

Retrieval Guided Unsupervised Multi-domain Image-to-Image Translation

Raul Gomez*

Centre Tecnològic de Catalunya
Computer Vision Center, Spain
raul.gomez@cvc.uab.es

Dimosthenis Karatzas

Universitat Autònoma de Barcelona
Computer Vision Center, Spain
demos@cvc.uab.es

Yahui Liu*

University of Trento, Italy
Fondazione Bruno Kessler, Italy
yahui.liu@unitn.it

Bruno Lepri

Fondazione Bruno Kessler, Italy
lepri@fbk.eu

Marco De Nadai

Fondazione Bruno Kessler, Italy
denadai@fbk.eu

Nicu Sebe

University of Trento, Italy
Huawei Research, Ireland
niculae.sebe@unitn.it

ABSTRACT

Image to image translation aims to learn a mapping that transforms an image from one visual domain to another. Recent works assume that images descriptors can be disentangled into a domain-invariant content representation and a domain-specific style representation. Thus, translation models seek to preserve the content of source images while changing the style to a target visual domain. However, synthesizing new images is extremely challenging especially in multi-domain translations, as the network has to compose content and style to generate reliable and diverse images in multiple domains. In this paper we propose the use of an image retrieval system to assist the image-to-image translation task. First, we train an image-to-image translation model to map images to multiple domains. Then, we train an image retrieval model using real and generated images to find images similar to a query one in content but in a different domain. Finally, we exploit the image retrieval system to fine-tune the image-to-image translation model and generate higher quality images. Our experiments show the effectiveness of the proposed solution and highlight the contribution of the retrieval network, which can benefit from additional unlabeled data and help image-to-image translation models in the presence of scarce data.

CCS CONCEPTS

- Computing methodologies → Computer vision; Visual content-based indexing and retrieval; Machine learning.

KEYWORDS

GANs, image-to-image translation, retrieval system, unsupervised learning

*Both authors contributed equally to this research.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

MM '20, October 12–16, 2020, Seattle, WA, USA

© 2020 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-7988-5/20/10...\$15.00
<https://doi.org/10.1145/3394171.3413785>

ACM Reference Format:

Raul Gomez, Yahui Liu, Marco De Nadai, Dimosthenis Karatzas, Bruno Lepri, and Nicu Sebe. 2020. Retrieval Guided Unsupervised Multi-domain Image-to-Image Translation. In *Proceedings of the 28th ACM International Conference on Multimedia (MM '20), October 12–16, 2020, Seattle, WA, USA*. ACM, New York, NY, USA, 16 pages. <https://doi.org/10.1145/3394171.3413785>

1 INTRODUCTION

Humans are remarkably good at generalizing. For example, given a picture of a blonde woman we can easily imagine in our mind how she would look like with black hair, even if we have never seen her with that look. Great efforts have been made to mimic this ability with learning-based models, especially through Generative Adversarial Networks (GANs) [8]. In particular, unsupervised image-to-image translation models focus on transferring visual appearance from one domain (e.g. blonde hair people) to another (e.g. black hair people) learning from unpaired images (i.e. no ground truth of the transformation is available) [5, 12, 17, 19, 37]. However, despite the recent improvements in both quality and diversity of the results, it is still challenging to learn realistic transformations, especially with a limited amount of data.

Most of existing unsupervised models assume that all images can be grouped in visually distinctive categories (i.e. domains) having a domain-specific style and a domain-invariant content [7, 12, 19]. Thus, they try learning the latent spaces containing the content and the styles to synthesize images with the content of a source image and the style of a desired domain [5, 12, 19, 37]. As there is an infinite number of mappings between two unpaired images, these approaches require a large amount of data. This requirement is exacerbated in multi-domain problems, where a single model can map many domains. Despite recent results on few-shot transformations [2, 18], learning to generate visually realistic images in multiple domains and with limited data is still an open problem.

In this paper, we present a novel approach that helps to learn better mappings between domains. To do so, we propose the use of an image Retrieval model to Guide and improve UNsupervised Image-to-image Translation performance, i.e., RG-UNIT. Our model consists of three parts. First, an image-to-image translation model that learns to translate images to multiple domains by disentangling style and content. Second, an image retrieval system that exploits the image-to-image translation model to learn to find images similar to the source one, but in the target visual domain. Third and final,

we fine-tune the image-to-image translation model exploiting the information provided by the image retrieval system, which results in a performance improvement.

The contributions of this paper are as follows:

- We propose the use of a retrieval system to improve image-to-image translation models at combining content and style latent spaces and thus generate higher quality images from multi-modal and multi-domain image-to-image translations;
- To our knowledge, we are the first to train a retrieval system exploiting image-to-image translation model generated images and then use it to fine-tune the image translation model, a technique that can take advantage of unlabeled data;
- We validate the proposed solution in the challenging task of multi-domain translation of facial attributes on CelebA [20] dataset, where style is defined as a combination of attributes. Quantitative and qualitative results show better performance in all experiments, especially when additional unlabeled data is exploited.

2 RELATED WORK

In this paper we leverage an image retrieval system to ease the difficult task of translating an image from one domain (e.g. blonde hair people) to another one (e.g. black hair people). Thus, our work is at the meeting point between image-to-image translation and image retrieval.

Image Translation. Image-to-image translation models usually generate new images through conditional Generative Adversarial Networks (cGANs) [8, 23], where an adversarial game between a generator and a discriminator helps learning to synthetize images in the target domain. Isola *et al.* [13] proposed an encoder-decoder network that tries to reconstruct the source image in the target domain, instead of reconstructing the source image in the original one. Wang *et al.* [34] later extended it to higher resolutions. A desirable property of image-to-image translation is the ability to generate multiple plausible translations [7], as there are many different translations of an image from one domain to another. For example, a nightlight picture might be translated to multiple daylight pictures with different weather/light conditions. Hence, Zhu *et al.* [38] mixed GANs and Variational Auto-Encoders (VAEs) to learn a latent distribution and later sample from this distribution. However, it is expensive to collect pairs of images in different domains (e.g. daylight \leftrightarrow nightlight scene with people in the same pose) and often impossible (e.g. a picture of the author of this paper 10 years from now). Thus, the most interesting results come undoubtedly from unsupervised image-to-image translation which learns a mapping from unpaired images (e.g. daylight \leftrightarrow nightlight pictures of different places). Liu *et al.* [17] assumed that all images share a domain-invariant latent space and force all images to map into it before translating them to the target domain. Zhu *et al.* [37] outperformed previous results with CycleGAN, a network that requires generated images to be translated back in the original domain after the translation. Mo *et al.* [24] extended it to multiple instances per image, while Huang *et al.* [12] assumed that domains share a common content space but different style spaces, which help achieving multi-modality in an unsupervised setting. Recently, literature extended the previous models [5, 19]

by modelling multiple domains at once (e.g. blonde person to be translated to black, brown hair colours). Choi *et al.* [5] proposed StarGAN which uses a domain label and a domain classifier to map images in different domains. Liu *et al.* [19] later proposed a unifying approach to multi-modal translation in multiple domains by using a VAE-like approach to model the latent styles of the domains with a Gaussian Mixture Model (GMM).

Image Retrieval. Image retrieval is the task of finding images similar to a given query. In the image-to-image scenario the query is composed by an image and distances between images in an embedding space are optimized using a triplet ranking loss [29, 32]. However, queries are not necessarily limited to images. Considerable effort has been made especially in using textual queries [14, 15, 25, 33], where researchers focus on learning alignments between images and textual representations. Others exploited image labels [27, 35] such as Siddiquie *et al.* [30] who used multi-attribute queries, modeling attributes correlations, and Scheirer *et al.* [28] focused on attribute-based visual similarity searches.

Queries might also be multi-modal, as in the case of finding images similar to a given picture but with some characteristics specified by a condition (usually a textual sentence). Smith *et al.* [31] proposed an image-to-image retrieval setup where image queries are graphically modified by user-mouse interaction. Gordo *et al.* [9] used textual captions associated with images to learn a shared embedding space for images and text. Gomez *et al.* [6] instead learned this embedding space from social media data in a weakly-supervised setup.

