MDPI

Article

Masked Style Transfer for Source-coherent Image-to-Image Translation

Filippo Botti *, Tomaso Fontanini, Massimo Bertozzi, Andrea Prati

Department of Engineering and Architecture, University of Parma, 43124 Parma, Italy
* Correspondence: filippo.botti@unipr.it

Abstract: The goal of image-to-image translation (I2I) is to translate images from one domain to another while maintaining the content representations. A popular method for I2I translation involves the use of a reference image to guide the transformation process. However, most of the architectures fail to maintain the input main characteristics and produce images too close to the reference during style transfer. In order to avoid this problem, we propose a novel architecture which is able to perform source-coherent translation between multiple domains. Our goal is to preserve input details during I2I translation by weighting the style code obtained from the reference images before applying it to the source image. Therefore, we choose to mask the reference images in an unsupervised way, before extracting the style from them. By doing so, the input characteristics while performing style transfer are better maintained. As a result, we also increase the diversity in images generated by extracting style from the same reference. Additionally, the adaptive normalization layers, commonly used to inject styles into the model, are substituted with an attention mechanism for the purpose of increasing the quality in generated images. Several experiments are performed on CelebA-HQ and AFHQ datasets in order to prove the efficacy of the proposed system. Quantitative results, measured with LPIPS and FID metrics, demonstrate the superiority the proposed architecture compared to the state of art.

Keywords: deep learning; style transfer; image-to-image translation; generative adversarial networks

1. Introduction

Image-to-image translation (I2I) aims to generate an output image with a different style while preserving the content information of the input [1]. More specifically, the goal of I2I is to convert an image x_A, belonging to a *source* domain A, into an image y_B, belonging to a *target* domain B, by preserving its intrinsic contents belonging to the source domain and modifying its extrinsic contents by making them as similar as possible to those characterizing the target domain.

A lot of frameworks that use generative models to perform I2I translation are emerging in a variety of areas: from face editing [2], to style transfer [3] and automotive field [4]. Focusing on style transfer, StarGANv2 [5] introduced an innovative approach. Specifically, StarGANv2 incorporates a Style Encoder, designed to extract the style characteristics of an image referred to as the *reference image*. Subsequently, the extracted style is applied to the input image using a single Generator that is able to perform image translation across multiple domains. StarGANv2 also has a Mapping Network in charge of generating styles for the Generator from random noise. In their work, authors of StarGANv2 introduce diversity as the characteristic of each image within a domain to be different despite coming from the same domain. By this definition, the authors show how the output changes as the reference changes, even if picked from the same domain. However, this architecture design tends to apply global changes to the entire input image without preserving its intrinsic content representation. This can be described as a heavy form of *reference-based* style transfer, which can be seen in the output images generated by extracting the style from

Citation: Botti, F.; Fontanini, T.; Bertozzi, M.; Prati, A. Masked Style Transfer for Source-coherent Image-to-Image Translation. *Appl. Sci.* **2024**, *1*, 0. https://doi.org/

Received:
Revised:
Accepted:
Published:

the same reference. In such cases, the output images tend to closely mirror the reference image and the generated images collapse to the reference image as it can be seen in Fig. 1.

To address this limitation, we present a novel architecture which is able to perform *source-coherent* I2I translation across multiple domains. Our solution involves adding a segmentation layer before the Style Encoder: this layer computes segmentation masks which will be used to separate the subjects within the reference image and to select only the desired part of the image. In this way, we can remove all of the unnecessary content in the reference, like background or out-of-domain parts. Ultimately, the Style Encoder extracts the style only from the relevant part of the reference image and the Generator produces images with the style of the reference image without collapsing to it.

Our network architecture takes inspiration from StarGANv2 [5], though with some fundamental changes; in particular, we change the style application by using Cross Attention layers [6] and not Adaptive Instance Normalization (AdaIN) [7]; then, we adapted the Style Encoder by feeding it with both image and its correspondent mask.

Moreover, a crucial aspect lies in the utilization of an unsupervised architecture for extracting masks from reference images. Specifically, we choose to use STEGO [8] model, an architecture which can perform unsupervised semantic segmentation, in order to produce masks. With STEGO we produce binary images, which will be used in order to separate information on where to extract the style code inside the Style Encoder.

To summarize, the main contributions of the proposed work are the followings:

- **Innovative architecture for style transfer.** Introduction of a novel architecture which is able to perform source-coherent I2I translation between multiple domains by preserving input details and increasing diversity during generation.
- **Semantic style separation.** The model utilizes unsupervised segmentation architecture to produce masks in order to localize the style only on specific subjects of the images and removing useless areas like background or out-of-domain details. By this weighing of the reference images, the model is able to focus only on the relevant parts and better understand the characteristics of the style images, resulting in more accurate style code compared to the ones generated by state-of-art architectures.
- **Transferring style using Cross Attention.** The proposed architecture also shows how attention mechanisms, more in details Cross-Attention layers, are able to improve the quality of style transfer, with respect to commonly used Adaptive Instance Normalization layers.

2. Related work

Image-to-image translation. Image-to-image translation was first introduced by [9] as the task of translating one possible representation of a scene into another, given sufficient training data. Pix2Pix [10] was the first attempt to use GANs, in particular Conditional GANs (CGANs), in order to translate an image from source domain to target domain and viceversa with paired datasets. Later, CycleGAN [3] improved Pix2Pix performance by removing the requirement of paired datasets and suggested a method for I2I translation on unpaired datasets by employing a cycle consistency loss that guarantees that an image should accurately replicate the source image when it is translated to the target domain and then reversed. MUNIT [11] was one of the first attempt to enhance diversity of the generated images by feeding the generator with a style code that is randomly sampled from Gaussian noise. Later, MSGAN [12], tried to improve diversity of generated images by maximizing the ratio of the distance of two images in the image space with respect to the distance of their correspondent latent code in the latent space. StarGAN [13] reached better performance both on diversity and quality terms by using only a single generator to train between multiple domains. StarGANv2 [5] later improved StarGAN architecture by introducing a Style Encoder which is in charge to learn style from image and then use this style code in order to condition the output. Nevertheless, all of the cited architectures tend to share the same limitation of lack of diversity when using the same reference.

