Editing Out-of-domain GAN Inversion via Differential Activations

Haorui Song[1], Yong Du[2⋆], Tianyi Xiang[1], Junyu Dong[2], Jing Qin[3], and Shengfeng He[1][0000−0002−3802−4644]⋆⋆

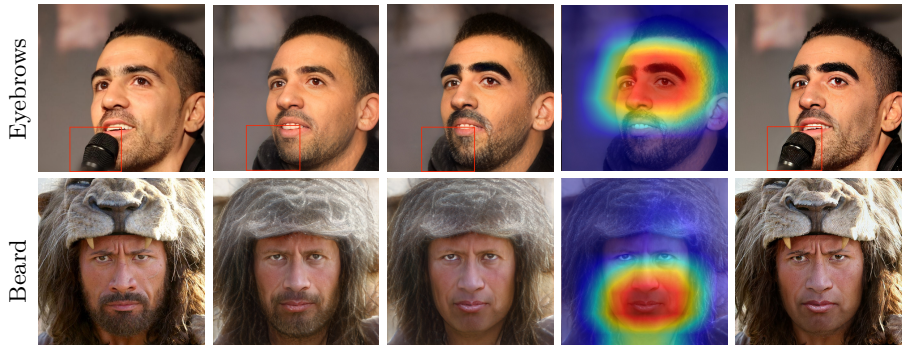
[1] South China University of Technology, Guangzhou, China
[2] Ocean University of China, Qingdao, China
[3] The Hong Kong Polytechnic University, Hong Kong SAR, China

Abstract. Despite the demonstrated editing capacity in the latent space of a pretrained GAN model, inverting real-world images is stuck in a dilemma that the reconstruction cannot be faithful to the original input. The main reason for this is that the distributions between training and real-world data are misaligned, and because of that, it is unstable of GAN inversion for real image editing. In this paper, we propose a novel GAN prior based editing framework to tackle the out-of-domain inversion problem with a composition-decomposition paradigm. In particular, during the phase of composition, we introduce a differential activation module for detecting semantic changes from a global perspective, *i.e.*, the relative gap between the features of edited and unedited images. With the aid of the generated Diff-CAM mask, a coarse reconstruction can intuitively be composited by the paired original and edited images. In this way, the attribute-irrelevant regions can be survived in almost whole, while the quality of such an intermediate result is still limited by an unavoidable ghosting effect. Consequently, in the decomposition phase, we further present a GAN prior based deghosting network for separating the final fine edited image from the coarse reconstruction. Extensive experiments exhibit superiorities over the state-of-the-art methods, in terms of qualitative and quantitative evaluations. The robustness and flexibility of our method is also validated on both scenarios of single attribute and multi-attribute manipulations. Code is available at https://github.com/HaoruiSong622/Editing-Out-of-Domain.

1 Introduction

Generative Adversarial Networks (GANs) [5, 12, 17] have demonstrated impressive image editing capability. From a random noise input, GAN models can encode abundant semantic information and spontaneously excavate interpretable directions in a latent space (*e.g.*, $\mathcal{W}$ space [18], $\mathcal{W}^+$ space [19] and etc.). By varying the latent codes along the controllable directions, highly realistic images with diverse attributes can be synthesized using GANs. However, such manipulations

⋆ The first two authors contribute equally.
⋆⋆ Corresponding author (hesfe@scut.edu.cn).



(a) Original Image (b) GAN Inversion (c) Latent Code Editing (d) Diff-CAM Mask (e) Our Result

Fig. 1: We delve deep into the editing problem of out-of-domain GAN inversion. (b) shows that for out-of-domain real images, GAN inversion cannot obtain a faithful reconstruction and therefore produce unacceptable editing (c). Our framework localizes semantic changes with differential activations (d), enabling the preservation of out-of-domain image content (like the lion hat and microphone) while activating the editing ability of GAN priors.

are applicable only in the latent space. For real images, a mapping function is required to transform the RGB input to a latent code.

GAN inversion [28, 43] which aims at inverting a given image back into the latent space of a pretrained GAN model such as StyleGAN [18], can enable the corresponding semantic directions to be applicable for real image editing. As a consequence, numerous GAN inversion based image processing frameworks [2, 13, 34, 38, 40, 42] have emerged. However, existing inversion methods are stuck in a dilemma that they cannot faithfully invert those images that are not from the distribution of training data. For example, as shown in Fig. 1b, both real images are fed into the pSp encoder [28] for inversion, and then the codes are sent to a pretrained StyleGAN2 [19] for generation, but it turns out that the microphone and the lion hat are vanished or distorted. This is due to the misaligned data domains. Such an out-of-domain issue can undoubtedly lead to unstable editing performance and thus severely hinder the practicality of GAN inversion. On the other hand, the powerful attribute-aware manipulation capability of pretrained GANs is indispensable for image editing. These facts motivate us to raise a natural idea: it would be feasible if we can properly integrate the edited region from the corresponding inversion with its unedited counterpart from the original input.

To achieve this goal, we turn to consider how to detect the edited region in the inversion. This reminds us of class activation mapping (CAM) [29, 41]. Such techniques focus on producing an attention map that highlights the regions that contribute to the classification decision, and have been widely used in visual explanations, such as weakly-supervised localization [6] and visual question answering [26]. Also for image manipulation, Kim *et al.* [20] utilize Grad-CAM [29]

(a) CAM [41] (b) Grad-CAM [29] (c) Ours

Fig. 2: Activation maps generated by CAM, Grad-CAM and our Diff-CAM models. Our activation map has a broader and more comprehensive coverage.

to generate a mask for localizing the attribute-relevant regions. A critical problem of CAM-based methods is that, its principle is to locate the activation regions that make the final decision, but making such a decision does not require a comprehensive activation on the attribute-relevant regions (*e.g.*, locating the wrinkle instead of the whole face can classify the "old" attribute). As a result, they tend to produce localized activations (see Fig. 2a and 2b). Relying on CAM for editing is apparently not flexible, as the editing of some attributes like "sex" may change the entire face, but binarily classifying male or female typically concentrate on facial components. This contradicts with our objective to combine edited region and its unedited counterpart.