In this paper we aim to retrieve images relevant to complex queries composed by image content and image style encodings, with the peculiarity that those representations are extracted by pre-trained feature extractors and from synthetically generated images. In our experiments we work with face images, were style is defined by face attributes (i.e blond hair) and the content is the visual appearance not defined by those attributes. As far as we know, we are the first ones proposing that setup for a retrieval model training, and using it later to improve the image translation model performance. The recent work of Anoosheh *et al.* [3] is the only similar approach we know about. They propose to localize previously seen places through image retrieval systems but, instead of exploiting a large amount of labeled data, they use an image translation model to generate images of places under different weather conditions, and therefore enlarge their dataset. Unlike the mentioned method, we learn image representations in a multi-domain setting, exploring different image translation strategies, and we use the learned image retrieval model to improve the image translation performance.

3 OUR METHOD

Given a collection of image domains $\{X_1, X_2, \dots, X_K\}$, we aim at transforming an input image $x \in \mathbb{R}^{H \times W \times 3}$ belonging in domain X_i to domain X_j without requiring paired images (here i, j refer to domain indices). Following recent seminal work [12, 16, 19], we assume that each image x shares a domain-invariant content latent representation $c \in C$ and has a domain-specific latent representation (also called style) $s_i \in S_i$, where S is termed as the style latent space. Thus, for each image $x \in X_i$, we would like to generate an

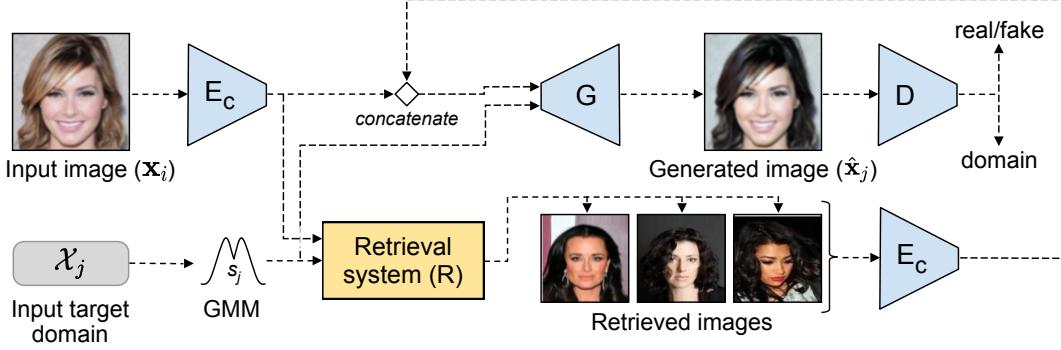


Figure 1: RG-UNIT training pipeline. We here translate an input image $\mathbf{x}_i \in \mathcal{X}_i$ to domain \mathcal{X}_j . We sample a style s_j from the latent GMM distribution, then we feed the generator with s_j and we merge with the content features of the input image ($E_c(\mathbf{x}_i)$) with the content features of the retrieved images.

image $\tilde{\mathbf{x}} \in \mathcal{X}_j$ with the same content as \mathbf{x} (e.g., a person looking left) but the style of \mathcal{X}_j (e.g., blonde hair).

Following the general framework GMM-UNIT [19], we model the style latent space through a K -component d -dimensional Gaussian Mixture Model (GMM). Formally, its probability density $p(\mathbf{s})$ is defined as:

$$p(\mathbf{s}) = \sum_{k=1}^K \phi_k \mathcal{N}(\mathbf{s}; \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \quad (1)$$

where ϕ_k denotes the weight ($\phi_k \geq 0$, $\sum_{k=1}^K \phi_k = 1$), $\boldsymbol{\mu}_k \in \mathbb{R}^d$ and $\boldsymbol{\Sigma}_k \in \mathbb{R}^{d \times d}$ refer to the mean and the covariance matrix of the Gaussian component k , respectively. Thus, each domain \mathcal{X}_i is represented by the i -th GMM component $\mathbf{s}_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$. The content latent space is instead modeled through the network, which is forced to understand the domain-invariant part of images.

We propose to use an image retrieval system to assist the network on this difficult task. Figure 1 depicts our architecture. The model is trained in an adversarial fashion and it's composed of a content encoder E_c , which extracts the content information from an image, a latent GMM distribution where we sample the style in a Variational Auto-Encoder (VAE) fashion, a retrieval system R , a generator G and a discriminator D . The network also has a style encoder E_s (not shown in the picture), which extracts the style from an image by predicting the parameters of the GMM. E_s will be used to learn the latent distribution.

The inputs of the model are a source image and a target domain. During a forward pass, the retrieval system R finds the r most similar images in content, but with the style of the target domain. Then, the content extracted from these images is used along with the content extracted from the input image and the target style to synthesize a new image. Our hypothesis is that the retrieved images can provide information to the decoder about how the input images should look like in the target domain, leading it to generate more realistic images.

Training together the retrieval system and the image translation models would be very challenging as we would need to jointly learn to disentangle content and style, generate new images and also retrieve the optimal images. Thus, we split the training in three stages. First, we train an image-to-image translation model

that is not guided by retrieved images ($r = 0$) following the GMM-UNIT network architecture [19]. Second, exploiting its generated images and its content and style features disentanglement, we train an image retrieval system. Finally, we train the RG-UNIT image translation model with the retrieval model guidance. In this last training stage, we freeze the encoders E_c, E_s because the retrieval model relies on them, so updating their weights would hurt the retrieval model performance.

3.1 Translating images

In this section we focus on the first training step of our method: The image-to-image translation model that does not rely on the image retrieval system. Translating images from one domain to another without paired images is a very challenging task, since the model has to learn to disentangle the style and the content from unpaired input images (i.e. different people in two domains).

Given an image $\mathbf{x} \in \mathcal{X}_i$ and a target image domain \mathcal{X}_j , let $\mathbf{s} \sim \mathcal{N}(\boldsymbol{\mu}_j, \boldsymbol{\Sigma}_j)$ be the latent style sampled from the GMM component. Then, we extract the content from the input image $\mathbf{c} = E_c(\mathbf{x})$ and we use it together with \mathbf{s} to synthesize a new image. Since there are many combinations of content and style that result in the input image, the problem has to be constrained through different losses.

Content and style disentanglement. Similar to [12, 19], we learn to disentangle content and style by requiring the content and style latent representations to be correctly reconstructed after the translation. So, we use a content encoder E_c and a style encoder E_s and impose:

$$\mathcal{L}_{c/recon} = \mathbb{E}_{\mathbf{x}, \mathbf{s}} [\|E_c(G(\tilde{\mathbf{c}}, \mathbf{s})) - E_c(\mathbf{x})\|_1] \quad (2)$$

$$\mathcal{L}_{s/recon} = \mathbb{E}_{\mathbf{x}, \mathbf{s}} [\|E_s(G(\tilde{\mathbf{c}}, \mathbf{s})) - E_s(\mathbf{x})\|_1] \quad (3)$$

where $E_s(\cdot)$ refers to the style extractor encoding images into style latent space, and $\tilde{\mathbf{c}}$ refers to the latent content. For the image-to-image translation model without retrieval $\tilde{\mathbf{c}} = E_c(\mathbf{x})$, we will modify this equivalence in Section 3.3 by leveraging the trained retrieval system.

Pixel-level reconstruction. To encourage the pixel-level consistency of generated and real images, we employ a reconstruction

loss and a cycle consistency loss [1, 37], generating an image and then translating it back to the original one.

$$\mathcal{L}_{x/recon} = \mathbb{E}_{\mathbf{x}, s} [\|G(\tilde{\mathbf{c}}, s) - \mathbf{x}\|_1] \quad (4)$$

$$\mathcal{L}_{cycle} = \mathbb{E}_{\mathbf{x}, s} [\|G(E_c(G(\tilde{\mathbf{c}}, s)), E_s(x)) - \mathbf{x}\|_1] \quad (5)$$

The \mathcal{L}_1 loss was found to encourage the generation of sharper images than the \mathcal{L}_2 [13].

Latent distribution. We aim to learn a style latent distribution from which we can sample a style code and translate one image from one domain to another. Inspired by the VAEs approaches, we encourage the encoder conditional distribution to match the prior latent distribution with the Kullback-Leibler divergence. Since the latent GMM we define is diagonal, it can be computed as:

$$\mathcal{L}_{KL} = \sum_{k=1}^K \mathbb{E}_{\mathbf{x} \sim p_{\mathbf{X}^k}} [\mathcal{D}_{KL}(E_s(\mathbf{x}) \| \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k))]$$

where $\mathcal{D}_{KL}(p\|q) = -\int p(t) \log \frac{p(t)}{q(t)} dt$ is the Kullback-Leibler divergence. This loss ultimately is expected to learn the true posterior of the latent distribution.