Diffusion Probabilistic Models (DPM) [14] has recently showed impressive results in the generative field. Despite this, DPMs are still not at the same level of GANs for I2I translation problems. Architectures like ControlNet [15], BBDM [16] or Palette [17] show good results for primitive form of I2I, but they lack of capacity to perform I2I from multiple domains. For this reason, in this paper, we chose to adapt StarGANv2 architecture to perform our task.

Style transfer. Style transfer is a way to perform I2I translation by generating a sample with the same content of the input image but with another style. In this way, we can translate images between multiple domains and preserving the input intrinsic characteristics. One of the first application used conditional GAN [18] in order to perform style transfer, but was based on a slow optimization process that iteratively updates the image to minimize content and style losses. Later, Adaptive Instance Normalization (AdaIN) [7] became the state of art for style transfer application. AdaIN enables fast arbitrary style transfer in real-time without being limited to a specific set of styles as in previous works. Recently, Transformers [19] exhibited impressive results in NLP and a lot of Transformer-based architecture have been used across a multitude of vision-related tasks. In particular, StyTr^2 [20] and Latent Diffusion [6] model highlighted the power of Transformers and Cross-Attention layers when used to transfer style from multimodal reference like text, class labels or images. For this reason, we selected Cross Attention layers in order to apply domain style to the generated input. Additionally, recent approaches leveraged the capability of Latent Diffusion to perform style transfer between pictures and paintings [21,22].

One of the main challenges during style transfer is to identify only the regions where to extrapolate the style and remove unnecessary regions like background or other parts of the image. [23] introduced an attention layer in order to select the area on where to apply style during I2I translation. [24] proposed a cycle consistent attention loss in order to train the model to apply changes on the same area during translation and reconstruction by using residual block activation map. Recently, SEAN [25] demonstrated that using a mask that represents only the relevant area of the image is possible to perform an average pooling operation on the extracted features inside the style encoder and to produce more accurate style codes. Following a similar idea, we propose to modify StarGANv2 style encoder by introducing the mask multiplication and pooling.

Unsupervised semantic segmentation. Semantic segmentation aims to discover and localize semantically meaningful categories present in an image. Tipically, Mask R-CNN [26] or YOLO [27] are used in order to produce segmentation from an image, but they require labelled datasets and this is not always feasible and, in any case, not scalable. Recently, several works introduced semantic segmentation systems that could learn from weaker forms of labels, such as classes, tags, bounding boxes, scribbles, or point annotations. IIC system [28] focuses on maximizing the mutual information of patch-level cluster assignments between an image and its augmentations. It operates as an implicit clustering method, with the network directly predicting the (soft) clustering assignment for each pixel-level feature vector [8,28,29]. PiCIE [29] enhances the semantic segmentation outcomes achieved by IIC by leveraging invariance to photometric effects and equivariance to geometric transformations as an inductive bias. In PiCIE, the network aims to minimize the distance between features subjected to different transformations. The distance metric is determined through an in-the-loop k-means clustering process [8,29]. Conversely, STEGO [8] gains impressive results in semantic segmentation without any kind of labelled dataset. STEGO shows that unsupervised deep network features have correlation patterns that are largely consistent with true semantic labels and uses these patterns to categorize every pixel of the image. Based on its valuable characteristics, STEGO is a perfect candidate for our purposes and it represents the state of art in unsupervised semantic segmentation.

Figure 1. Results generated using StarGANv2. It can be seen how the generated images lose input intrinsic characteristics like hair length or coat color and end up looking too similar to the reference, resulting in a lack of diversity.

3. Proposed system

In the next sections, the proposed model, which is able to perform source-coherent translation by preventing results to collapse to the references, will be described.

3.1. Network Architecture

As stated above, the network architecture of our system follows that of StarGANv2 and is composed by a Generator G, a Discriminator D, a Style Encoder E and a Mapping Network M (see Fig. 2). Style code s_{trg} can be generated both from images by using the Style Encoder or from random noise z by using the Mapping Network. During generation, the Generator G takes both source image x_{src} and style code s_{trg} and generates the output $x_{trg} = G(x_{src}, s_{trg})$. In finer details, G is designed as an encoder-decoder architecture featuring four downsampling residual blocks and four upsampling residual blocks, but uses Cross-attention layers (instead of Adaptive Instance Normalization layers) to apply style. The discriminator D serves the role of evaluating which domain the generated samples belong to. D follows StarGANv2 implementation, and is a multitask discriminator, which consists of multiple output branches. Finally, the style encoder E is a CNN with two residual blocks as features extractor and an average pooling layer on the area covered by the mask, while M is an MLP in charge of generating style codes from noise.

3.1.1. Generating Style from Reference

As previously stated, the Style Encoder is heavily modified w.r.t. the one introduced by StarGANv2. In fact, we took inspiration from SEAN Style Encoder [25] and adapted it to our goal. First, only two, instead of six, downsampling layers are used, following the implementation reported in [30], because we do not need shape information inside our style code. Then, we multiply the extracted features with the precomputed mask, computed from segmentation map obtained with STEGO (see section 3.3). This allows us to delete information irrelevant for the style computation, such as the background. Finally, an average pooling followed by a fully connected layer for each domain is applied (see Fig. 3(a)).