In this paper, we propose a novel GAN prior based editing framework to resolve the above problems. Specifically, our editing method is executed in a composition-decomposition manner. In the composition stage, our aim is to generate a coarse reconstruction via combining the edited inversion with the original input, weighting by a Diff-CAM mask which is used for indicating the edited region. In particular, we present a simple yet effective *differential activation* mechanism to track the semantic changes rather than locating classification-relevant regions. It is performed by capturing the variational features between the edited and the original inversions, and we shed light on the differential features that reveal the editing attributes. In this way, the produced mask can specify the range of the edited region more accurately (see in Fig. 2c), as the semantical differences are explicitly embedded in the hidden responses. While in the decomposition stage, we need to remove the ghosting effect occurred in the coarse result. To deal with it, we further design a deghosting network that reuses the GAN prior, which is used for separating the final fine edited inversion from the coarse reconstruction in a multi-scale aggregation manner. Extensive experiments show that our method is the first feasible real-image editing method that built upon GAN inversion.

In summary, our key contributions are as follows:

- We delve deep into the out-of-domain problem existed in GAN inversion, and propose a novel GAN prior based editing framework in a composition-decomposition manner. Our method can use the original input to generate the unedited region, as well as maintaining a high quality of editing.

- We tailor a differential activation strategy to track semantic changes before and after editing. This design allows to embed more accurate range of the edited region with neglectable additional computational cost.
- We present a deghosting network with hierarchical GAN priors, for effectively alleviating the ghosting effect in the coarse reconstruction.
- We outperform state-of-the-art methods in terms of qualitative and quantitative evaluations, and we demonstrate the flexibility and robustness in both scenarios of single attribute and multi-attribute manipulations.

2 Related Work

Non-GAN prior based image manipulation. Non-GAN prior based methods [17,21,45] usually manipulate attributes of images via an adversarial training process. Kim *et al.* [21] propose a GAN based framework to discover cross-domain relations. Isola *et al.* [17] propose to use conditional GANs for image-to-image translation. And Zhu *et al.* [45] propose to translate images across different domains without paired training data. In general, these methods are designed to learn a model that corresponds to a specific translation, which leads to inflexibility in practical applications. To address this problem, StarGAN [9] is proposed to learn the mapping among multiple domains, using only a single generator and a discriminator. CMP [20] proposes to refine image-to-image translation results by introducing a cam-consistency loss to force the network to focus on attribute-relevant regions. Note that all these methods need to train models from scratch, and thus cannot capture GAN priors which is proven to be extremely effective for image manipulation [18,19]. Also, they are limited in synthesizing images at high resolution.

GAN prior based real image editing. GAN prior based methods, *i.e.*, GAN inversion, are proposed to inference a latent code of a given image based on a pretrained GAN model such as StyleGAN [18,19]. These methods can be roughly divided into two categories, optimization-based [1,2,10,13,27,35] and learning-based [7,8,13,14,32,37,44]. The main advantage of the former techniques is that they can ensure superior image reconstruction, while the corresponding cost is a higher computational complexity. In contrast, learning-based methods have a fast inference speed. Richardson *et al.* [28] propose a pSp encoder that can embed real images into an extended $\mathcal{W}^+$ space. Xu *et al.* [36] propose to train a hierarchical encoder based on a feature pyramid network. Alaluf *et al.* [4] introduce an iterative refinement mechanism for learning the inversion of real images. However, these methods cannot faithfully reconstruct the image content, mainly due to the misalignment between training and test data. We aim to solve this problem in a novel composition-decomposition paradigm via differential activations.

Interpreting CNN. Recent interpreting CNN models [11,29,41] attempt to understand the behaviour of the networks. Zhou *et al.* [41] propose CAM which aims to highlight the model's attention regarding a specific class. Selvaraju *et al.* [29] propose Grad-CAM without relying on the global average pooling layer. Lee *et al.* [22] propose LFI-CAM which treats the feature maps as masks and

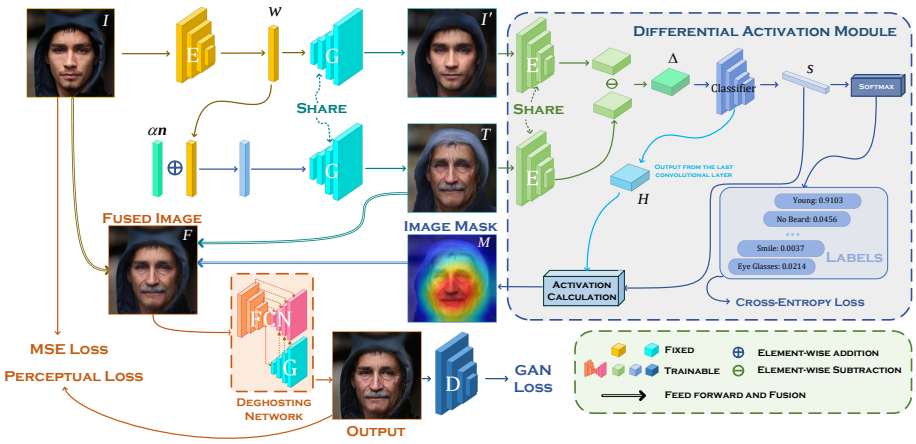


Fig. 3: Overall pipeline of our model. Given an input image, we first invert it to the latent space and perform user-desired editing. Then a differential activation module is applied to track the semantic changes of the manipulation. Edited region and unedited one are further combined using a Diff-CAM mask to produce a coarse reconstruction, on which the ghosting artifacts can be mitigated by the deghosting network with the aid of GAN priors.

learns the feature importance for generating the attention maps. Note that these methods are usually performed on responses themselves, while our strategy, with a clear aim of real image editing, explores to capture the variation between edit and unedited image features.