Domain loss. To support the network at learning to generate images in the correct domain, we employ a classification loss on the discriminator [5]. Thus,

$$\mathcal{L}_{cls}^D = \mathbb{E}_{\mathbf{x}, d_{\mathbf{X}_i}} [-\log D_{cls}(d_{\mathbf{X}_i} | \mathbf{x})],$$

$$\mathcal{L}_{cls}^G = \mathbb{E}_{\mathbf{x}, d_{\mathbf{X}_j}, s} [-\log D_{cls}(d_{\mathbf{X}_j} | G(\tilde{\mathbf{c}}, s))] \quad (6)$$

where $d_{\mathbf{X}}$ is the label of domain. The generator and discriminator are trained using the \mathcal{L}_{cls}^G and \mathcal{L}_{cls}^D loss, respectively.

Adversarial game. We train our model in an adversarial fashion with a Least Square loss [21]. We construct the adversarial game with:

$$\mathcal{L}_{r/f}^D = \mathbb{E}_{\mathbf{x}} [(D_{real}(\mathbf{x}))^2] + \mathbb{E}_{\mathbf{x}, s} [(D_{real}(G(\tilde{\mathbf{c}}, s)) - 1)^2] \quad (6)$$

$$\mathcal{L}_{r/f}^G = \mathbb{E}_{\mathbf{x}, s} [(D_{real}(G(\tilde{\mathbf{c}}, s)))^2] \quad (7)$$

where D_{real} is the discriminator predicting whether an image is real or fake.

The final optimization problem is defined as:

$$\mathcal{L}_D = \mathcal{L}_{r/f}^D + \mathcal{L}_{cls}^D$$

$$\begin{aligned} \mathcal{L}_G = & \mathcal{L}_{r/f}^G + \mathcal{L}_{c/recon} + \lambda_{s/recon} \mathcal{L}_{s/recon} + \mathcal{L}_{x/recon} \\ & + \lambda_{cycle} \mathcal{L}_{cycle} + \lambda_{KL} \mathcal{L}_{KL} + \mathcal{L}_{dom}^G \end{aligned}$$

where $\{\lambda_{s/rec}, \lambda_{cyc}, \lambda_{KL}\}$ are hyper-parameters of weights for the corresponding loss terms. The values of these parameters come from the literature. We refer to the Supplementary material for details.

3.2 Retrieval System

The style and content disentanglement is particularly difficult and often requires multiple constraints and tricks [12, 16, 19] to converge to a satisfactory result. To reduce the load on the generator, we use a retrieval system to fetch images similar in content to the input image but with the target style, which might help the decoder to generate more realistic images.

We train a network with three branches that share weights. Each one has as input content and style latent representations and outputs an embedded representation of the input information. We

train the network with a triplet ranking loss [4] to produce close representations for elements with a similar content and in the same domain. Specifically, given triplet samples $\langle \mathbf{r}_a, \mathbf{r}_p, \mathbf{r}_n \rangle$ (i.e. anchor, positive and negative samples, respectively), we train the model to achieve that the Euclidean distance between the anchor and the negative samples is greater (and bigger than a margin) than the Euclidean distance between the anchor and the positive samples. We define the loss as:

$$\mathcal{L}_{ret} = \max(0, m + d(\mathbf{r}_a, \mathbf{r}_p) - d(\mathbf{r}_a, \mathbf{r}_n)) \quad (8)$$

where $m > 0$ is a constant margin.

We use a pre-trained image-to-image translation model to get disentangled representations of images content and style (GMM-UNIT in our experiments [19]), which are the ones fed to the proposed retrieval system. The image-to-image translation model is also used to generate images for the training triplets. As far as we know, we are the first ones to use image-to-image translation models features' disentanglement and generated images to train an image retrieval system.

Specifically, given an input image $\mathbf{x} \sim p_{\mathbf{X}_i}$ with content $\mathbf{c} = E_c(\mathbf{x})$ and a target style $\mathbf{s}_t \sim \mathcal{N}(\boldsymbol{\mu}_t, \boldsymbol{\Sigma}_t)$, we define an anchor sample as $\mathbf{r}_a = (\mathbf{c}, \mathbf{s}_t)$, as shown in Figure 2(a). To get the positive sample, we use the pre-trained translation network to generate an image $\hat{\mathbf{x}} = G(\mathbf{c}, \mathbf{s}_t)$ in the target domain and define the triplet positive sample as $\mathbf{r}_p = (E_c(\hat{\mathbf{x}}), E_s(\hat{\mathbf{x}}))$ (see Figure 2(b)). The negative samples \mathbf{r}_n are defined with different competing strategies that lead the network to learn distances between samples based both on content and attributes. Being $\mathbf{x}_r \in \mathcal{X}_r$ a random training image, where $t \neq r$, we define the following strategies for the negatives samples mining:

- *Easy Negative:* $\mathbf{r}_n = (E_c(\mathbf{x}_r), E_s(\mathbf{x}_r))$. We use as negative the extracted content and style of the random image (see Figure 2(c)).
- *Medium Negative:* $\mathbf{r}_n = (E_c(\hat{\mathbf{x}}_r), E_s(\hat{\mathbf{x}}_r))$, where $\hat{\mathbf{x}}_r = G(E_c(\mathbf{x}_r), \mathbf{s}_t)$. We translate \mathbf{x}_r to the target style \mathcal{X}_t and use it as negative. These negative samples have the same style as the anchor and positive samples, but different content, as shown in Figure 2(d).
- *Hard Negative:* $\mathbf{r}_n = (E_c(\hat{\mathbf{x}}_r), E_s(\hat{\mathbf{x}}_r))$, where $\mathbf{s}_r \in \mathcal{S}_r$ and $\hat{\mathbf{x}}_r = G(\mathbf{c}, \mathbf{s}_r)$. We translate the anchor content to a random style \mathcal{S}_r and use it as negative. These negatives will have similar content as the anchor and positive samples, but different style (see Figure 2(e)). We also create another hard negative: $\mathbf{r}_n = (E_c(\tilde{\mathbf{x}}), E_s(\tilde{\mathbf{x}}))$, where $\mathbf{s}_i \sim \mathcal{N}(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ and $\tilde{\mathbf{x}} = G(\mathbf{c}, \mathbf{s}_i)$. We sample a transformation of \mathbf{x} in its original domain \mathcal{X}_i and use it as negative. These negatives will have similar content as the anchor and positive samples and will be in the same domain as \mathbf{x} .

The combination of these negatives' strategies forces the network to learn to differentiate between images that have different content but similar style (medium negatives) and between images that have similar content but different style (hard negatives), which in the end allows the network to rank images to fulfill its objective, i.e., retrieve images similar in content and style to a given query.

3.3 Combining Retrieval and Translation

As aforementioned, we train our model in three steps: 1) train a image-to-image model without retrieval guidance; 2) train a retrieval system leveraging the model trained in step 1; and 3) fine-tune the image-to-image translation model exploiting the retrieval

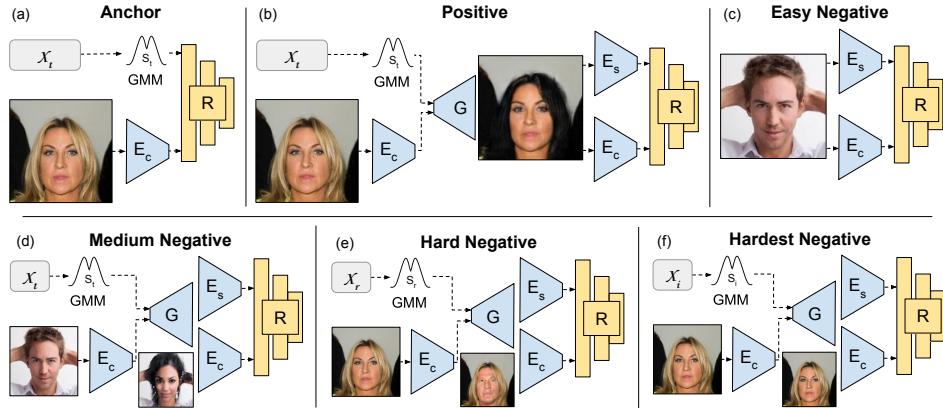


Figure 2: Retrieval model triplets creation pipeline. In this example, the target domain t is: Black Hair, Female, Young. The domain i is the one of the input image, and r is a randomly selected domain different than t . The box R stands for the retrieval model, which outputs 300 dimensional vectors that are compared using a ranking loss.

system. In this section we explain how we use the retrieval model to assist the image translation task in the model trained in step 3.

