute Manipulation: For this case, a set of examples with both datasets are provided in Figure 6. For the DeepFashion dataset, attention masks which are obtained from "sleeveless" attribute value are used to perform regions-specific attribute manipulations. Following that, we perform "red color" and "orange color" attribute manipulations on sleeve masks using Eq. 12. Compared to the Shopping100k dataset, it is more difficult to localize towards clothing products due to the occlusion from the wearer. In addition, since we are using a hard threshold method, it may sometimes be problematic to find an optimum value. Regardless, attribute manipulations are applied successfully in the first two rows. For the last two rows, we get decent results given the fact that we only use attributes for this process.

For the Shopping100k dataset, it can be seen that attention masks which are obtained from "sleeveless" attribute value can highlight the sleeve regions successfully. Following that, we perform "blue color" and "color gradient pattern" attribute manipulations. Looking at the final output, the idea of mask intervention is successfully applied for the region-specific attribute manipulation task. These experiments also show the success of attention masks produced by the generator network.

7. Conclusion

Attribute Manipulation Generative Adversarial Networks (AMGAN) for multi-domain image-to-image translation introduced in this paper has a major performance advantage over competing methods due to its improved attention mechanism and attribute manipulation. The performance boost is made possible by the guidance of CAMs and perceptual loss as well as having an additional discriminator network. By taking advantage of attention masks, AMGAN is able to conduct attribute manipulations towards specific regions. Through experiments conducted on the DeepFashion and Shopping100k datasets, we show that AMGAN is able to perform better than the state-of-the-art image-to-image translation methods based on traditional metrics as well as a new one that is based on image retrieval. An interesting challenge for future work would be to extend AMGAN towards different domains or use it as a tool for image retrieval after attribute manipulation task.

References

[1] Kenan E. Ak, Ashraf A. Kassim, Joo Hwee Lim, and Jo Yew Tham. Fashionsearchnet: Fashion search with attribute manipulation. In *The European Conference on Computer Vision (ECCV) Workshops*, September 2018. 1

[2] Kenan E. Ak, Ashraf A. Kassim, Joo Hwee Lim, and Jo Yew Tham. Learning attribute representations with localization for flexible fashion search. In *CVPR*, June 2018. 1

[3] Kenan E. Ak, Joo Hwee Lim, Jo Yew Tham, and Ashraf A. Kassim. Efficient multi-attribute similarity learning towards attribute-based fashion search. In *WACV*, pages 1671–1679. IEEE, 2018. 1, 2, 6

[4] Kenan E Ak, Joo Hwee Lim, Jo Yew Tham, and Ashraf A Kassim. Which shirt for my first date? towards a flexible attribute-based fashion query system. *Pattern Recognition Letters*, 112:212–218, 2018. 1

[5] Martin Arjovsky, Soumith Chintala, and Léon Bottou. Wasserstein generative adversarial networks. In *ICML*, pages 214–223, 2017. 4

[6] Andrew Brock, Theodore Lim, James M Ritchie, and Nick Weston. Neural photo editing with introspective adversarial networks. *arXiv:1609.07093*, 2016. 2

[7] Yunjey Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*, June 2018. 1, 2, 5

[8] Alexey Dosovitskiy and Thomas Brox. Generating images with perceptual similarity metrics based on deep networks. In *Advances in Neural Information Processing Systems*, pages 658–666, 2016. 2

[9] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016. 2

[10] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NIPS*, pages 2672–2680, 2014. 1, 2

[11] Ishaan Gulrajani, Faruk Ahmed, Martin Arjovsky, Vincent Dumoulin, and Aaron C Courville. Improved training of wasserstein gans. In *NIPS*, pages 5767–5777, 2017. 4

[12] Xintong Han, Zuxuan Wu, Zhe Wu, Ruichi Yu, and Larry S Davis. Viton: An image-based virtual try-on network. In *CVPR*, 2018. 2, 3

[13] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, pages 770–778, 2016. 5, 6

[14] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 1, 2, 5

[15] Nikolay Jetchev and Urs Bergmann. The conditional analogy gan: Swapping fashion articles on people images. In *ICCVW*, page 8, 2017. 2

[16] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711. Springer, 2016. 2

[17] Diederik P Kingma and Max Welling. Auto-encoding variational bayes. *arXiv:1312.6114*, 2013. 5