3 Approach

3.1 Overview

Due to the misaligned distributions between training and test data, existing GAN inversion methods cannot guarantee the fidelity of the reconstruction. And the quality of the subsequent edited image is therefore severely limited by such an out-of-domain problem. To remedy this, we propose a composition-decomposition paradigm for image editing and illustrate its overall pipeline in Fig. 3.

Specifically, given an image as input, our method firstly inverts it into the latent space. Semantic manipulation can then be produced by feeding and varying the latent code into a pretrained and fixed generator. Consequently, an initial result can be obtained by fusing the original input and the edited inversion with a Diff-CAM mask as weight. In particular, the procedure of generating the Diff-CAM mask is encapsulated in a self-contained differential activation module, which exploits the differential information between two reconstructions to promote the accuracy for determining the range of the editing-relevant region. The final output is further generated by a deghosting network, which resorts to the

diverse facial prior for mitigating the ghosting effect and enhancing the realism of the initial result. Note that we use the StyleGAN2 generator as the pretrained one in our model.

3.2 GAN Inversion and Single Attribute Editing

To achieve GAN inversion of a given image $\boldsymbol{I}$, we need to map it into a latent space in which rich semantic information is embedded. This can be implemented via many existing methods, for example, a pretrained pSp encoder $E_{\text{fixed}}(\cdot)$, and the latent code $\boldsymbol{w}$ can thus be formulated by $\boldsymbol{w} = E_{\text{fixed}}(I)$. Then we can obtain the inverted image $\boldsymbol{I}'$ by a pretrained StyleGAN2 generator $G(\cdot)$, which is formulated by $\boldsymbol{I}' = G(\boldsymbol{w})$. Note that there most likely exists a bias between $\boldsymbol{I}'$ and $\boldsymbol{I}$, due to the out-of-domain problem.

And to manipulate the corresponding attributes, the latent code would be varied along various intepretable directions that are discovered in the latent space. The edited inversion $\boldsymbol{T}$ can then be produced based on the altered code by the same generator $G(\cdot)$. Given a specific direction $\boldsymbol{n}$ for single attribute editing, this process can be formulated as $\boldsymbol{T} = G(\boldsymbol{w} + \alpha\boldsymbol{n})$, where α is a scaling factor. Note that our method can also support multi-attribute editing, and we will discuss this in Sec. 3.6.

3.3 Differential Activation Module

Once we have the paired images $\{\boldsymbol{I}', \boldsymbol{T}\}$, we respectively feed them into a plain trainable encoder $E_{\text{trainable}}(\cdot)$, and can easily obtain the differential features $\boldsymbol{\Delta}$ via a simple subtraction operation:

$$\boldsymbol{\Delta} = E_{\text{trainable}}(\boldsymbol{I}') - E_{\text{trainable}}(\boldsymbol{T}). \tag{1}$$

Subsequently, these features are sent to a lightweight network which serves as a classifier and consists of convolutional and fully connected layers. We use the cross-entropy loss $\mathcal{L}_{\text{ce}}$ to train the encoder and the classifier together, which can be formulated as follows:

$$\mathcal{L}_{\text{ce}} = -\sum_{c=1}^{N} y_c log \frac{e^{s_c}}{\sum_{i=1}^{N} e^{s_i}}, \tag{2}$$

where $\boldsymbol{y} = \{y_1, y_2, \cdots, y_N\}$ is a one-hot vector that indicates which attribute has been edited, $\boldsymbol{s} = \{s_1, s_2, \cdots, s_N\}$ denotes the output vector of the classifier before softmax operation, e denotes the natural constant, and N is the total number of attributes.

Now we are ready for performing activation calculation. The first step is to define the weight β_c^k that corresponded to the kth channel of $\boldsymbol{H}$ and the cth attribute, which is formulated as follows:

$$\beta_c^k = \overbrace{\frac{1}{Z}\sum_i \sum_j}^{\text{global average pooling}} \frac{\partial s_c}{\partial \boldsymbol{H}_{ij}^k}, \tag{3}$$

where $\boldsymbol{H}$ is the features generated by the last convolutional layer in the classifier, i and j respectively denotes the height and the width of the features.

Then our Diff-CAM mask $\boldsymbol{M}_{\text{Diff-CAM}}$ can be represented as a piecewise linear transformation of weighted differential features, that is

$$\boldsymbol{M}_{\text{Diff-CAM}} = \text{ReLU}(\sum_{k} \beta_c^k \boldsymbol{H}^k). \tag{4}$$

Finally, we normalize the above mask into the interval of $[0, 1]$ via $\boldsymbol{M}_{\text{Diff-CAM}} = \boldsymbol{M}_{\text{Diff-CAM}}/\max(\boldsymbol{M}_{\text{Diff-CAM}})$. Since the Diff-CAM mask is generated based on the differential features that describe semantically changes, the range of the editing-relevant region can be detected more accurately via a comprehensive activation. It is thus more suitable than other CAM-based masks for image editing.

3.4 Composition

After obtaining the Diff-CAM mask, it is time to composite the edited image with the original input for resolving the out-of-domain issue. We have the fused image $\boldsymbol{F}_{\text{fused}}$ by the following weighted average formula:

$$\boldsymbol{F}_{\text{fused}} = \boldsymbol{T} \odot \boldsymbol{M}_{\text{Diff-CAM}} + \boldsymbol{I} \odot (1 - \boldsymbol{M}_{\text{Diff-CAM}}), \tag{5}$$

where $\odot$ denotes the hadamard product. However, the quality of such an initial blending result is unsatisfactory due to an inevitable ghosting effect.

3.5 Deghosting Network

To cope with ghosting artifacts, we treat the coarse reconstruction $\boldsymbol{F}_{\text{fused}}$ as a combination of a target image and a ghost image. In order to decompose the target image out, we further perform a deghosting process on the coarse result via a deghosting network. As shown in Fig. 3, the architecture of the network includes a fully convolutional network which consists of an encoder (the orange part), a decoder (the pink part), a pretrained StyleGAN2 generator, and a discriminator $D(\cdot)$. Note that we denote the aggregation of the first three modules as $\phi(\cdot)$.