Given an input image $\mathbf{x} \in \mathcal{X}_i$ to be translated to domain \mathbf{X}_j we use the retrieval system to find images similar in content to $E_c(\mathbf{x})$ and with the s_j attributes of the target domain. To do that we compute the Euclidean distance between the pre-computed embeddings of the query image and the elements in our retrieval set. We keep the n most similar images $\{\mathbf{r}_1, \mathbf{r}_2, \dots, \mathbf{r}_n\}$ (in our experiments $n = 3$) and encode the content of retrieved images using $E_c: \{E_c(\mathbf{r}_1), E_c(\mathbf{r}_2), \dots, E_c(\mathbf{r}_n)\}$. Then, we concatenate these encodings in the channel dimension. To fuse the features of the retrieved images, we apply a channel-wise convolution. We concatenate that feature map with the content of the input image $E_c(\mathbf{x})$ in the channel dimension, and again apply a channel wise convolution that fuses the content information of the input image and the retrieved images. We can write the expression of the fused content information as:

$$\mathbf{c}_{total} = f(E_c(\mathbf{x}) \oplus f(E_c(\mathbf{r}_1) \oplus E_c(\mathbf{r}_2) \oplus \dots \oplus E_c(\mathbf{r}_n))) \quad (9)$$

where \oplus is the concatenation operation and f a channel-wise convolution. Finally, we modify the previously defined losses to feed them with \mathbf{c}_{total} instead of solely with the content of the input image, as we did in step 1. Thus, for Equation (2), Equation (3), Equation (4), Equation (5), Equation (6) and Equation (7) we replace the previously defined $\tilde{\mathbf{c}} = E_c(\mathbf{x})$ with $\tilde{\mathbf{c}} = \mathbf{c}_{total}$. Different from state-of-the-art methods [7, 12, 19], in this paper we feed to the generator not only the content features of the source image, but also those of the retrieved ones. Figure 1 shows the complete architecture of the retrieval-guided image-to-image translation model.

An attention mechanism is used as a simple yet efficient method to preserve the background pixels while manipulating some visual attributes, which has been proved in state-of-the-art methods [19, 22, 26]. We add a convolutional layer at the end of the decoder G to learn a one-channel attention mask \mathbf{M} . The final generated image $\hat{\mathbf{x}}$ is obtained through linearly combining the input image \mathbf{x} and its initial prediction $\tilde{\mathbf{x}}$: $\hat{\mathbf{x}} = \tilde{\mathbf{x}} * \mathbf{M} + \mathbf{x} * (1 - \mathbf{M})$.

4 EXPERIMENTS

We evaluate the proposed retrieval system to ensure it provides valuable information to the retrieval-guided image-to-image translation model. We use the CelebFaces Attributes (CelebA) dataset [20] to verify the efficacy of the proposed method. This dataset contains 202,599 face images of celebrities, each one annotated with 40 binary attributes. We crop the initial 178×218 images to 178×178 , and then resize them to 128×128 . We randomly select 2,000 images as test set and use all remaining images as training data. We follow the setting of the previous methods [5, 19] to construct seven domains using the following attributes: hair color (*black/blond/brown*), gender (*male/female*), and age (*young/old*).

As baseline we use GMM-UNIT [19], a multi-domain and multimodal model that offers to be a unified framework for all the multi-domain image-to-image translation models. As explained before, this model disentangles the content and style latent representations from images, and uses a GMM to model the style latent space. We did not test the similar MUNIT approach [12] which also disentangles content and style, as it is limited to one-to-one domain translations.

4.1 Evaluation

Image Retrieval. We use one query per each test set image with a target domain randomly selected from the testing set, and define several metrics to ensure that the retrieved images are similar in content to the query image and in style to the query attributes. Those metrics are based on Precision at 10 (P@10) which measures the similarity of the top 10 retrieval results with the query. We use P@10 to measure the following similarities:

Attributes Similarity. We evaluate the P@10 of each one of the attributes of the query with the groundtruth attributes of the retrieved images. Then, we average the P@10 of all attributes.

Content Similarity. CelabA groundtruth contains 40 attributes per image, from which only 5 are used to define domains in this work. This metric exploits the additional 35 attributes to provide an insight on how similar the images are in other features aside from the 5 attributes. Therefore, it is an insight of content similarity. It is



Figure 3: Top 2 retrieval examples for different queries. Queries are composed by the content of the query images, on the left, and the attributes resulting by modifying the input image attributes with the ones written on top of each column.

computed as the former metrics, but averaging the P@10 over the 35 unused attributes. Examples of those content attributes are *smiling*, *pointy nose* or *mustache*.

Average Similarity. Average between Attributes Similarity and Content Similarity to measure the overall performance, since we aim to train a retrieval system with a good compromise between them.

Image Translation. We quantitatively evaluate our model through image quality, diversity and domain correctness. Specifically, we use the Inception Score (IS) and the FrÃ©chet Inception Distance (FID) [11] for image quality, the LPIPS [36] for image diversity and the F1 score for the domain accuracy.

FID. Measures the distance in feature space between the real images and the generated ones. We estimate the FID using 10,000 input images vs their transformation in a random target domain. Lower FID values indicate better quality of synthesized images.

IS. Evaluates the quality of the generated images using their class probabilities computed by a pre-trained network, assuming that strong and varied classification scores indicate higher quality. We compute it using the same transformed images as for FID.

LPIPS. The LPIPS is defined as the \mathcal{L}_2 distance between the features extracted by a deep learning model of two images and it is a proxy for the perceptual distance between images [36]. Consistently with [12, 16, 38], we randomly select 100 input images and translate them to different domains by generating, for each domain, 10 images and evaluate the average LPIPS distance between the 10 generated images. Then, we average all distances to get the final score. Higher LPIPS distance indicates better diversity among the generated images.

ACC. To evaluate if the generated images have the desired attributes, i.e., they are in the target domain, we train an attribute classifier. We use a pre-trained ResNet-50 [10], modifying the last linear layer to have the same number of outputs as our number of attributes. We train the network in a multi-label classification setup using sigmoid activations and a cross-entropy loss. At testing time, we consider an attribute positive if its score is above 0.5, we evaluate the accuracy (ACC) of each attribute separately and we compute the mean of them.

5 RESULTS

5.1 Image Retrieval

We begin by evaluating our retrieval system, which is trained to retrieve images similar in content to the query one but with the style of the target domain. Figure 3 shows top two ranked images for different query images and target attributes. Note that the retrieved images have the desired style while keeping content similarity (visual appearance not related with the defined attributes) with the query image.

Table 1 shows the quantitative evaluation of our retrieval model trained with different triplet selection strategies. We observe that the model trained with a combination of easy, medium and hard negatives achieves the best performance. In particular, the introduction of hard negatives significantly boosts the similarity of the attributes at a small cost of content similarity, resulting in the best average performance. This trade-off between content and style similarity is motivated by the fact that hard negatives share the same content as the positives, thus guiding the network to learn image similarities based on style. These results are consistent for each domain and prove that the proposed triplet selection strategy is crucial to achieve a consistent retrieval performance. Readers can find additional results in the Supplementary Material.

Table 1: Quantitative results on the image retrieval system. For each training strategy we show the P@10 for attributes and content similarity, but also the average between the two.