3.2. Transferring Style with Cross Attention

In order to apply style to the image, we use Cross-Attention layers instead of AdaIN during the decoding phase of the Generator. As shown in Fig. 3(b), we first extract features from source image x_{src} and style code s_{trg} extracted from x_{ref}. Subsequently, the features

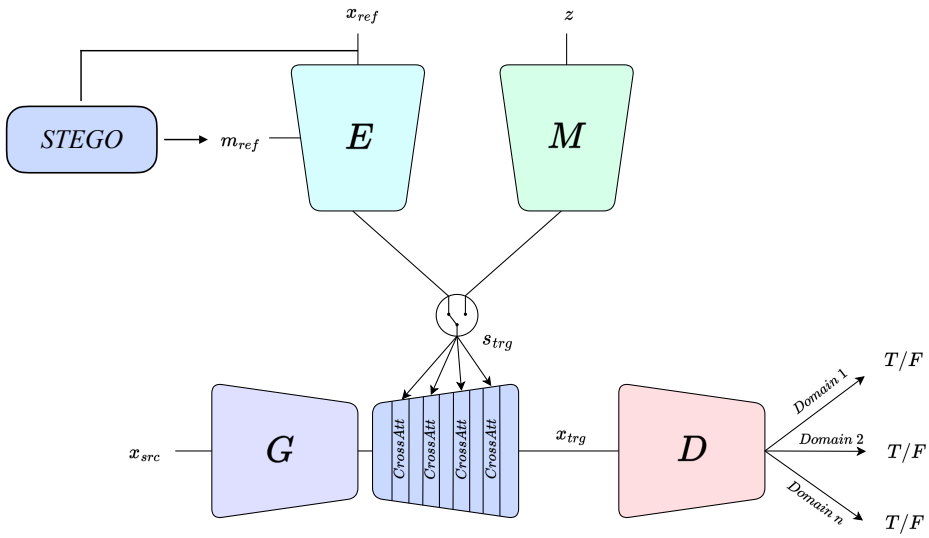

Figure 2. Overview of proposed system. The generator takes the input image x_{src} and applies the style code, computed using reference (x_{ref}) and its correspondent mask (m_{ref}) or using random noise (z), with Cross Attention layers. Finally, the result x_{trg} passes through the Discriminator.

extracted from the source image are normalized with a Layer Normalization and then, for every Cross-Attention, the style code is injected as follows:

$$Att(x, s) = Softmax\left(\frac{Q \cdot K^T}{\sqrt{d}}\right) \cdot V$$

where Q is the projection of the features extracted from x_{src}, K, V are projections of the style code s_{trg} and d is the dimension of a single attention head. Finally, the resulting tensors are normalized with Layer Normalization and linearly transformed with a Feed Forward layer.

3.3. Extracting Mask with STEGO

For the purpose of maintaining the network fully unsupervised, the unsupervised semantic segmentation architecture STEGO is employed for extracting masks from the reference image before extracting style from it. As described in [8], features f are first extracted from the reference image using DiNo [31] feature extractor, then STEGO segmentation head is devoted to extract a non-linear projection and to learn patterns inside the image. Finally, results are clustered and refined with a Conditional Random Field (CRF) layer [32].

Only the semantic cluster of the style that needs to be transferred is selected and the others are set to zero in the segmentation mask. More in detail, the selected semantic cluster/class is the one corresponding to the main subject of the image (*e.g.*, animals in AFHQ dataset [5], person in CelebA-HQ dataset [33]).

3.4. Training and Losses

In order to train the proposed model, we choose to maintain the training phase of StarGANv2 without any change. Therefore, the total loss is composed of four losses:

- *Adversarial loss* used to learn the generation of realistic results:

$$\mathcal{L}_{\text{adv}} = \mathbb{E}_{src}[\log D_{src}(x_{src})] + \mathbb{E}_{trg}\left[\log\left(1 - D_{trg}\left(G\left(x_{src}, s_{trg}\right)\right)\right)\right]$$

  where $x_{src} \in \mathcal{X}$ is the input image, $G(\cdot)$ is the generator that takes x_{src} and s_{trg}, which is the style code extracted from the reference image x_{ref}.

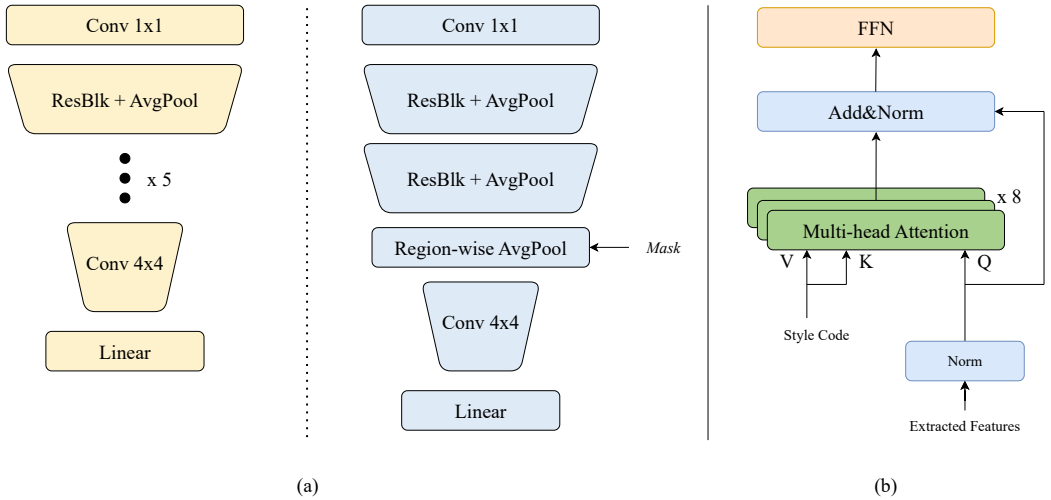

Figure 3. Overview of Style Encoder architecture. **(a)** Comparison between StarGANv2 Style Encoder (left) and the proposed Style Encoder (right). Differently from StarGANv2, we utilize Mask for computing the style code. **(b)** Overview of Cross Attention layers utilized for style transfer. We compute Cross Attention using extracted features of the input image as Query vector and the style code as Key and Value vector.

