What Decreases Editing Capability? Domain-Specific Hybrid Refinement for Improved GAN Inversion

Pu Cao[1,2] Lu Yang[1] Dongxv Liu[1] Xiaoya Yang[1] Tianrui Huang[1] Qing Song[1*]
[1]Beijing University of Posts and Telecommunications [2]Metavatar
{caopu, soeaver, liudongxv, yangxiaoya, huangtianrui, priv}@bupt.edu.cn

Abstract

Recently, inversion methods have been exploring the incorporation of additional high-rate information from pretrained generators (such as weights or intermediate features) to improve the refinement of inversion and editing results from embedded latent codes. While such techniques have shown reasonable improvements in reconstruction, they often lead to a decrease in editing capability, especially when dealing with complex images that contain occlusions, detailed backgrounds, and artifacts. To address this problem, we propose a novel refinement mechanism called ***Domain-Specific Hybrid Refinement*** *(DHR), which draws on the advantages and disadvantages of two mainstream refinement techniques. We find that the weight modulation can gain favorable editing results but is vulnerable to these complex image areas and feature modulation is efficient at reconstructing. Hence, we divide the image into two domains and process them with these two methods separately. We first propose a Domain-Specific Segmentation module to automatically segment images into in-domain and out-of-domain parts according to their invertibility and editability without additional data annotation, where our hybrid refinement process aims to maintain the editing capability for in-domain areas and improve fidelity for both of them. We achieve this through Hybrid Modulation Refinement, which respectively refines these two domains by weight modulation and feature modulation. Our proposed method is compatible with all latent code embedding methods. Extension experiments demonstrate that our approach achieves state-of-the-art in real image inversion and editing. Code is available at* https://github.com/caopulan/Domain-Specific_Hybrid_Refinement_Inversion.

Figure 1. **Inversion and editing results of our method.** We preserve image details, including background and occlusion, in both inversion and manipulation processes.

1. Introduction

Generative Adversarial Networks (GANs) have shown promising results in image generation. Synthetic images are photorealistic with high resolution and are difficult to distinguish from real images [16,18–20,42]. Based on their highly semantic latent space, image manipulation and controllable generation are deeply explored. Moreover, GANs can represent a high-quality image prior to improving various tasks, such as face parsing [37–41, 43], style transfer [21, 44], and face super-resolution [35]. However, real images are inapplicable for these applications, since most of them require latent codes in GANs' feature space.

Inversion is built to convert real images into GANs' la-

*corresponding author.

tent space. The inverted latent codes are required to reconstruct given images by a pretrained generator, which also embeds semantic information to edit or apply in GAN-based tasks. Two types of methods generally reach image embedding. One is training an image encoder to convert given images to latent codes [25, 31], while another is minimizing the discrepancy between given images and reconstructed images to optimize initial latent codes iteratively [20]. This process attains the corresponding latent codes to reconstruct or edit the images. However, latent codes are low bit-rate [34], and high-rate details of images may not be reconstructed faithfully. Hence, many recent works focus on refining results by modulating additional high-rate information, which can be mainly divided into weight modulation and feature modulation. Weight modulation improves fidelity by iteratively tuning generator's weight by minimizing reconstruction error [27] or predicting weight offsets [5] by Hypernetworks [12]. Feature modulation encodes spatial details into intermediate features in generator, where the encoding process is also reached by iterative optimization [23] or model prediction [23, 34].

As reconstruction performance increases by refinement with high-rate information, editing capability is inevitably decreased, especially on images containing complex parts, which we show in Figure 3. This phenomenon is due to the destruction of the generation ability of pretrained GANs. Supervised by a discriminator, GANs are constrained to generate realistic images from sampled latent codes. However, the additional high-rate information degrades the highly semantic latent space of pretrained generator, making the latent manifold sharp by weight modulation [9] or fixing the spatial distribution by intermediate features modulation to overfit the given images to reconstruct. Particularly, high-rate information needs drastic change to reconstruct complex parts. Notably, complex images prevail in the natural world. For example, face accessories, hats, occlusions, and complex backgrounds usually appear in face photos. Hence, our goal is to design a robust refinement inversion method that can faithfully reconstruct given images and retain editing capability.