Our goal is to utilize the ghosting-free nature of the inherent facial prior in the pretrained GAN model, such that ghosting artifacts can be removed without destroying the original facial details. In particular, we first feed $\boldsymbol{F}_{\text{fused}}$ to an FCN-like [25] encoder-decoder architecture for two purposes: the encoder generates the latent code of $\boldsymbol{F}_{\text{fused}}$, and the decoder is trained to produce ghosting-free results. Meanwhile, with the predicted latent code, $\boldsymbol{F}_{\text{fused}}$ is inverted in the latent space and reconstructed by the StyleGAN2 generator without ghosting artifacts. We aggregate the corresponding features of the generator with the decoder hierarchically, yielding the final deghosting result.

Since the fused image $\boldsymbol{F}_{\text{fused}}$ has no ground-truth counterpart, we synthesize a set of paired data $\{\boldsymbol{F}_{\text{train}}, \boldsymbol{I}\}$ to train the deghosting network. The training image $\boldsymbol{F}_{\text{train}}$ is given by

$$\boldsymbol{F}_{\text{train}} = \boldsymbol{T} \odot \boldsymbol{M}_{\text{train}} + \boldsymbol{I} \odot (1 - \boldsymbol{M}_{\text{train}}), \tag{6}$$

and the corresponding Diff-CAM mask $\boldsymbol{M}_{\text{train}}$ is defined as follows:

$$\boldsymbol{M}_{\text{train}}(i,j) = \begin{cases} \boldsymbol{M}_{\text{Diff-CAM}}(i,j), & \text{if} \quad \boldsymbol{M}_{\text{Diff-CAM}}(i,j) \leq 0.5, \\ 1 - \boldsymbol{M}_{\text{Diff-CAM}}(i,j), & \text{if} \quad \boldsymbol{M}_{\text{Diff-CAM}}(i,j) > 0.5. \end{cases} \tag{7}$$

The rationale behind this setting is that: 1) the mask $\boldsymbol{M}_{\text{train}}$ is thus regularized into an interval of $[0, 0.5]$ so that the content of $\boldsymbol{I}$ dominates that of $\boldsymbol{F}_{\text{train}}$. We can then treat image $\boldsymbol{I}$ as the required ground truth. And 2) meanwhile, the ghosting effect still exists in $\boldsymbol{F}_{\text{train}}$. Note that the corresponding attribute in regard to generate the mask $\boldsymbol{M}_{\text{train}}$ is consistent with $\boldsymbol{T}$ and randomly selected. The total objective $\mathcal{L}_{\text{deghost}}$ for optimizing the deghosting network is defined as follows:

$$\mathcal{L}_{\text{deghost}} = \lambda_m \mathcal{L}_{\text{mse}} + \lambda_p \mathcal{L}_{\text{percep}} + \lambda_a \mathcal{L}_{\text{adv}}, \tag{8}$$

where $\mathcal{L}_{\text{mse}}$, $\mathcal{L}_{\text{percep}}$ and $\mathcal{L}_{\text{adv}}$ respectively denotes MSE loss, perceptual loss and adversarial loss, λ_m, λ_p, λ_a are the balance factors. And the involved three losses are respectively defined as follows:

$$\mathcal{L}_{\text{mse}} = \frac{1}{Q} \|\boldsymbol{I} - \phi(\boldsymbol{F}_{\text{train}})\|_2, \tag{9}$$

$$\mathcal{L}_{\text{percep}} = \frac{1}{Q} \|V(\boldsymbol{I}) - V(\phi(\boldsymbol{F}_{\text{train}}))\|_2, \tag{10}$$

$$\mathcal{L}_{\text{adv}} = \mathop{\mathbb{E}}_{\boldsymbol{I} \sim P_r} log(D(\boldsymbol{I})) + \mathop{\mathbb{E}}_{\boldsymbol{F}_{\text{train}} \sim P_g} log(1 - D(\phi(\boldsymbol{F}_{\text{train}}))), \tag{11}$$

where Q indicates the number of pixels, P_r and P_g respectively denotes the distribution of real data and generated data, $V(\cdot)$ denotes a pretrained VGG-16 network, and we select the features produced by the conv4_3 layer for modeling the loss.

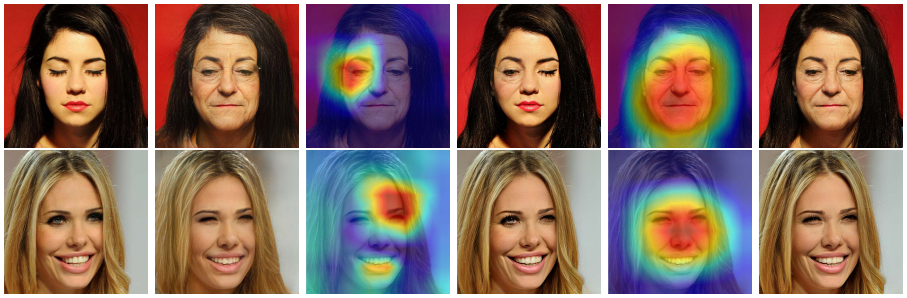
3.6 Multi-attribute editing

Our method also has a flexibility to handle multi-attribute editing. In fact, it can be decomposed into a sequence of single attribute editing tasks. Suppose the number of the attributes needed to be edited is r, three special points should be noted: 1) In the ith ($i \neq 1$) single attribute editing, the paired images $\{\boldsymbol{T}_i, \boldsymbol{T}_{i-1}\}$ are used for calculating the Diff-CAM mask. At last we will have a set of r masks. 2) The final Diff-CAM mask is the result of performing element-wise maximization operation on the mask set. And 3) The final fused image is the composition of $\boldsymbol{T}_r$ and $\boldsymbol{I}$ with the final mask as weight.