Training Triplets	Attributes	Content	Average score
Easy Negatives	0.656	0.792	0.724
Medium Negatives	0.656	0.790	0.723
Hard Negatives	0.797	0.786	0.791
All	0.830	0.775	0.802
Random	0.648	0.754	0.701

5.2 Image Translation

We now move our attention to the main task: the image-to-image translation. Figure 4 shows some generated results for different input images and target styles. We can see that the proposed RG-UNIT model learns to translate images to multiple domains and that

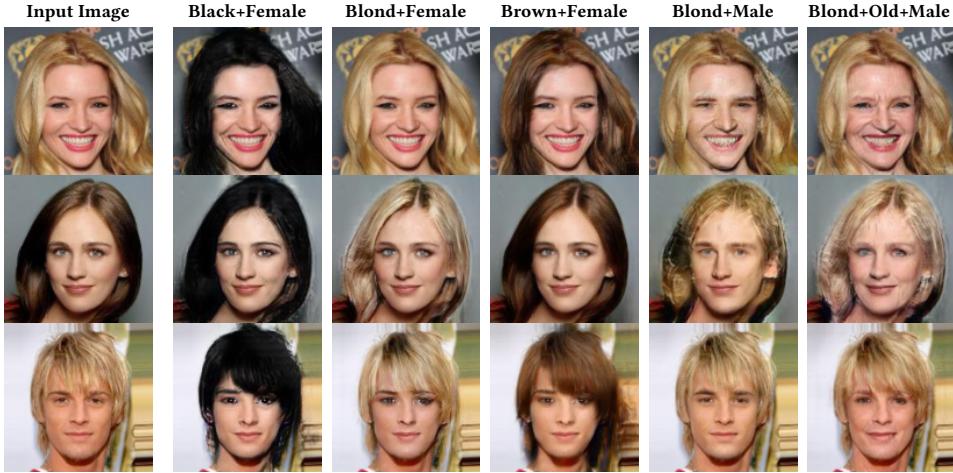


Figure 4: Qualitative image-to-image transformation results of the RG-UNIT model.

it can model multiple transformations at once (e.g. changing the gender and the age in a single pass). We observe that cross-gender translations are the most challenging transformations as CelebA is an unbalanced dataset, where some attributes are correlated with each other. As we use an attention mechanism to learn which pixels of the images have to be transformed, the generated results are clean and the transformation affects solely on the attributes to be changed. Figure 6 shows examples of the learned attention masks.

RG-UNIT models the latent style through a GMM, enabling to generate multiple plausible results of the same translation. For an input image and a target style, we can sample multiple latent codes from the latent distribution (the GMM) and generate the images. To verify this claim and that we do not achieve higher quality results than the baseline at the expense of diversity, we generate multiple samples of the same transformation. Thus, given an input image \mathbf{x} we sample multiple styles s_1, s_2, s_3 and generate multiple images $G(E(\mathbf{x}), s_1), G(E(\mathbf{x}), s_2), G(E(\mathbf{x}), s_3)$. Figure 5 shows different results for an input image and a target transformation. As it can be seen by zooming in, different styles of the faces, beard and hair can be seen among the results.

Moreover, a smooth and continuous latent space allows to interpolate between styles. Figure 7 shows an example where we first translate an input image to black hair and female, to then interpolate the latent space and get intermediate results of the transformation to blond hair.

The qualitative results are corroborated also by the quantitative ones. Table 2 shows that RG-UNIT significantly outperforms the framework GMM-UNIT in image quality, measured by both the FID and IS scores. The LPIPS diversity score is similar, corroborating that the retrieval system guidance does not hurt the image generation diversity.

We observe that the classification accuracy for all generated images is high for all the considered models, superior even to the accuracy achieved in real images, but a bit lower for RG-UNIT. We hypothesize that this result is a consequence of two main reasons. First, CelebA contains significant noise in the labels, which might

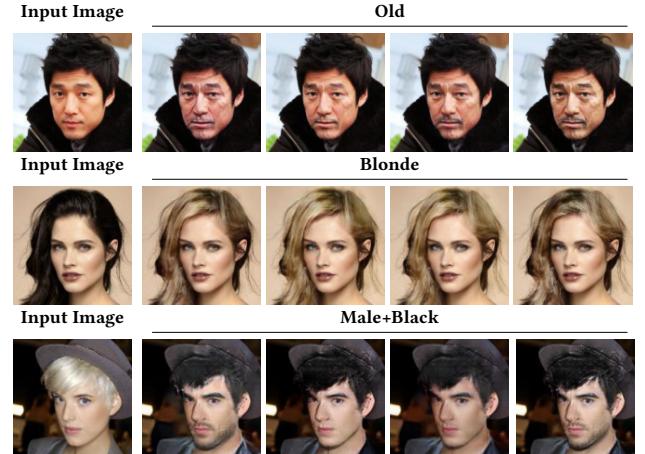


Figure 5: Different results for a given input image and target transformation. Zoom in for better details.

be alleviated in the generated images thanks to the domain classification loss. Second, the additional content information provided by the retrieval images might hurt the accuracy of the generated images as the decoder might rely more on the content than on the style to generate images.

To further prove the contribution of the retrieval system, we test a RG-UNIT version where, instead of using the images top ranked by the retrieval system to assist the training, we retrieve random images. As expected this model gets worse image quality results, but better accuracy (second row of Table 2). We refer to the Supplementary Material for additional per-class accuracy.

Altogether, our results show that the retrieval model guidance is able to boost the image-to-image translation performance.

Exploiting Unlabeled Data. The image-to-image translation learning addressed here is unsupervised, meaning that it learns to map one visual domain to another from unpaired images (e.g. two different people with two different visual domains). However, it still

Table 2: Quantitative results for GMM-UNIT, the proposed RG-UNIT, and a model with the same architecture as RG-UNIT but provided with random retrieved images.

Model	FID↓	IS↑	LPIPS↑	ACC↑
GMM-UNIT	30.04	3.051	0.048	97.0
RG-UNIT - Random Retrieval	31.08	2.967	0.036	98.2
RG-UNIT	27.61	3.160	0.042	94.72
Real images	-	3.470	-	93.1

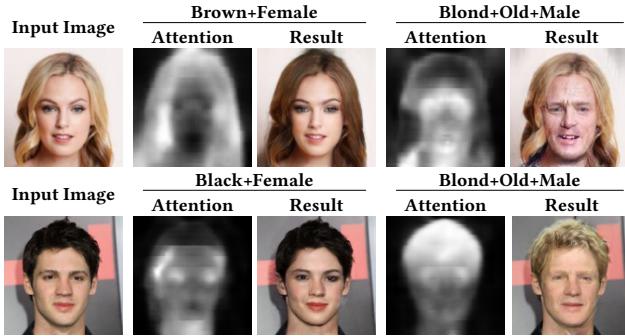


Figure 6: Unsupervised-learned attention of the translation.

relies on annotations that indicate what features are present in the images. For example, in CelebA we have binary annotations to indicate the gender of a face, the hair color, etc.

One of the main features of the proposed retrieval-guided method is that it can benefit from unlabeled data to train the image translation system. This can be easily achieved by feeding additional unlabeled data to the retrieval set during the third stage of the training (RG-UNIT training).

To simulate the scenario where we leverage additional unlabeled data, we train the baseline GMM-UNIT and retrieval models with a subset of the training dataset. Then, we train a RG-UNIT model using the same training dataset subset and the whole training data as retrieval set. This allows the translation network to benefit from additional unlabeled data through the image retrieval system guidance.

Table 3 shows image translation metrics for models trained with subsets of the original training data and using all the training data in the retrieval set. Results show that the smaller the dataset is and larger the additional retrieval data is the higher the contribution of the retrieval system to the translation model is. This experiment proves that the boost provided by the retrieval guidance is bigger when the training data are scarce, and that the proposed method can benefit from additional unlabeled data.

5.3 Ablation Study

We further investigate the role of the number of retrieved images in the performance of RG-UNIT. Table 4 shows that all models improve the baseline performance, but using three retrieved images seems to be the optimal setup.

Table 3: Image translation scores of models trained with a subset of the dataset and score shifts provided by the retrieval guidance.

	Data %	FID↓	ΔFID	IS↑	ΔIS
GMM-UNIT	25	38.83	4.92↓	2.648	0.431↑
RG-UNIT		33.91		3.079	
GMM-UNIT	50	35.62	2.90↓	2.726	0.524↑
RG-UNIT		32.72		3.250	
GMM-UNIT	100	30.04	2.43↓	3.051	0.109↑
RG-UNIT		27.61		3.160	

Table 4: Image translation results for RG-UNIT with different number of retrieved images (r).