- *Style Reconstruction loss* introduced in order to avoid the generator G from ignoring the style s_{trg} during the generation phase:

$$\mathcal{L}_{sty} = \mathbb{E}_{src,trg}\left[\left\|s_{trg} - E\left(x_{trg}\right)\right\|\right]$$

  where $E\left(x_{trg}\right)$ is the style code extracted from the generated image.
- *Style Diversification loss* used to differentiate the style generated from two different images:

$$\mathcal{L}_{div} = \mathbb{E}_{src,trg_1,trg_2}\left[\left\|x_{trg_1} - x_{trg_2}\right\|\right]$$

- *Cycle Consistency loss* to maintain the domain-invariant characteristics of the generated image like pose and shape:

$$\mathcal{L}_{cyc} = \mathbb{E}_{src,trg}\left[\left\|x_{src} - G\left(x_{trg}, \tilde{s}_{src}\right)\right\|\right]$$

  where $\tilde{s}_{src}$ is the estimated style code extracted from the input.

The final loss is therefore as follows:

$$\min_{G,M,E} \max_{D} \mathcal{L}_{\text{adv}} + \lambda_{sty}\mathcal{L}_{sty} - \lambda_{div}\mathcal{L}_{div} + \lambda_{cyc}\mathcal{L}_{cyc}$$

It is worth noticing that, during training, a reference image and random noise are used alternatively for generating the style code through the mapping network.

4. Experimental Results

This section will report details on the experimental results, both qualitative and quantitative.

4.1. Selected Baseline

Since our work is an extension of StarGANv2, we decided to not compare the results that we produce with other architectures, following [30] comparison. In fact, our work can be seen as an improved version of StarGANv2 with the objective to show how to leverage the mistakes made by StarGANv2 and how to improve them by adding our masked style encoder. As discussed, StarGANv2 style transfer is limited in terms of diversity and tends to generate images with the same style applied, on the contrary we show a proper way

to perform style transfer without losing diversity. Moreover, is quite difficult to compare our architecture with others since, when considering diffusion models for example, the style transfer is a totally different task and cannot be compared to our architecture. All the hyperparameters and training strategies are the ones proposed in the original paper of StarGANv2.

4.2. Datasets

We tested our model on two datasets: CelebA-HQ, composed by 30k images [33] and AFHQ, composed by 16k images [5]. CelebA-HQ is organized in two domains (male and female) and AFHQ in three (cat, dog and wildlife animals). Binary masks are extracted using STEGO pretrained on COCOstuff [34] and selecting “person" and “animal" attributes in order to identify the subject of the image for the two datasets. No other information is employed during training or inference. We resized all images to 256×256 and all masks to 64×64 during training.

4.3. Implementation Details

During all the experiments, we trained the network for 100k iterations and we use Adam [35] as optimizer. Learning rates of 10^{-4} for G, D, E and 10^{-6} for M are used. Training took about 1 day on a single NVIDIA A100 GPU which is the same as the one for StarGANv2 proving that our approach does not add complexity in the training. For CelebA-HQ training, we weighed every loss equally; on the contrary, for AFHQ we set λ_{div} to 2 while λ_{cyc} and λ_{sty} to 1, following [5] implementation, in order to make an equal comparison with StarGANv2 and to show that the results obtained are better because of the architecture and not because of these hyperparameters.

4.4. Evaluation metrics

In order to evaluate our model we used Frechét Inception Distance (FID) [36] for image quality and Learned Perceptual Image Patch Similarity (LPIPS) [37] to measure diversity in the generated results. More in detail, FID metric measures the distance between two distribution and in our case is used in order to measure the distance between generated images, e.g. cat generated images, and test set that contains real images, e.g. real cat images. So, intuitively, a low value of FID means that two distributions are similar. Indeed, given two Gaussian distribution (m, C) and (m_w, C_w) the FID is computed as follow:

$$d^2((m, C), (m_w, C_w)) = ||m - m_w||_2^2 + \mathrm{Tr}(C + C_w - 2(CC_w)^{\frac{1}{2}})$$

The Learned Perceptual Image Patch Similarity (LPIPS) calculates perceptual similarity between two images. LPIPS essentially computes the similarity between the activations of two image patches for some pre-defined and pre-trained network. This measure has been shown to match human perception well. A low LPIPS score means that image patches are perceptual similar. Indeed, given two patches x and x_0 their distance is computed as follow:

$$d(x, x_0) = \sum_l \frac{1}{H_l W_l} \sum_{h,w} ||w_l \odot (\hat{y}^l_{hw} - \hat{y}^l_{0hw})||_2^2$$

where $\hat{y}^l_{hw}$ and $\hat{y}^l_{0hw}$ are the stacked features extracted from the patches. These features are normalized, and the distance between them is modulated by a learned weight vector w_l which adjusts the contribution of different feature channels [37].
Since the main contribution of our model is to perform source-coherent translation, which aims at improving diversity for image generated with the same reference, the evaluation was designed as follows:

- firstly, we randomly select one image for every domain as reference;
- secondly, given a set of source images, we generate samples with those reference images;
- thirdly, we compute FID and LPIPS (with consecutive pair of images);

- finally, we repeat this evaluation phase for 10 times in order to remove randomness in the results.

It is worth emphasizing that we decided to not use StarGANv2 FID algorithm, which calculates FID by using ten references for each domain, because we want to improve diversity in the results generated from a single reference. Therefore, FID is computed with only one reference per domain in order to evaluate the quality of images generated with our method.