4 Experiments

4.1 Implementation Details

We implement our method in Pytorch on a PC with an Nvidia GeForce RTX 3090. During training, we use Adam as the optimizer with a learning rate of 0.0001, $\beta_1 = 0.9$, $\beta_2 = 0.99$. The hyperparameters in Eq. (8) are empirically



(a) Input | (b) "Age" & "Eyes open" | (c) Grad-CAM [29] | (d) Editing based on (c) | (e) Our Diff-CAM | (f) Editing based on (e)

Fig. 4: Comparison with Grad-CAM in our editing framework. Grad-CAM exhibits localized attentions, while ours can correctly locate semantic changes during editing.

set to $\lambda_m = 1$, $\lambda_p = 0.8$ and $\lambda_a = 0.01$. Before being sent to E_{train}, $\boldsymbol{I}'$ and $\boldsymbol{T}$ are downsampled from a resolution of 1024×1024 to that of 256×256. And the Diff-CAM mask computed by Eq. (4) will be upsampled to a resolution of 1024×1024 before being used in the composition process.

4.2 Experimental Data

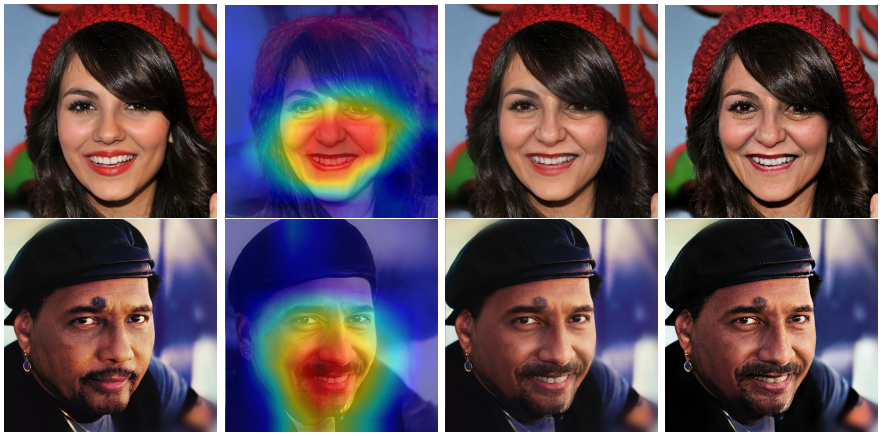
FFHQ dataset [18] and Celeba-HQ dataset [24] both contain human face images of high quality and resolution, with 70000 and 30000 images respectively. We employ the FFHQ dataset for training the differential activation module and the deghosting network, while we utilize the Celeba-HQ dataset for testing. All the quantitative metrics are calculated on the Celeba-HQ dataset.

4.3 Component Analysis

Effectiveness of DA module. First, in order to prove the effectiveness of our design of the DA module structure, we replace the DA module with the commonly used Grad-CAM [29] and check out how the masks differ and influence the editing.

The results are shown in Fig. 4. The results show the limitation of Grad-CAM that activates only in local areas. The resulted masks cannot suit for discovering semantic differences and therefore not suit for blending GAN inversion with the source image. On the contrary, the masks generated by our DA module successfully cover the editing-relevant regions, producing a global coverage for "age" attribute while a local attention for "eyes open" attribute. This largely aids the blending of edited and unedited regions.

Effectiveness of Deghosting Network. Here we show the results before and after deghosting in Fig. 5. Even with a correctly detected mask, blending two images inevitably produces blurry and ghosting artifacts. Our deghosting network takes advantage of the rich facial priors from the pretrained GAN model, and effectively removes non-face artifacts as well as generating a clear blending of faces.



(a) Input (b) Editing & Mask (c) Composition (d) Deghosting

Fig. 5: Effect of our deghosting network. Directly blending two images with a mask inevitably presents ghosting artifacts (teeth and face shape, zoom in for better view). Our deghosting network utilizes GAN priors to faithfully remove artifacts while retaining facial details.

Flexibility Analysis. Our method is flexible and independent of the applied GAN inversion and interpretable directions. We choose several different inversion and interpretable direction models to work with our framework. Three combinations of inversion and direction methods are used, *i.e.*, the pSp encoder [28] together with directions found by StyleGAN2 distillation [33], the IdInvert encoder [43] with InterfaceGAN [30, 31], and the e4e encoder [32] together with the directions obtained by StyleFlow [3].

The results are shown in Fig. 6. From the result we can see that all the encoding methods fail to retain the out-of-domain information. As for the first person wearing a blue hat and holding a fist, all encoders treat the hat as the hair and the fist as the background. Similar problem exists in the second person, in which his cap is inverted to hair and becomes white as he gets older. All these out-of-domain problems are addressed by our framework, regardless of their inversions and applied editings. The results show that our DA module and our deghosting network are encoder-independent and are robust enough to work with different types of inversion and editing methods.

4.4 Comparison with SOTAs

In order to prove the superiority of our model, we compare our model with other state-of-the-art facial attributes editing methods. Note that we do not make an additional comparison with StyleGAN inversion based editing method other than Fig. 6. This is due to that they would obviously fail on out-of-domain samples as the original GAN was not trained on them. To maintain fairness, here we mainly compare our results with those non-StyleGAN based image-to-image translation methods. In particular, we compare our model with StarGAN [9],