Based on the above illustration, we explore the idea of "divide and conquer" to address this problem. Specifically, we divide the image into *in-domain* and *out-of-domain* parts. *In-domain* parts imply areas close to generators' output distribution and are desired to perform well on both inversion and editing. Correspondingly, *out-of-domain* parts are areas challenging to inverse or edit and desired to reconstruct faithfully. Hence, we introduce a hybrid refinement method to handle them. We refine *in-domain* parts by tuning generator weight since it is better to maintain editing capability. For *out-of-domain* parts, we straightforwardly invert them by intermediate features to keep spatial image details. Notably, our hybrid refinement method is the first work to analyze and combine feature and weight modulation for improved GAN inversion and achieves extraordinary results as shown in Figure 1.

Extensive experiments are presented to demonstrate the effects of our **D**omain-Specific **H**ybrid **R**efinement (DHR). We achieve state-of-the-art and gain significant improvement in both fidelity and editability. The key contributions of this work are summarized as follows:

- We analyze the reasons for editing capability degradation in the refinement process. Based on our analysis, we introduce *in-domain* and *out-of-domain* and propose Domain-Specific Segmentation to segment images into these two parts for better inversion.
- We propose Hybrid Modulation Refinement to improve inversion results of *in-domain* and *out-of-domain* parts. We conduct weight modulation on *in-domain* part and feature modulation on *out-of-domain* part, which can preserve editing capability when refining the image details.
- We conduct extensive experiments and user studies to demonstrate the effects of our method. We reach extraordinary performance on real-world image inversion and editing and achieve the state-of-the-art.

2. Related Work

2.1. GAN Inversion

GAN inversion aims to embed real-world images into a pretrained generator's latent space, which can be used to reconstruct and edit input images. Generally, methods can be divided into two stages.

The first stage aims to attain low-rate latent codes, usually in $Z/W/W^+$ spaces. The latent codes are gained by an encoder or optimization process. Training an encoder [6, 11, 25, 26, 31, 36] to predict latent codes is efficient for inference and is easier to get better trade-offs between fidelity and manipulation [25, 31]. Optimizing initial latent codes by reconstruction discrepancy gains better fidelity. However, it may cost several minutes per image [1, 2, 6, 20] and decreases editability during per-image tuning. Due to low-rate property, latent codes can only reconstruct coarse information and drop the details from original images. Meanwhile, there is a trade-off between fidelity and editability, and many methods introduce additional regularization modules (e.g., latent code discriminator [31] and latent space alignment [25]) to address it.

In the second stage, reconstruction and manipulation results from latent codes are refined by high-rate information. Refinement methods are mainly divided into weight modulation and feature modulation. Weight modulation