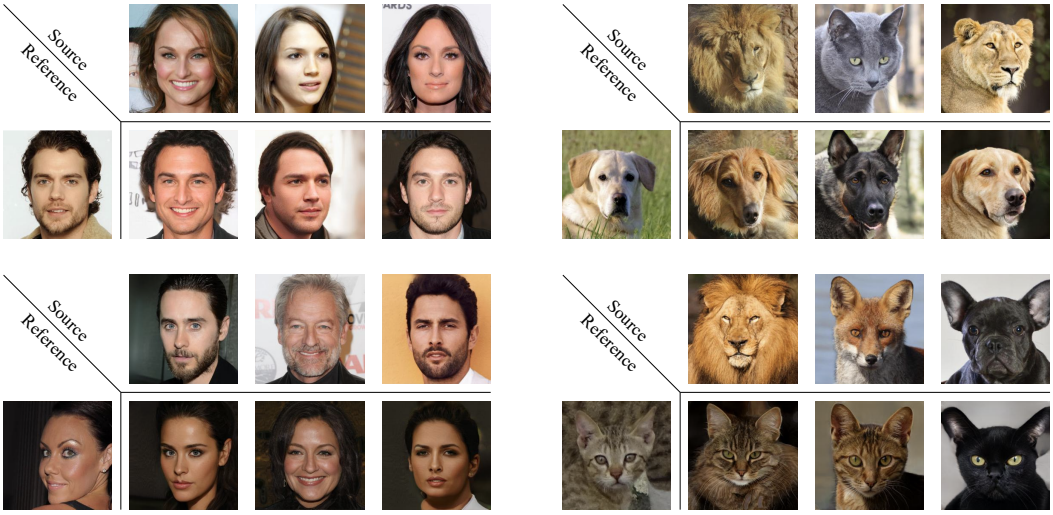

Figure 4. Main results obtained with our architecture. We can see how the results are indistinguible from real images and how they maintain the input intrinsic characteristics like expression, age or fur color.

4.5. Discussion

Depending on the dataset different styles are transferred. For CelebA-HQ the *male2female* and *female2male* where chosen. While for the AFHQ dataset we transferred *cat-dog2wildlife* and *wildlife2cat-dog*. As it can be seen from Fig. 4, the proposed architecture can perform I2I translation between multiple domains like StarGANv2, but gaining the capability to preserve input intrinsic characteristics during translation. Looking at the CelebA-HQ results, the proposed architecture maintains the input facial attributes, but applies changes to gender and hair color, which are taken from the reference images. In AFHQ results, our method maintains the same expression and preserves better fur color during translation, but changes the class of the animal. We introduced our work with the claim of increasing diversity in results generated using the same reference. This is shown clearly in Figs. 5 and 6, where the results are compared with the ones obtained using StarGANv2. From these examples, it is evident that StarGANv2 tends to collapse to the reference image and looses the majority of intrinsic attributes from input, except for the pose in AFHQ and the expression in CelebA-HQ. In contrast, our results maintain much more original details, such as fur color in AFHQ or age and hair style in CelebA-HQ. This leeds to more diversity and variety in generated images and in less reference-based generation.

More in details, StarGANv2 seems to capture attributes from reference, like hair color and length in Fig. 5, and apply them to the input in a rigid scheme that does not take care of the input attributes. On the contrary, the proposed method, in addition to understanding which are the main details from the reference, also considers the details from the input image before applying it. This leads, for instance, to a higher variety of hair lengths and not in a prefixed hair length, like StarGANv2. This is also visible in Fig. 6, where StarGANv2 produces the same animal with different poses. Our method, on the contrary, better understands the input characteristics and generates different breeds of dog/cats/wildlife animals based on the input breed.

All the previous considerations are valid also when the style code s_{trg} is sampled from random noise by using the Mapping Network M. This is presented in Fig. 7, where our architecture produces various and more source-coherent results than the ones generated by StarGANv2.

In order to support our claim we also show how our method produces similar results when similar reference images are employed, as shown in Fig. 8. This can be seen as a positive effect of our source-coherent method which does not ignore input attributes.

4.5.1. Quantitative results

Architecture	AFHQ		CelebA-HQ	
	FID ↓	LPIPS ↑	FID ↓	LPIPS ↑
StarGANv2 [2]	104.86	0.457	81.175	0.365
Ours (AdaIN)	76.15	**0.523**	57.67	0.420
Ours	**67.72**	0.517	**54.12**	**0.425**

Table 1. Quantitative comparison between StarGANv2 and our architecture. For our architecture we also include the model with AdaIN instead of Cross Attention Layers.

The above considerations are reflected in the quantitative results reported in Table 1. The proposed architecture significantly improves LPIPS results for both datasets. Furthermore, FID results highlight how our architecture produces much higher quality images.