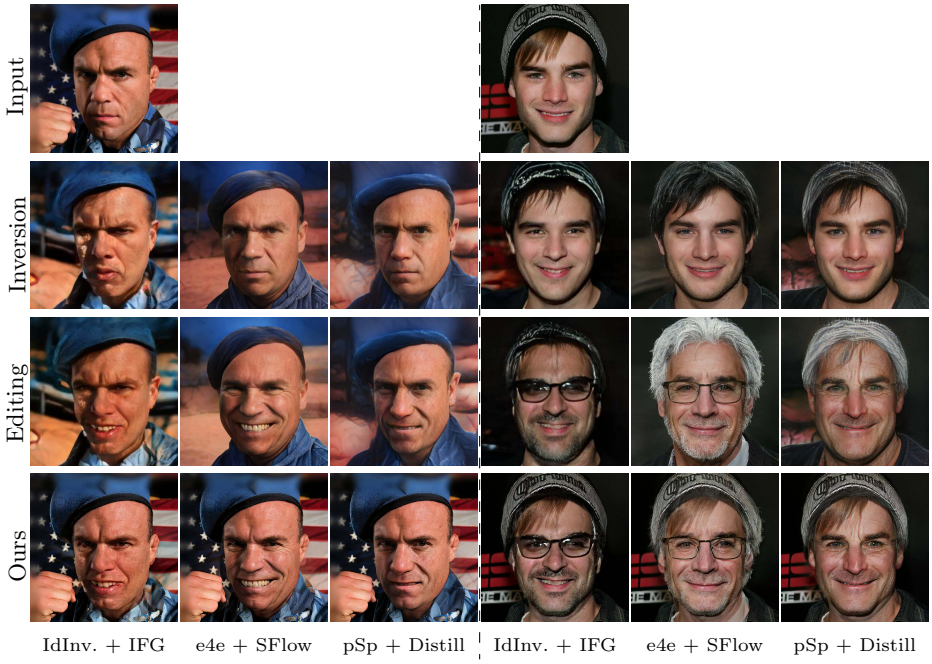


Fig. 6: Our framework is independent to GAN inversion and interpretable editing directions. We can work with arbitrary combinations of encoders and directions. (Zoom in for better view.)

AttGAN [15], and STGAN [23]. Kim *et al.* [20] propose to refine an image-to-image translation method by introducing a CAM-consistency loss to force the network to focus on attribute-relevant regions, and we also compare to the refined version of StarGAN and AttGAN (denoted as StarGAN* and AttGAN*). All the results are generated with their official codes, except the refinement method of Kim *et al.* [20] that is not publicly available and implemented by ourselves.

Quantitative Evaluation. To quantitatively compare our method with state-of-the-arts, we use the Fre´chet inception distance (FID) [16] and learned perceptual image patch similarity (LPIPS) [39] metrics to measure the quality of the results. FID measures the distribution distance between the original image dataset and the manipulated dataset. We calculate FID metric for each model, and select 4 common attributes ("age", "bushy eyebrows", "eyeglasses", and "beard") that can be modified by all of the models to generate the manipulated dataset. The final FID value of each model is obtained by averaging the FID values corresponding to each attribute. LPIPS metric measures the perceptual similarity between the two images. The smaller the value of LPIPS, the greater the similarity. We use it to evaluate non-edited region consistency.

The numerical results are shown in Table 1. As can be seen, the proposed method shows the best FID and LPIPS scores among the competitors. This reveals that our model can better maintain the distribution of the original dataset, and a strong capability to preserve the image quality for non-edited regions.

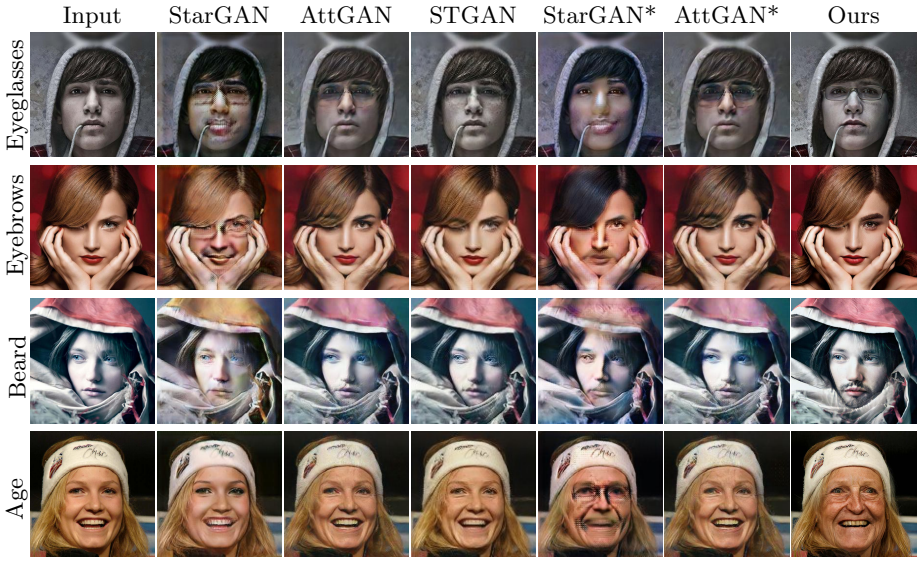


Fig. 7: Qualitative evaluation with respect to state-of-the-art image-to-image face editing methods on 4 attributes, "age", "bushy eyebrows", "eyeglasses", and "beard". StarGAN* and AttGAN* represent the refined version using the cam-consistency loss [20]. Our method can produce high-resolution and semantically correct editing on these challenging cases. (Zoom in for better view.)

Qualitative Evaluation. In order to demonstrate the superiority of our model, we conduct a qualitative study by contrasting the results generated from different models. Again, here we mainly compare with the image-to-image translation models. The results of editing 4 attributes by different models are shown in Fig. 7. We can see that, the outputs of our model are of the highest quality compared to all other methods. Our outputs best change the attributes while retaining other irrelevant information. StarGAN and its refined version suffer from checkerboard-like artifacts. AttGAN* can achieve better editing than the original version, but it still produces blurry details and semantically incorrect editing (like the eyebrow on the left wrongly appears on the hair in the second example), indicating that using an additional mask-guided loss is not reliable for challenging cases. In contrast, our Diff-CAM mask driven framework obtains significantly preferable editing performance, not to mention the high-resolution features provided by StyleGAN-based editing.