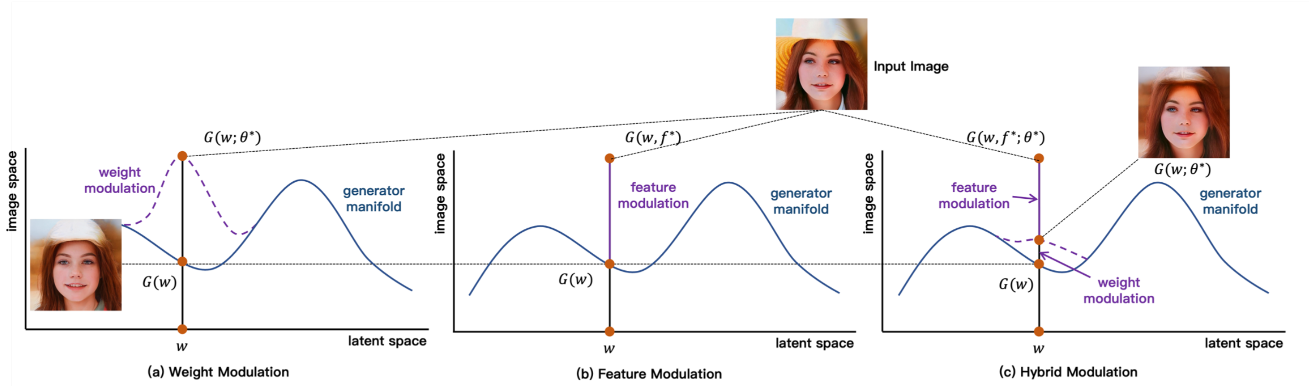

Figure 2. **Comparison of different refinement mechanisms.** We suppose that the refinement results are similar to the given images in all pipelines. The first two illustrate previous mainstream refinement mechanisms: weight, and feature modulation. Weight modulation changes the generator manifold and feature modulation introduces spatial high-rate information to recover image details. The third demonstrates our hybrid refinement method, combining these two modulation mechanisms to retain editing capability, i.e. a smooth updated manifold with image details preservation. We tune the generator on highly invertible and editable areas, which causes lower manifold deviation, and the result of this step is shown on $G(w;\theta^*)$. To reconstruct faithfully, we use feature modulation on the other area and attain $G(w,f^*;\theta^*)$.

methods predict or finetune generator weight to improve fidelity. Some methods [5, 8] use hypernetwork [12] to predict weight offsets. The others tune generator by given images, which attain better fidelity but cost much time [9, 27]. Another branch further inverts images to latent feature, which we call feature modulation. HFGI [34] proposes a distortion consultation approach for high-fidelity reconstruction. SAM [23] segments images into various parts and inverts them into different intermediate layers by predicting “invertibility.” All of them only use one of feature and weight modulation to refine results and suffer editing capability degradation.

2.2. GAN-based Manipulation

GANs’ latent spaces encode highly rich semantic information, which develops the GAN-based manipulation task. It aims to edit given images by changing latent codes in certain directions. Many works propose multiple methods to find semantic editing directions in GANs’ latent spaces. Some methods obtain the edit vectors of the corresponding attributes by supervision with the help of attribute-labeled datasets [7, 10, 30]. For example, InterfaceGAN trains SVMs to find attributes’ boundaries in latent space and achieve promising editing results [28]. Without label annotation, other methods explore the latent space by unsupervised [13, 29, 32, 33] or self-supervised ways [15, 24] to find more semantic directions way.

3. Method

3.1. Preliminaries

Inversion is built to bridge real-world images and GANs’ latent space. As latent codes are low-rate, which limits their reconstruction performance, much research has recently focused on additional high-rate information in generation process, which we call refinement methods. They can be mainly divided into two categories: weight modulation and feature modulation. We first formulate them and analyze the causes of editing capacity degradation.

Formulation. We denote the original generation process as $X = G(w)$, where G is the generator, and w is latent code which can represent each latent space (e.g., $Z/W/W^+$). In the refinement process, we use encoded latent codes which can be attained by off-the-shelf encoders (e.g., pSp [26] and e4e [31]).

Weight modulation methods predict [5] or optimize [20] weight θ by minimizing reconstruction error to get θ^*, and the inference process is denoted as $X = G(w;\theta^*)$. Feature modulation methods invert images into the intermediate feature, which follows $X = G(w, f)$. Defining $\mathcal{L}$ as the distance of images, we can illustrate these two refinement processes as follow:

$$\theta^* = \arg\min_{\theta} \mathcal{L}(x, G(w;\theta)) \tag{1}$$

$$f^* = \arg\min_{f} \mathcal{L}(x, G(w, f)) \tag{2}$$

Impacts on editing capability. Weight and feature modulation impact image manipulation in different aspects. The schematics are shown in Figure 2. Since the feature modulation mechanism fixes the intermediate feature distribution at one of the layers, the effects of edit vectors applied to previous layers cannot edit the features of the latter layers. Although many existing works make efforts to maintain the editing effects, including training with adaptive distortion alignment [17, 34], their solutions still sacrifice fidelity or