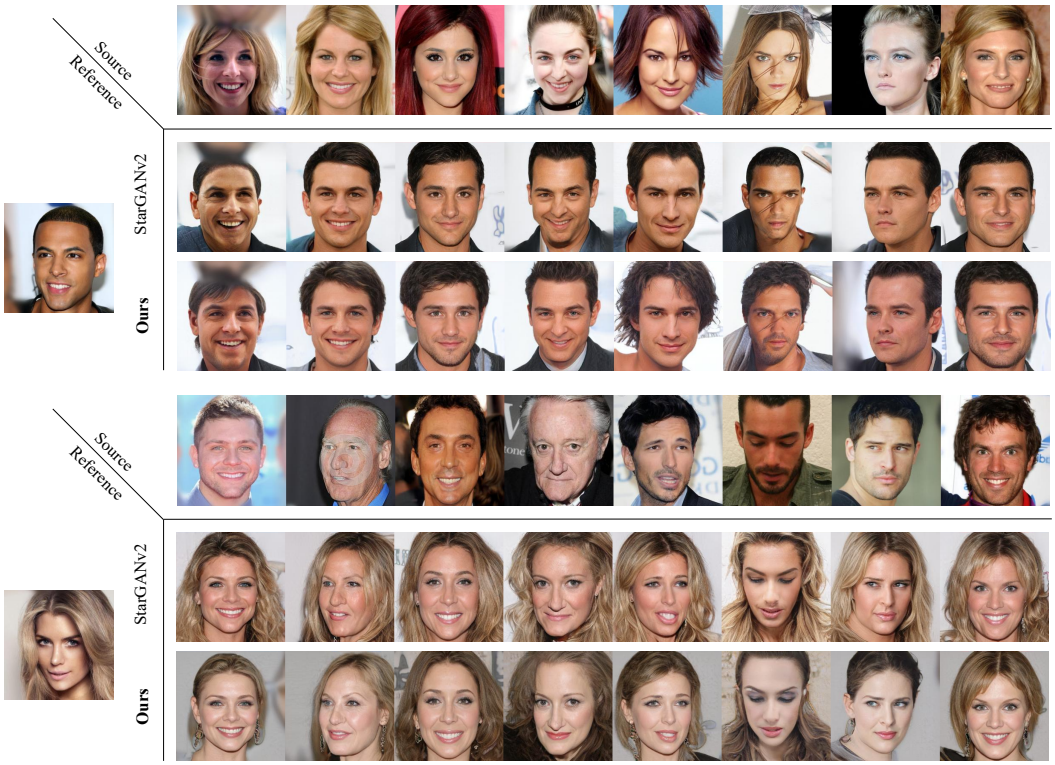

Figure 5. Comparison between StarGANv2 (first rows) and our architecture (second rows) on CelebA-HQ dataset.

Figure 6. Comparison between StarGANv2 (first rows) and our architecture (second rows) on AFHQ dataset.

Architecture	CelebA-HQ FID ↓
StarGANv2 [2]	**29.88**
Ours (AdaIN)	32.94
Ours	30.99

Table 2. Quantitative comparison between StarGANv2 and our architecture using StarGANv2 FID algorithm.

As shown in Table 2, we also compute FID using StarGANv2 algorithm on CelebA-HQ and we obtained opposite results. This is due to how FID works (also explained in Section 4.4): given the alignment of StarGANv2 generated images with the reference image shown in qualitative results, FID computed as in StarGANv2 original paper is natively lower. Indeed, FID tends to measure the difference between two distribution and since the images generated by StarGANv2 have less diversity than the ones generated by our architecture, the FID score in this case is better for StarGANv2 results. Nevertheless, in section 4.4 we

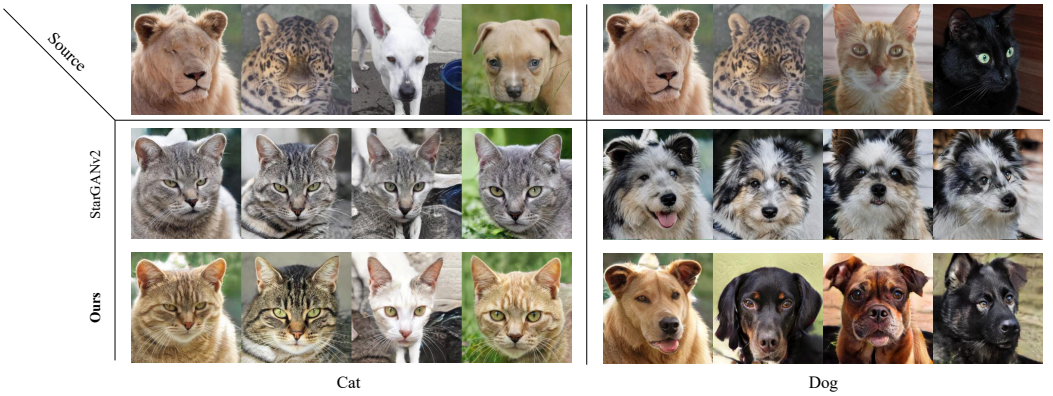

Figure 7. Comparison between StarGANv2 (first rows) and our architecture (second rows) on AFHQ dataset in results generated using random noise as a reference.

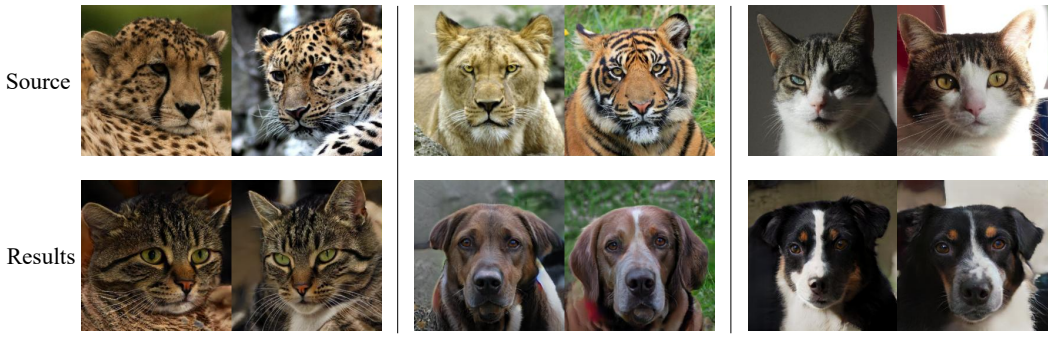

Figure 8. Similar input generates similar output due to the fact that we preserve input characteristics during translation.

justified how is not fair to compute FID in this way in order to consider the diversity in generated images. However, this comes at the cost of more limitations for StarGANv2 w.r.t. to our architecture, such as loosing input characteristics, lack of diversity in generated results and results collapsed to the reference images.

4.6. Ablation

Finally, we perform ablation studies to find the optimal configuration for our architecture. First, we try to transfer the style using AdaIN and not Cross Attention layers. As shown in Table 1 and Fig. 9, using AdaIN leads to slightly better results compared to StarGANv2 in terms of diversity, but the network still does not maintain the input characteristics like our final configuration. Additionally, we also tested employing 2 and 3 downsampling layers inside our Style Encoder, as it can be seen from second and third rows in Fig. 9. Indeed, the configuration with 3 downsampling layers tends to collapse more to the reference than the ones with 2 downsampling layer, as it can be seen from the fur style. Furthermore, by comparing these results with the ones produced by the final architecture, it is evident how Cross Attention layers improve the quality in generated results and better maintain input characteristics. Finally, results generated without masks are reported in the fourth row, proving that masks are necessary to identify major input information, like fur color or ear pose.