4.5 Editing on non-facial attributes and domains

We also verify the generalization ability of our model and display the qualitative results in Fig. 8. In particular, regarding non-facial attributes (like "hair color" or "hairstyle") and other domains (like "car"), the modifications may no longer happen in the center region like most of those occured in facial attributes editing, *e.g.*, logo, wheels, and hair. Nevertheless, our Diff-CAM can always precisely

	StarGAN	AttGAN	STGAN	StarGAN*	AttGAN*	Ours
FID↓	30.98	17.96	20.97	28.52	15.72	**13.76**
LPIPS↓	0.208	0.107	0.178	0.138	0.099	**0.094**

Table 1: Quantitative comparison with state-of-the-art face editing methods. StarGAN* and AttGAN* represent the refined version using the cam-consistency loss [20]. Our method achieves the best numerical performance.

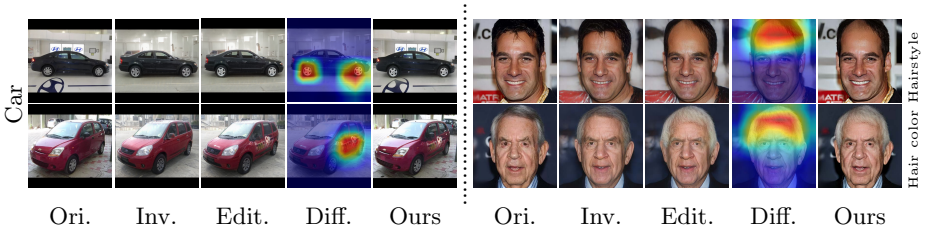


Fig. 8: Evaluation on "car" domain and non-facial attributes "hairstyle", "hair color". (Zoom in for better view.)

distinguish the edited and non-edited areas. And our framework can thus be robust to small regions and different attributes for editing.

4.6 Multi-attribute Editing

In addition to the single-attribute editing, our model shows its high degree of flexibility by also supporting editing multiple attributes. The process of modifying multiple attributes is completed by modifying the attributes one by one as is described in Sec. 3.6. Fig. 9 shows two examples of editing two attributes, "eyes open" and "smile". The final outputs of our model successfully introduce the changes involved in the two editing steps and also manage to maintain the uninvertible information such as the hats and the fingers.

4.7 Limitation

Although our model has achieved promising performance on the editing of facial or non-facial attributes and other domains, its ability to handle attribute changing is not unlimited. Fig. 10 shows the examples of our limitation. Our model is heavily relied on the performance of GAN inversion methods and only introduces changes covered by the mask from the DA module. Therefore, if the inverted result cannot faithfully reconstruct the original image which is likely occurred in the case of non-human domains, serious distortion and ghosting artifacts will be existed in the final result.

5 Conclusion

In this paper, we propose a novel GAN prior based editing technique to tackle the out-of-domain inversion problem with a composition-decomposition paradigm.

(a) Input (b) “Eyes open” (c) “Smile” (d) Ours

Fig. 9: Multi-attribute editing results. Our method can successfully edit multiple attributes one by one, while still retaining the out-of-domain regions.

Fig. 10: Challenging cases of our model. The two examples from different domains show the editing results when the inversion cannot faithfully reconstruct the original image.

We introduce a differential activation mechanism to track semantic changes before and after editing. With the aid of the calculated Diff-CAM mask, a coarse reconstruction can be obtained by the composition of the edited image and the original input. We further present a deghosting network to mitigate the ghosting effect in the coarse result. Both qualitative and quantitative evaluations validate the superiority of our method.

Acknowledgement

This project is supported by the National Natural Science Foundation of China (62102381, U1706218, 41927805, 61972162); Shandong Natural Science Foundation (ZR2021QF035); Fundamental Research Funds for the Central Universities (202113035); the National Key R&D Program of China (2018AAA0100600); the China Postdoctoral Science Foundation (2020M682240, 2021T140631); Guangdong International Science and Technology Cooperation Project (No. 2021A0505030009); Guangdong Natural Science Foundation (2021A1515012625); Guangzhou Basic and Applied Research Project (202102021074); and CCF-Tencent Open Research fund (RAGR20210114).

References

1. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan: How to embed images into the stylegan latent space? In: ICCV. pp. 4432–4441 (2019) 4
2. Abdal, R., Qin, Y., Wonka, P.: Image2stylegan++: How to edit the embedded images? In: CVPR. pp. 8296–8305 (2020) 2, 4
3. Abdal, R., Zhu, P., Mitra, N.J., Wonka, P.: Styleflow: Attribute-conditioned exploration of stylegan-generated images using conditional continuous normalizing flows. ACM TOG **40**(3), 1–21 (2021) 10
4. Alaluf, Y., Patashnik, O., Cohen-Or, D.: Restyle: A residual-based stylegan encoder via iterative refinement. In: ICCV. pp. 6711–6720 (2021) 4
5. Arjovsky, M., Chintala, S., Bottou, L.: Wasserstein generative adversarial networks. In: ICML. pp. 214–223. PMLR (2017) 1
6. Bae, W., Noh, J., Kim, G.: Rethinking class activation mapping for weakly supervised object localization. In: ECCV. pp. 618–634 (2020) 2
7. Bau, D., Zhu, J.Y., Wulff, J., Peebles, W., Strobelt, H., Zhou, B., Torralba, A.: Inverting layers of a large generator. In: ICLR. vol. 2, p. 4 (2019) 4
8. Chai, L., Wulff, J., Isola, P.: Using latent space regression to analyze and leverage compositionality in gans. arXiv preprint arXiv:2103.10426 (2021) 4
9. 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: CVPR. pp. 8789–8797 (2018) 4, 10
10. Collins, E., Bala, R., Price, B., Susstrunk, S.: Editing in style: Uncovering the local semantics of gans. In: CVPR. pp. 5771–5780 (2020) 4
11. Fukui, H., Hirakawa, T., Yamashita, T., Fujiyoshi, H.: Attention branch network: Learning of attention mechanism for visual explanation. In: CVPR. pp. 10705–10714 (2019) 4
12. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. Advances in neural information processing systems **27** (2014) 1
13. Gu, J., Shen, Y., Zhou, B.: Image processing using multi-code gan prior. In: CVPR. pp. 3012–3021 (2020) 2, 4
14. Guan, S., Tai, Y., Ni, B., Zhu, F., Huang, F., Yang, X.: Collaborative learning for faster stylegan embedding. arXiv preprint arXiv:2007.01758 (2020) 4
15. He, Z., Zuo, W., Kan, M., Shan, S., Chen, X.: Attgan: Facial attribute editing by only changing what you want. IEEE TIP **28**(11), 5464–5478 (2019) 11
16. 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. In: NeurIPS (2017) 11
17. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: CVPR. pp. 1125–1134 (2017) 1, 4
18. Karras, T., Laine, S., Aila, T.: A style-based generator architecture for generative adversarial networks. In: CVPR. pp. 4401–4410 (2019) 1, 2, 4, 9
19. Karras, T., Laine, S., Aittala, M., Hellsten, J., Lehtinen, J., Aila, T.: Analyzing and improving the image quality of stylegan. In: CVPR. pp. 8110–8119 (2020) 1, 2, 4
20. Kim, D., Khan, M.A., Choo, J.: Not just compete, but collaborate: Local image-to-image translation via cooperative mask prediction. In: CVPR. pp. 6509–6518 (2021) 2, 4, 11, 12, 13