Model	N	FID↓	IS↑	ACC↑
RG-UNIT	1	28.45	3.053	97.28
RG-UNIT	3	27.61	3.160	94.72
RG-UNIT	10	28.61	3.018	96.62
GMM-UNIT	-	30.04	3.051	97.0

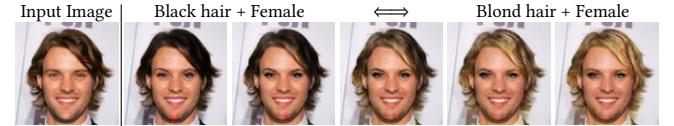


Figure 7: Domain interpolation given an input image.

6 CONCLUSIONS

In this paper, we presented a novel approach for unsupervised multi-domain image-to-image translation that uses a retrieval model to improve the quality of the generated images. First, we train an image translation model that learns to disentangle content and style from unpaired images. Then, we leverage these disentangled representations to train an image retrieval system. For that purpose, we design a novel triplet selection strategy exploiting the image translation model, which is proven to be crucial to achieving a superior retrieval performance. Then, we fine-tune the image-to-image translation model exploiting the retrieval model guidance and learn a better decoder. Our experiments show that the use of an image retrieval system improves the quality of image-to-image translation, especially when additional unlabeled data is used through the retrieval guidance. In particular, we observe that the improvement in performance increases as the dataset size decreases. As image-to-image translation models require a large amount of data, learning from unlabeled examples is of paramount importance. We hope that the proposed approach gives rise to further research on exploiting image retrieval systems to improve image translation models.

REFERENCES

- [1] Amjad Almahairi, Sai Rajeswar, Alessandro Sordoni, Philip Bachman, and Aaron Courville. 2018. Augmented cyclegan: Learning many-to-many mappings from

- unpaired data. In *ICML*.
- [2] Ivan Anokhin, Pavel Solovev, Denis Korzhenkov, Alexey Kharlamov, Taras Khakulin, Gleb Sterkin, Alexey Silvestrov, Sergey Nikolenko, and Victor Lempitsky. 2020. High-Resolution Daytime Translation Without Domain Labels. *arXiv preprint arXiv:2003.08791* (2020).
- [3] Asha Anoosheh, Torsten Sattler, Radu Timofte, Marc Pollefeys, and Luc Van Gool. 2019. Night-to-day image translation for retrieval-based localization. In *2019 ICRA*. IEEE, 5958–5964.
- [4] Gal Chechik, Varun Sharma, Uri Shalit, and Samy Bengio. 2010. Large scale online learning of image similarity through ranking. *Journal of Machine Learning Research* 11, Mar (2010), 1109–1135.
- [5] Yunjeong Choi, Minje Choi, Munyoung Kim, Jung-Woo Ha, Sunghun Kim, and Jaegul Choo. 2018. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In *CVPR*. 8789–8797.
- [6] Raul Gomez, Lluís Gomez, Jaume Gibert, and Dimosthenis Karatzas. 2019. Self-Supervised Learning from Web Data for Multimodal Retrieval. *Multimodal Scene Understanding* (2019).
- [7] Abel Gonzalez-Garcia, Joost Van De Weijer, and Yoshua Bengio. 2018. Image-to-image translation for cross-domain disentanglement. In *Advances in neural information processing systems*. 1287–1298.
- [8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. In *Advances in neural information processing systems*. 2672–2680.
- [9] Albert Gordo and Diane Larlus. 2017. Beyond Instance-Level Image Retrieval: Leveraging Captions to Learn a Global Visual Representation for Semantic Retrieval. *CVPR* (2017).
- [10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *CVPR*. 770–778.
- [11] Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter. 2017. Gans trained by a two time-scale update rule converge to a local nash equilibrium. In *NIPS*.
- [12] Xun Huang, Ming-Yu Liu, Serge Belongie, and Jan Kautz. 2018. Multimodal unsupervised image-to-image translation. In *ECCV*. 172–189.
- [13] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. 2017. Image-to-image translation with conditional adversarial networks. In *CVPR*. 1125–1134.
- [14] Andrej Karpathy, Armand Joulin, and Li Fei-Fei. 2014. Deep Fragment Embeddings for Bidirectional Image Sentence Mapping. In *NIPS*.
- [15] Ryan Kiros, Ruslan Salakhutdinov, and Richard S. Zemel. 2014. Unifying Visual-Semantic Embeddings with Multimodal Neural Language Models. *arXiv* (2014).
- [16] Hsin-Ying Lee, Hung-Yu Tseng, Jia-Bin Huang, Maneesh Singh, and Ming-Hsuan Yang. 2018. Diverse image-to-image translation via disentangled representations. In *ECCV*. 35–51.
- [17] Ming-Yu Liu, Thomas Breuel, and Jan Kautz. 2017. Unsupervised image-to-image translation networks. In *NIPS*. 700–708.
- [18] Ming-Yu Liu, Xun Huang, Arun Mallya, Tero Karras, Timo Aila, Jaakko Lehtinen, and Jan Kautz. 2019. Few-shot unsupervised image-to-image translation. In *ICCV*. 10551–10560.
- [19] Yahui Liu, Marco Du Nadai, Jian Yao, Nicu Sebe, Bruno Lepri, and Xavier Alameda-Pineda. 2020. GMM-UNIT: Unsupervised Multi-Domain and Multi-Modal Image-to-Image Translation via Attribute Gaussian Mixture Modeling. *arXiv preprint arXiv:2003.06788* (2020).
- [20] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. 2015. Deep learning face attributes in the wild. In *ICCV*. 3730–3738.
- [21] Xudong Mao, Qing Li, Haoran Xie, Raymond YK Lau, Zhen Wang, and Stephen Paul Smolley. 2017. Least squares generative adversarial networks. In *ICCV*. 2794–2802.
- [22] Youssef Alami Mejjati, Christian Richardt, James Tompkin, Darren Cosker, and Kwang In Kim. 2018. Unsupervised attention-guided image-to-image translation. In *NeurIPS*. 3693–3703.
- [23] Mehdi Mirza and Simon Osindero. 2014. Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784* (2014).
- [24] Sangwoo Mo, Minsu Cho, and Jinwoo Shin. 2019. Instance-aware Image-to-Image Translation. In *ICLR*.
- [25] Yash Patel, Lluís Gomez, Marçal Rusiñol, Dimosthenis Karatzas, and C V Jawahar. 2019. Self-Supervised Visual Representations for Cross-Modal Retrieval. In *International Conference on Multimedia Retrieval*.
- [26] Albert Pumarola, Antonio Agudo, Aleix M Martinez, Alberto Sanfeliu, and Francesc Moreno-Noguer. 2018. Ganimation: Anatomically-aware facial animation from a single image. In *ECCV*. 818–833.
- [27] Bharti S Satpute. 2013. Content-Based Face Image Retrieval Using Attribute-Enhanced Sparse Codewords. *International Journal of Computer Science and Information Technologies*, (2013).
- [28] Walter J Scheirer, Neeraj Kumar, Peter N Belhumeur, and Terrance E Boult. 2012. Multi-Attribute Spaces: Calibration for Attribute Fusion and Similarity Search. In *CVPR*.
- [29] Florian Schroff, Dmitry Kalenichenko, and James Philbin. 2015. FaceNet: A Unified Embedding for Face Recognition and Clustering. In *CVPR*.
- [30] Behjat Siddiquie, Rogerio S. Feris, and Larry S. Davis. 2011. Image ranking and retrieval based on multi-attribute queries. In *CVPR*.
- [31] Brandon M Smith, Shengqi Zhu, and Li Zhang. 2011. Face Image Retrieval by Shape Manipulation. In *CVPR*.
- [32] Jiang Wang, Yang Song, Thomas Leung, Chuck Rosenberg, Jingbin Wang, James Philbin, Bo Chen, and Ying Wu. 2014. Learning Fine-Grained Image Similarity with Deep Ranking. In *CVPR*.
- [33] Liwei Wang, Yin Li, Svetlana Lazebnik, and Slazebni@illinois Edu. 2016. Learning Deep Structure-Preserving Image-Text Embeddings. *CVPR*.
- [34] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. 2018. High-resolution image synthesis and semantic manipulation with conditional gans. In *CVPR*. 8798–8807.
- [35] Zhong Wu, Qifa Ke, Jian Sun, and Heung-Yeung Shum. 2010. Scalable Face Image Retrieval with Identity-Based Quantization and Multi-Reference Re-ranking. In *CVPR*.
- [36] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. 2018. The unreasonable effectiveness of deep features as a perceptual metric. In *CVPR*. 586–595.
- [37] Jun-Yan Zhu, Taesung Park, Phillip Isola, and Alexei A Efros. 2017. Unpaired image-to-image translation using cycle-consistent adversarial networks. In *ICCV*. 2223–2232.
- [38] Jun-Yan Zhu, Richard Zhang, Deepak Pathak, Trevor Darrell, Alexei A Efros, Oliver Wang, and Eli Shechtman. 2017. Toward multimodal image-to-image translation. In *NIPS*. 465–476.