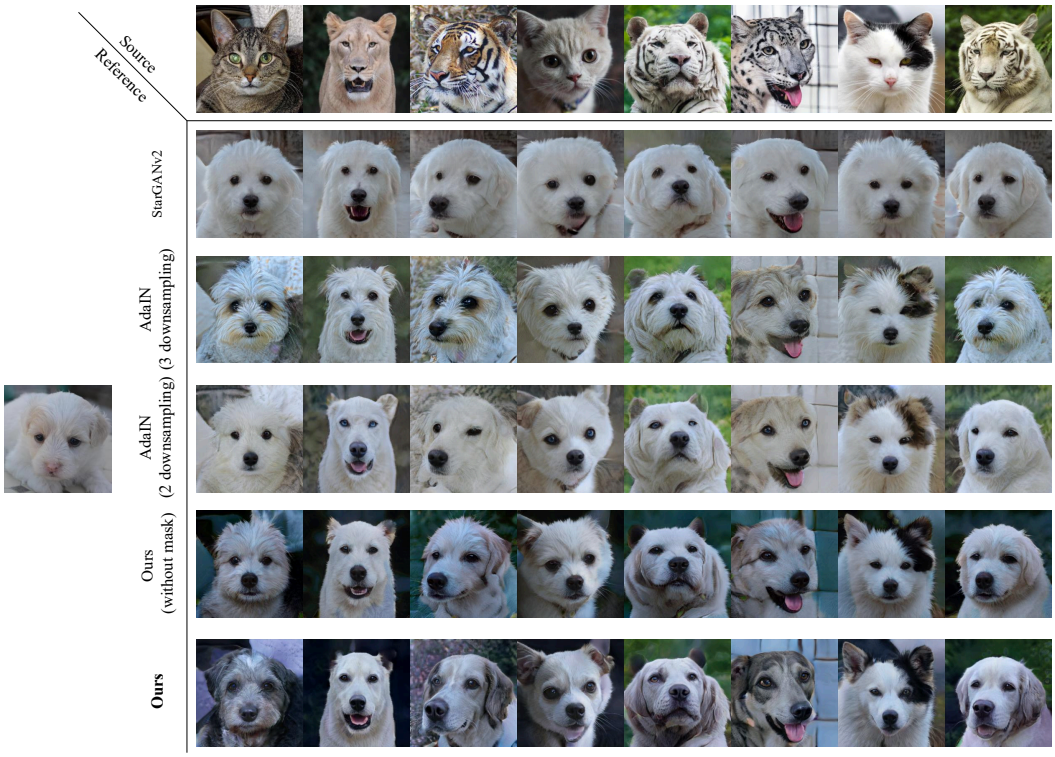

Figure 9. Difference between results generated with different architectures: first row with StarGANv2, second with our Style Encoder and AdaIN for style transfer, third with the same architecture as before but with 3 downsampling layers inside the Style Encoder, fourth with Cross Attention layers for style transfer but without masks and with 3 downsampling layers in the Style Encoder and finally with the proposed architecture.

5. Conclusions

For the task of I2I translation, StarGANv2 has shown limitations in preserving input details during translation. Additionally, StarGANv2 is not able to generate diverse samples when using the same reference image. For these reasons, this paper proposes a novel architecture for source-coherent image-to-image translation, which preserves input characteristics and increases diversity in the generated results. More in detail, style extracted from the reference images are masked in order for the model to focus only over the relevant information and are injected in the model using cross attention layers. By doing so, we managed to improve both quantitative and qualitative results.

Future works could focus on improving results generated using two different references and only one source, since our method, by preserving input intrinsic characteristics, tends to produce similar results when the same source image is utilized.

Author Contributions: Conceptualization, F.B. and T.F.; methodology, F.B. and T.F.; software, F.B.; validation, T.F.; formal analysis, F.B. and T.F.; investigation, F.B.; resources, F.B.; data curation, F.B.; writing—original draft preparation, F.B.; writing—review and editing, F.B. and T.F. and M.B. and A.P..; visualization, F.B.; supervision, T.F. and M.B. and A.P.; project administration, F.B. and T.F. and M.B. and A.P. All authors have read and agreed to the published version of the manuscript.

Informed Consent Statement: Not Applicable.