21. Kim, T., Cha, M., Kim, H., Lee, J.K., Kim, J.: Learning to discover cross-domain relations with generative adversarial networks. In: ICML. pp. 1857–1865. PMLR (2017) 4
22. Lee, K.H., Park, C., Oh, J., Kwak, N.: Lfi-cam: Learning feature importance for better visual explanation. In: ICCV. pp. 1355–1363 (2021) 4
23. Liu, M., Ding, Y., Xia, M., Liu, X., Ding, E., Zuo, W., Wen, S.: Stgan: A unified selective transfer network for arbitrary image attribute editing. In: CVPR. pp. 3673–3682 (2019) 11
24. Liu, Z., Luo, P., Wang, X., Tang, X.: Deep learning face attributes in the wild. In: ICCV. pp. 3730–3738 (2015) 9
25. Long, J., Shelhamer, E., Darrell, T.: Fully convolutional networks for semantic segmentation. In: CVPR. pp. 3431–3440 (2015) 7
26. Patro, B.N., Lunayach, M., Patel, S., Namboodiri, V.P.: U-cam: Visual explanation using uncertainty based class activation maps. In: ICCV. pp. 7444–7453 (2019) 2
27. Raj, A., Li, Y., Bresler, Y.: Gan-based projector for faster recovery with convergence guarantees in linear inverse problems. In: ICCV. pp. 5602–5611 (2019) 4
28. Richardson, E., Alaluf, Y., Patashnik, O., Nitzan, Y., Azar, Y., Shapiro, S., Cohen-Or, D.: Encoding in style: a stylegan encoder for image-to-image translation. In: CVPR (2021) 2, 4, 10
29. Selvaraju, R.R., Cogswell, M., Das, A., Vedantam, R., Parikh, D., Batra, D.: Grad-cam: Visual explanations from deep networks via gradient-based localization. In: ICCV. pp. 618–626 (2017) 2, 3, 4, 9
30. Shen, Y., Gu, J., Tang, X., Zhou, B.: Interpreting the latent space of gans for semantic face editing. In: CVPR. pp. 9243–9252 (2020) 10
31. Shen, Y., Yang, C., Tang, X., Zhou, B.: Interfacegan: Interpreting the disentangled face representation learned by gans. IEEE TPAMI (2020) 10
32. Tov, O., Alaluf, Y., Nitzan, Y., Patashnik, O., Cohen-Or, D.: Designing an encoder for stylegan image manipulation. ACM TOG **40**(4), 1–14 (2021) 4, 10
33. Viazovetskyi, Y., Ivashkin, V., Kashin, E.: Stylegan2 distillation for feed-forward image manipulation. In: ECCV. pp. 170–186. Springer (2020) 10
34. Xu, Y., Deng, B., Wang, J., Jing, Y., Pan, J., He, S.: High-resolution face swapping via latent semantics disentanglement. In: CVPR. pp. 7642–7651 (2022) 2
35. Xu, Y., Du, Y., Xiao, W., Xu, X., He, S.: From continuity to editability: Inverting gans with consecutive images. In: ICCV. pp. 13910–13918 (2021) 4
36. Xu, Y., Shen, Y., Zhu, J., Yang, C., Zhou, B.: Generative hierarchical features from synthesizing images. In: CVPR. pp. 4432–4442 (2021) 4
37. Yang, H., Chai, L., Wen, Q., Zhao, S., Sun, Z., He, S.: Discovering interpretable latent space directions of gans beyond binary attributes. In: CVPR. pp. 12177–12185 (2021) 4
38. Yang, T., Ren, P., Xie, X., Zhang, L.: Gan prior embedded network for blind face restoration in the wild. In: CVPR. pp. 672–681 (2021) 2
39. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR. pp. 586–595 (2018) 11
40. Zhong, Z., Chai, L., Zhou, Y., Deng, B., Pan, J., He, S.: Faithful extreme rescaling via generative prior reciprocated invertible representations. In: CVPR. pp. 5708–5717 (2022) 2
41. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: CVPR. pp. 2921–2929 (2016) 2, 3, 4

42. Zhou, Y., Xu, Y., Du, Y., Wen, Q., He, S.: Pro-pulse: Learning progressive encoders of latent semantics in gans for photo upsampling. IEEE TIP **31**, 1230–1242 (2022) 2
43. Zhu, J., Shen, Y., Zhao, D., Zhou, B.: In-domain gan inversion for real image editing. In: ECCV. pp. 592–608 (2020) 2, 10
44. Zhu, J.Y., Krähenbühl, P., Shechtman, E., Efros, A.A.: Generative visual manipulation on the natural image manifold. In: ECCV. pp. 597–613 (2016) 4
45. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: ICCV. pp. 2223–2232 (2017) 4