A IMPLEMENTATION DETAILS

A.1 GMM-UNIT

The first step of our training pipeline is to train a GMM-UNIT [19] with the CelebFaces Attributes (CelebA) dataset [20]. We follow the setup proposed in previous works [5, 19] to construct seven domains using the following attributes: hair color (*black, blond, brown*), gender (*male/female*), and age (*young/old*).

We train GMM-UNIT with the default settings: Adam optimizer with $\beta_1 = 0.5$, $\beta_2 = 0.999$ and an initial learning rate of 0.0001, which is decreased by half every $2e5$ iterations. We use a batch size of 1 and set the loss weights to $\lambda_{s/rec} = 10$, $\lambda_{cyc} = 10$, $\lambda_{KL} = 0.1$ and $\lambda_{iso} = 0.1$. We use the domain invariant perceptual loss with weight 0.1 and apply random mirroring to the training images.

A.2 Retrieval system

The second step of our training pipeline is to train a retrieval model exploiting the image translation model trained in the first step. The retrieval model takes as inputs the content and attributes representations extracted by the image translation model. In our experiments with 128×128 images the content feature map size is [256, 32, 32] while the attributes representation is a vector with 40 dimensions, where we have 8 dimensions for each one of the 5 attributes. A series of convolutional layers are applied to the content features map until it is downsized to [32, 4, 4]. Then, it is flattened and concatenated with the attributes representations, and the output is processed by a 2 layers MLP that generate a representation of 100 dimensions. All convolutional and linear layers but the last one are followed by ReLu activations and linear normalization. The architecture of the retrieval model is described in table 5.

The model is trained with a batch size of 32, a margin m of 0.2 and a learning rate of 0.01 which is decayed by half every $10K$ iterations. Negatives samples for triplets are selected randomly from one of the categories explained in the main paper.

A.3 Generator Fine-tuning

The third and final step of our training pipeline is to fine-tune the image translation model trained in step one in a setup such that it can exploit the information of the images provided by the retrieval system, which have similar content as the input image but are in the target domain.

We encode the content of each one of the retrieved images using E_c and concatenate those encodings in the channel dimension which. In our experiments with 128×128 images, that concatenation results in a $[M * 256, 32, 32]$ feature map. To fuse the features of the retrieved images, we apply a convolution that reduces the dimensionality to [256, 32, 32]. Next, we concatenate that feature map with the content of the input image $E_c(I_i)$ in the channel dimension, and again apply a convolution that fuses the content information of the input image and the retrieved images and reduces the dimensionality of the fused feature maps to [256, 32, 32], the dimensions for content encoding in the image translation model trained in step one. The rest of the architecture and training procedure of the image-to-image translation model is kept intact, but using the content that resulted from encoding together the content of the

input image and the retrieved images, instead of only the content of the input image.

The training setup and parameters are the same as for GMM-UNIT [19] but training only the generator while keeping the encoder frozen. This is mandatory since the retrieval system uses the encoder to get the images content information, and updating it would cause the retrieval system to struggle. The architecture of the RG-UNIT model is described in table 8.

A.4 Source code

We release the source code of our model at <https://github.com/yhllleo/RG-UNIT>.

B ADDITIONAL QUANTITATIVE RESULTS

B.1 Retrieved Images Attributes Similarity

Table 6 shows the P@10 of each one of the attributes for retrieval models trained with different negatives strategies. As seen in the paper, it shows that the negatives selection strategy proposed in this work is crucial to improve the retrieval system performance for all the attributes.

B.2 Attribute Classification

Table 7 shows the attributes classifier accuracy for each one of the attributes on the images generated the GMM-UNIT model, our retrieval-guided RG-UNIT model, a model guided with random images (RG-UNIT RR) and real images.

Part	Input → Output Shape	Layer Information
R	($\frac{h}{4}, \frac{w}{4}, 256$) → ($\frac{h}{4}, \frac{w}{4}, 128$)	CONV-(N128, K3x3, S1, P1), IN, ReLU
	($\frac{h}{4}, \frac{w}{4}, 128$) → ($\frac{h}{8}, \frac{w}{8}, 128$)	CONV-(N128, K4x4, S2, P1), IN, ReLU
	($\frac{h}{8}, \frac{w}{8}, 128$) → ($\frac{h}{8}, \frac{w}{8}, 64$)	CONV-(N64, K3x3, S1, P1), IN, ReLU
	($\frac{h}{8}, \frac{w}{8}, 64$) → ($\frac{h}{16}, \frac{w}{16}, 64$)	CONV-(N64, K4x4, S2, P1), IN, ReLU
	($\frac{h}{16}, \frac{w}{16}, 64$) → ($\frac{h}{16}, \frac{w}{16}, 32$)	CONV-(N32, K3x3, S1, P1), IN, ReLU
	($\frac{h}{16}, \frac{w}{16}, 32$) → ($\frac{h}{32}, \frac{w}{32}, 32$)	CONV-(N32, K4x4, S2, P1), IN, ReLU
	($\frac{h}{32}, \frac{w}{32}, 32$) → ($\frac{h}{32}, \frac{w}{32}, 32$)	FLATTEN
	($\frac{h}{32}, \frac{w}{32}, 32$) → ($\frac{h}{32}, \frac{w}{32}, 32 + Zn$)	CONCAT-(Zn)
	($\frac{h}{32}, \frac{w}{32}, 32 + Zn$) → (512,)	FC-(512), ReLU
	(512,) → (512,)	FC-(512), ReLU, Dropout-(0.1)
	(512,) → (100,)	FC-(100)

Table 5: Retrieval Network architecture. We use the following notation: Z : the dimension of attribute vector, n : the number of the domains, N : the number of output channels, K : kernel size, S : stride size, P : padding size, CONV: a convolutional layer, FC: fully connected layer, FLATTEN: flattens a d-dimensional tensor into a 1-dimensional tensor, CONCAT-(c): concatenates two 1-dimensional tensors, c and the input.

Table 7: Accuracy of the attributes classifier evaluated on the real images from CelaB dataset, on the images generated by GMM-UNIT, by the retrieval-guided model RG-UNIT, and by a model guided with random retrieval images (RG-UNIT RR). BA refers to Black Hair attribute, BN to Blond Hair, BW to Brown Hair, G to Gender, A to Age and Avg to the average scores between them.

	Avg	BA	BN	BW	G	A
GMM-UNIT	97.0	97.7	98.5	93.2	97.3	98.3
RG-UNIT - RR	98.2	98.3	96.1	99.5	98.9	98.4
RG-UNIT	94.7	95.0	96.1	89.4	98.1	95.0
Real Images	93.1	94.0	98.3	87.0	94.4	91.8

Table 6: Image retrieval P@10 per attribute metric (and average between attributes). Note that this metric does not measure content similarity. We show results of the model we use to boost the image-to-image model (All), a model trained only with easy negatives (Easy Negatives), only easy and medium negatives (Medium Negatives) and easy, medium and hard negatives (Hard Negatives). Random refers to the metrics obtained retrieving random images.

	P@10					
	BA	BN	BW	G	A	Avg
Easy Negatives	0.636	0.778	0.666	0.530	0.671	0.656
Medium Negatives	0.640	0.779	0.674	0.521	0.665	0.655
Hard Negatives	0.759	0.902	0.756	0.796	0.770	0.797
All	0.810	0.919	0.737	0.841	0.841	0.830
Random Retrieval	0.640	0.768	0.676	0.509	0.647	0.648

C ADDITIONAL QUALITATIVE RESULTS

Figure 8 shows additional image retrieval results of the proposed retrieval system. Note how the retrieval system is able to find images with the query attributes but keeping the query image content, such as the mouth gesture or the head pose.