Data Availability Statement: Both CelebA-HQ and AFHQ datasets can be downloaded from: https://github.com/clovaai/stargan-v2 (accessed on 23 July 2024).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Pang, Y.; Lin, J.; Qin, T.; Chen, Z. Image-to-Image Translation: Methods and Applications. *IEEE Transactions on Multimedia* **2022**, *24*, 3859–3881. https://doi.org/10.1109/TMM.2021.3109419.
2. Li, Y.A.; Zare, A.; Mesgarani, N. Starganv2-vc: A diverse, unsupervised, non-parallel framework for natural-sounding voice conversion. *arXiv preprint arXiv:2107.10394* **2021**.
3. Zhu, J.; Park, T.; Isola, P.; Efros, A.A. Unpaired Image-to-Image Translation using Cycle-Consistent Adversarial Networks. *CoRR* **2017**, *abs/1703.10593*, [1703.10593].
4. Zheng, Z.; Wu, Y.; Han, X.; Shi, J. ForkGAN: Seeing into the Rainy Night. In Proceedings of the The IEEE European Conference on Computer Vision (ECCV), August 2020.
5. Choi, Y.; Uh, Y.; Yoo, J.; Ha, J.W. Stargan v2: Diverse image synthesis for multiple domains. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 8188–8197.
6. Rombach, R.; Blattmann, A.; Lorenz, D.; Esser, P.; Ommer, B. High-resolution image synthesis with latent diffusion models. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 10684–10695.
7. Huang, X.; Belongie, S. Arbitrary style transfer in real-time with adaptive instance normalization. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2017, pp. 1501–1510.
8. Hamilton, M.; Zhang, Z.; Hariharan, B.; Snavely, N.; Freeman, W.T. Unsupervised Semantic Segmentation by Distilling Feature Correspondences, 2022, [arXiv:cs.CV/2203.08414].
9. Isola, P.; Zhu, J.Y.; Zhou, T.; Efros, A.A. Image-to-image translation with conditional adversarial networks. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2017, pp. 1125–1134.
10. Mirza, M.; Osindero, S. Conditional Generative Adversarial Nets. *CoRR* **2014**, *abs/1411.1784*, [1411.1784].
11. Huang, X.; Liu, M.Y.; Belongie, S.; Kautz, J. Multimodal unsupervised image-to-image translation. In Proceedings of the Proceedings of the European conference on computer vision (ECCV), 2018, pp. 172–189.
12. Mao, Q.; Lee, H.Y.; Tseng, H.Y.; Ma, S.; Yang, M.H. Mode seeking generative adversarial networks for diverse image synthesis. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2019, pp. 1429–1437.
13. Choi, Y.; Choi, M.; Kim, M.; Ha, J.W.; Kim, S.; Choo, J. Stargan: Unified generative adversarial networks for multi-domain image-to-image translation. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 8789–8797.
14. Ho, J.; Jain, A.; Abbeel, P. Denoising diffusion probabilistic models. *Advances in neural information processing systems* **2020**, *33*, 6840–6851.
15. Zhang, L.; Rao, A.; Agrawala, M. Adding conditional control to text-to-image diffusion models. In Proceedings of the Proceedings of the IEEE/CVF International Conference on Computer Vision, 2023, pp. 3836–3847.
16. Li, B.; Xue, K.; Liu, B.; Lai, Y.K. Bbdm: Image-to-image translation with brownian bridge diffusion models. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern Recognition, 2023, pp. 1952–1961.
17. Saharia, C.; Chan, W.; Chang, H.; Lee, C.; Ho, J.; Salimans, T.; Fleet, D.; Norouzi, M. Palette: Image-to-image diffusion models. In Proceedings of the ACM SIGGRAPH 2022 conference proceedings, 2022, pp. 1–10.
18. Gatys, L.A.; Ecker, A.S.; Bethge, M. Image style transfer using convolutional neural networks. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 2414–2423.
19. Vaswani, A. Attention is all you need. *Advances in Neural Information Processing Systems* **2017**.
20. Deng, Y.; Tang, F.; Dong, W.; Ma, C.; Pan, X.; Wang, L.; Xu, C. Stytr2: Image style transfer with transformers. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2022, pp. 11326–11336.
21. Chung, J.; Hyun, S.; Heo, J.P. Style injection in diffusion: A training-free approach for adapting large-scale diffusion models for style transfer. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 8795–8805.
22. Deng, Y.; He, X.; Tang, F.; Dong, W. Z*: Zero-shot Style Transfer via Attention Reweighting. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2024, pp. 6934–6944.
23. Alami Mejjati, Y.; Richardt, C.; Tompkin, J.; Cosker, D.; Kim, K.I. Unsupervised attention-guided image-to-image translation. *Advances in neural information processing systems* **2018**, *31*.
24. Fontanini, T.; Botti, F.; Bertozzi, M.; Prati, A. Avoiding Shortcuts in Unpaired Image-to-Image Translation. In Proceedings of the International Conference on Image Analysis and Processing. Springer, 2022, pp. 463–475.
25. Zhu, P.; Abdal, R.; Qin, Y.; Wonka, P. Sean: Image synthesis with semantic region-adaptive normalization. In Proceedings of the Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 5104–5113.
26. He, K.; Gkioxari, G.; Dollár, P.; Girshick, R. Mask r-cnn. In Proceedings of the Proceedings of the IEEE international conference on computer vision, 2017, pp. 2961–2969.
27. Redmon, J.; Divvala, S.; Girshick, R.; Farhadi, A. You only look once: Unified, real-time object detection. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2016, pp. 779–788.
28. Ji, X.; Henriques, J.F.; Vedaldi, A. Invariant Information Distillation for Unsupervised Image Segmentation and Clustering. *CoRR* **2018**, *abs/1807.06653*, [1807.06653].

29. Cho, J.H.; Mall, U.; Bala, K.; Hariharan, B. Picie: Unsupervised semantic segmentation using invariance and equivariance in clustering. In Proceedings of the Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2021, pp. 16794–16804.
30. Fontanini, T.; Ferrari, C. Would your clothes look good on me? towards transferring clothing styles with adaptive instance normalization. *Sensors* **2022**, *22*, 5002.
31. Caron, M.; Touvron, H.; Misra, I.; Jégou, H.; Mairal, J.; Bojanowski, P.; Joulin, A. Emerging properties in self-supervised vision transformers. In Proceedings of the Proceedings of the IEEE/CVF international conference on computer vision, 2021, pp. 9650–9660.
32. Krähenbühl, P.; Koltun, V. Efficient inference in fully connected crfs with gaussian edge potentials. *Advances in neural information processing systems* **2011**, *24*.
33. Liu, Z.; Luo, P.; Wang, X.; Tang, X. Deep Learning Face Attributes in the Wild. In Proceedings of the Proceedings of International Conference on Computer Vision (ICCV), December 2015.
34. Caesar, H.; Uijlings, J.; Ferrari, V. Coco-stuff: Thing and stuff classes in context. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1209–1218.
35. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* **2014**.
36. Heusel, M.; Ramsauer, H.; Unterthiner, T.; Nessler, B.; Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *Advances in neural information processing systems* **2017**, *30*.
37. Zhang, R.; Isola, P.; Efros, A.A.; Shechtman, E.; Wang, O. The unreasonable effectiveness of deep features as a perceptual metric. In Proceedings of the Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 586–595.