[18] Gustav Larsson, Michael Maire, and Gregory Shakhnarovich. Learning representations for automatic colorization. In *ECCV*, pages 577–593. Springer, 2016. 2

[19] Christoph Lassner, Gerard Pons-Moll, and Peter V Gehler. A generative model of people in clothing. In *ICCV*, volume 6, 2017. 2

[20] Ziwei Liu, Ping Luo, Shi Qiu, Xiaogang Wang, and Xiaoou Tang. Deepfashion: Powering robust clothes recognition and retrieval with rich annotations. In *CVPR*, pages 1096–1104, 2016. 1, 2, 6

[21] Mehdi Mirza and Simon Osindero. Conditional generative adversarial nets. *arXiv:1411.1784*, 2014. 2

[22] Augustus Odena, Christopher Olah, and Jonathon Shlens. Conditional image synthesis with auxiliary classifier gans. *arXiv:1610.09585*, 2016. 2

[23] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, pages 2536–2544, 2016. 2

[24] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. Ganimation: Anatomically-aware facial animation from a single image. In *ECCV*, pages 818–833, 2018. 1, 2, 3, 5, 7

[25] Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep convolutional generative adversarial networks. *arXiv:1511.06434*, 2015. 2

[26] Amit Raj, Patsorn Sangkloy, Huiwen Chang, James Hays, Duygu Ceylan, and Jingwan Lu. Swapnet: Image based garment transfer. In *ECCV*, pages 679–695. Springer, Cham, 2018. 2

[27] Scott Reed, Zeynep Akata, Xinchen Yan, Lajanugen Logeswaran, Bernt Schiele, and Honglak Lee. Generative adversarial text to image synthesis. *arXiv:1605.05396*, 2016. 2

[28] Tim Salimans, Ian Goodfellow, Wojciech Zaremba, Vicki Cheung, Alec Radford, and Xi Chen. Improved techniques for training gans. In *NIPS*, pages 2234–2242, 2016. 2

[29] Othman Sbai, Mohamed Elhoseiny, Antoine Bordes, Yann LeCun, and Camille Couprie. Design: Design inspiration from generative networks. *arXiv:1804.00921*, 2018. 2

[30] Bochao Wang, Huabin Zheng, Xiaodan Liang, Yimin Chen, Liang Lin, and Meng Yang. Toward characteristic-preserving image-based virtual try-on network. In *ECCV*, 2018. 2

[31] Wenqi Xian, Patsorn Sangkloy, JINGWAN Lu, CHEN Fang, FISHER Yu, and JAMES Hays. Texturegan: Controlling deep image synthesis with texture patches. In *CVPR*, 2018. 2

[32] Tao Xu, Pengchuan Zhang, Qiuyuan Huang, Han Zhang, Zhe Gan, Xiaolei Huang, and Xiaodong He. Attngan: Fine-grained text to image generation with attentional generative adversarial networks. In *CVPR*, 2018. 2

[33] Jianwei Yang, Anitha Kannan, Dhruv Batra, and Devi Parikh. Lr-gan: Layered recursive generative adversarial networks for image generation. In *ICLR*, 2017. 3

[34] Gang Zhang, Meina Kan, Shiguang Shan, and Xilin Chen. Generative adversarial network with spatial attention for face attribute editing. In *ECCV*, September 2018. 1, 2, 5, 7

[35] Han Zhang, Tao Xu, Hongsheng Li, Shaoting Zhang, Xiaogang Wang, Xiaolei Huang, and Dimitris Metaxas. Stackgan: Text to photo-realistic image synthesis with stacked generative adversarial networks. In *ICCV*, 2017. 2

[36] Bo Zhao, Bo Chang, Zequn Jie, and Leonid Sigal. Modular generative adversarial networks. In *ECCV*, September 2018. 2

[37] B. Zhao, J. Feng, X. Wu, and S. Yan. Memory-augmented attribute manipulation networks for interactive fashion search. In *CVPR*, 2017. 1

[38] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Learning deep features for discriminative localization. In *CVPR*, pages 2921–2929, 2016. 3, 4

[39] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*, 2017. 1, 2, 4, 5

[40] Shizhan Zhu, Sanja Fidler, Raquel Urtasun, Dahua Lin, and Chen Change Loy. Be your own prada: Fashion synthesis with structural coherence. In *ICCV*, 2017. 2, 6