Figure 9 shows an image translation results comparison between the baseline GMM-UNIT and the proposed RG-UNIT. It evidences how the retrieval guidance helps generating more realistic images.

Figures 10, 11 show additional image translation results of the proposed RG-UNIT model.

Part	Input → Output Shape	Layer Information
E_c	$(h, w, 3) \rightarrow (h, w, 64)$	CONV-(N64, K7x7, S1, P3), IN, ReLU
	$(h, w, 64) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	CONV-(N128, K4x4, S2, P1), IN, ReLU
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	CONV-(N256, K4x4, S2, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), IN, ReLU
f	$(\frac{h}{4}, \frac{h}{4}, 256xM) \rightarrow (\frac{h}{4}, \frac{h}{4}, 256)$	CONV-(N256, K3x3, S1, P1), ReLU
	$(\frac{h}{4}, \frac{h}{4}, 256x2) \rightarrow (\frac{h}{4}, \frac{h}{4}, 256)$	CONV-(N256, K3x3, S1, P1), ReLU
E_z	$(h, w, 3) \rightarrow (h, w, 64)$	CONV-(N64, K7x7, S1, P3), ReLU
	$(h, w, 64) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	CONV-(N128, K4x4, S2, P1), ReLU
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	CONV-(N256, K4x4, S2, P1), ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{8}, \frac{w}{8}, 256)$	CONV-(N256, K4x4, S2, P1), ReLU
	$(\frac{h}{8}, \frac{w}{8}, 256) \rightarrow (\frac{h}{16}, \frac{w}{16}, 256)$	CONV-(N256, K4x4, S2, P1), ReLU
	$(\frac{h}{16}, \frac{w}{16}, 256) \rightarrow (1, 1, 256)$	GAP
	$(256,) \rightarrow (CZ,)$	FC-(NCZ)
	$(256,) \rightarrow (CZ,)$	FC-(NCZ)
G	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), AdaIN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), AdaIN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), AdaIN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{4}, \frac{w}{4}, 256)$	Residual Block: CONV-(N256, K3x3, S1, P1), AdaIN, ReLU
	$(\frac{h}{4}, \frac{w}{4}, 256) \rightarrow (\frac{h}{2}, \frac{w}{2}, 128)$	UPCONV-(N128, K5x5, S1, P2), LN, ReLU
	$(\frac{h}{2}, \frac{w}{2}, 128) \rightarrow (h, w, 64)$	UPCONV-(N64, K5x5, S1, P2), LN, ReLU
	$(h, w, 64) \rightarrow (h, w, 3)$	CONV-(N3, K7x7, S1, P3), Tanh
	$(h, w, 64) \rightarrow (h, w, 1)$	CONV-(N3, K7x7, S1, P3), Sigmoid
D	$(h, w, 3) \rightarrow (\frac{h}{2}, \frac{w}{2}, 64)$	CONV-(N64, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{2}, \frac{w}{2}, 64) \rightarrow (\frac{h}{4}, \frac{w}{4}, 128)$	CONV-(N128, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{4}, \frac{w}{4}, 128) \rightarrow (\frac{h}{8}, \frac{w}{8}, 256)$	CONV-(N256, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{8}, \frac{w}{8}, 256) \rightarrow (\frac{h}{16}, \frac{w}{16}, 512)$	CONV-(N512, K4x4, S2, P1), Leaky ReLU
	$(\frac{h}{16}, \frac{w}{16}, 512) \rightarrow (\frac{h}{16}, \frac{w}{16}, 1)$	CONV-(N1, K1x1, S1, P0)
	$(\frac{h}{16}, \frac{w}{16}, 512) \rightarrow (1, 1, n)$	CONV-(Nn, K $\frac{h}{16} \times \frac{w}{16}$, S1, P0)

Table 8: Image-to-image network architecture. We use the following notations: Z : the dimension of attribute vector, n : the number of attributes, N : the number of output channels, K : kernel size, S : stride size, P : padding size, CONV: a convolutional layer, GAP: a global average pooling layer, UPCONV: a $2 \times$ bilinear upsampling layer followed by a convolutional layer, FC: fully connected layer, M : the number of retrieved images. We set $C = 8$.



Figure 8: Top 2 retrieval examples of the validation set for different queries. Queries are composed by the content of the query images, on the left, and the attributes resulting on modifying the input image attributes with the ones written on top of each column.

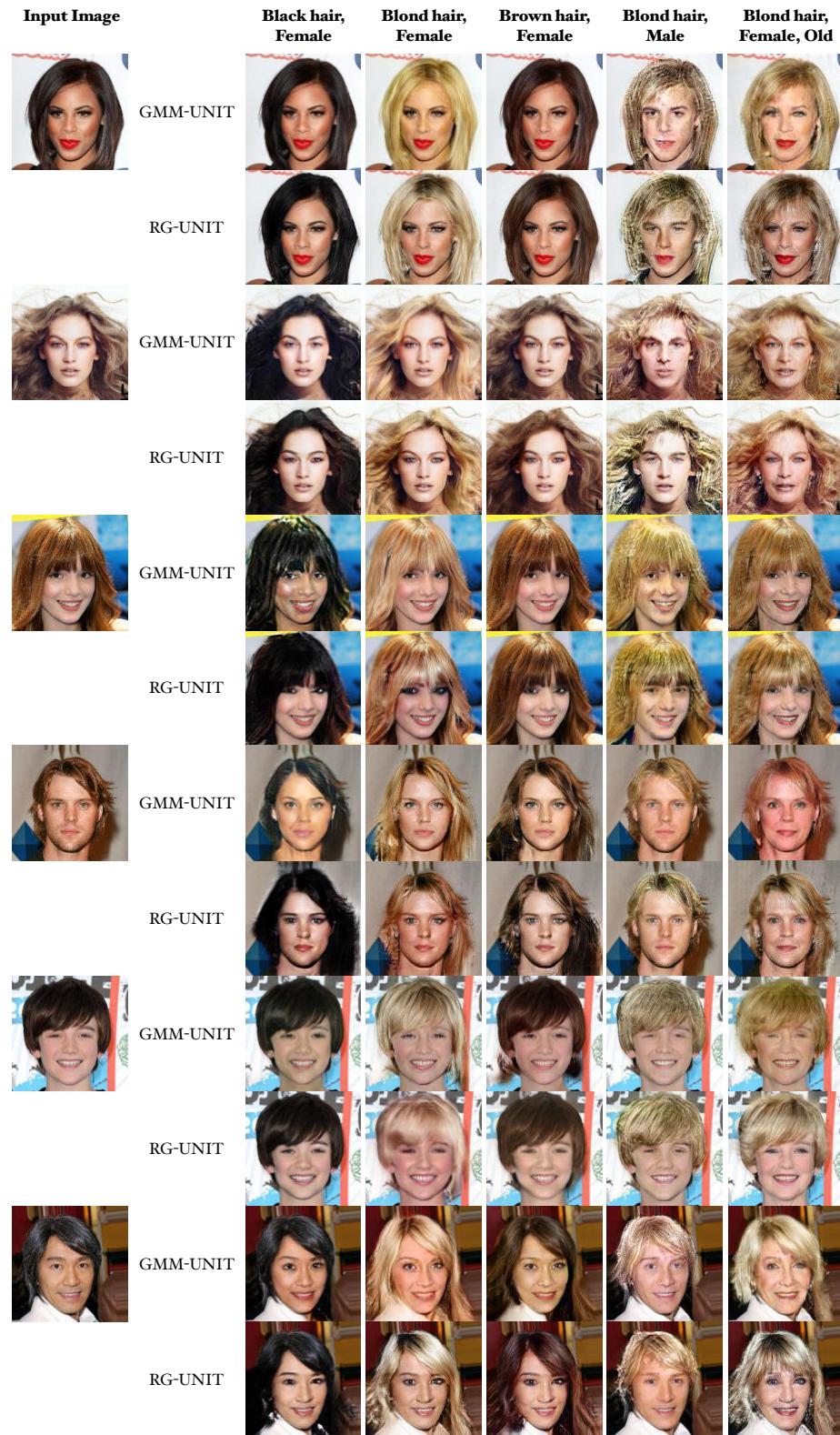


Figure 9: Visual comparisons between GMM-UNIT and RG-UNIT.



Figure 10: More female visual results of our method on CelebA dataset.



Figure 11: More male visual results of our method on CelebA dataset.