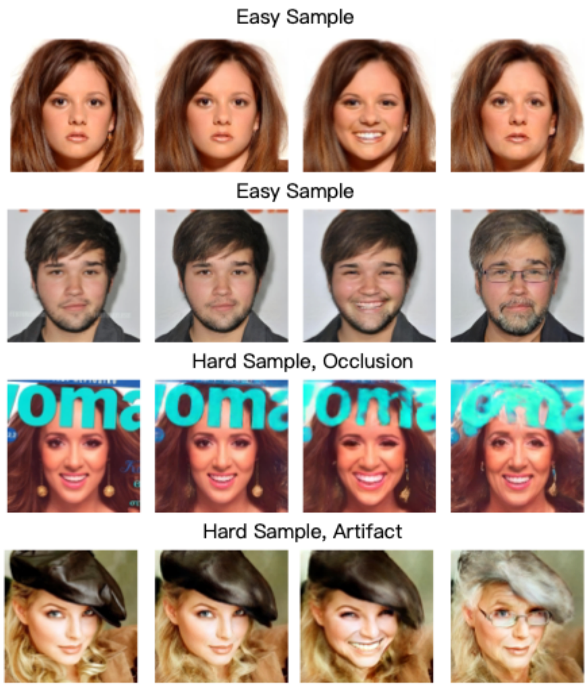

Figure 3. **Impacts on editing capability of weight modulation.** We show the input images, inversion results, and two editing results (smile and age) from PTI [27]. For those easy samples, editing results are reasonable. However, editing capability degrades significantly on hard samples.

editing ability [23]. Generally, it degrades editability more than weight modulation but is better to maintain more visual details due to high-rate spatial information.

Meanwhile, weight modulation shows promising editing performance but also gains unreasonable results on complex images, as shown in Figure 3. Editing results exhibit superior performance on easy samples in contrast to difficult ones. The primary cause of degradation in editing capacity is the considerable weight deviation resulting from refining complex areas, as demonstrated in Figure 2(a). To reconstruct given images, the weight modulation mechanism may change the generator manifold, making it much sharp on the sample point. Therefore, the highly semantic character of the pretrained generator is broken, which decreases the editing capability.

3.2. Domain-Specific Hybrid Refinement

Based on the above analysis, weight modulation gains better editing capability but is easily affected by complex image areas, while feature modulation has better reconstruction ability. Hence, we explore the idea of "divide and conquer" and use both of them to improve GAN inversion. In this work, we conduct **D**omain-Specific **H**ybrid **R**efinement (DHR) to deal with real-world image inversion, and the pipeline is shown in Figure 4.

We first propose the concepts of *in-domain* and *out-of-domain*. *In-domain* refers to areas that have a similar distribution to the generator's output space, which makes them easy to invert. Instead, *out-of-domain* areas misalign with output space, making them challenging to invert. For instance, in the face domain, *in-domain* areas mainly include the face and hair, while *out-of-domain* areas encompass occlusions, backgrounds, and artifacts. Additionally, *in-domain* areas are more editable, such as smiling, lipstick, and eye openness.

Hence, we propose a hybrid refinement method, which segments images into *in-domain* and *out-of-domain* and applies weight and feature modulation to them respectively. Our framework is shown in Figure 4, which consists of three components.

The image Embedding module aims to embed images into latent codes, which we use an off-the-shelf encoder (e.g., e4e [31] and LSAP [25]). Given input images X, the encoder predicts its W^+ space latent codes, which we denote as $w = E(X)$, where E is an encoder.

Domain-Specific Segmentation predicts a binary mask which indicates *in-domain* and *out-of-domain* areas:

$$m = S(X)$$

where $m \in \{0,1\}^{H \times W}$. It segments images into two domains, which will be used for refinement.

In Hybrid Modulation Refinement, weight modulation is employed to refine *in-domain* areas and restore image details for both inversion and editing results. Since the reconstruction discrepancy of *in-domain* areas is low, weight deviation is limited, thereby preserving the editing capacity. Conversely, for *out-of-domain* parts, we use feature modulation to refine them spatially and fix feature distribution during editing. Hence, those hard-to-invert parts do not affect editing ability. Given domain segmentation results m, we modulate weight θ and feature f by minimizing reconstruction error in *in-domain* and *out-of-domain* part, respectively:

$$\theta^*, f^* = \arg\min_{\theta, f} \mathcal{L}(X, G(w, f, m; \theta)) \tag{3}$$

Figure 2 illustrates the difference in the updated generator manifold between our hybrid technique and the previous refinement mechanisms. By adopting the hybrid approach, the generator manifold undergoes minimal changes and does not become too sharp at the given data point, which effectively maintains the editing capability. Notably, we conduct two refinements simultaneously and visualize $G(w, f^*; \theta^*)$ and $G(w; \theta^*)$ separately to demonstrate the effect of our method.

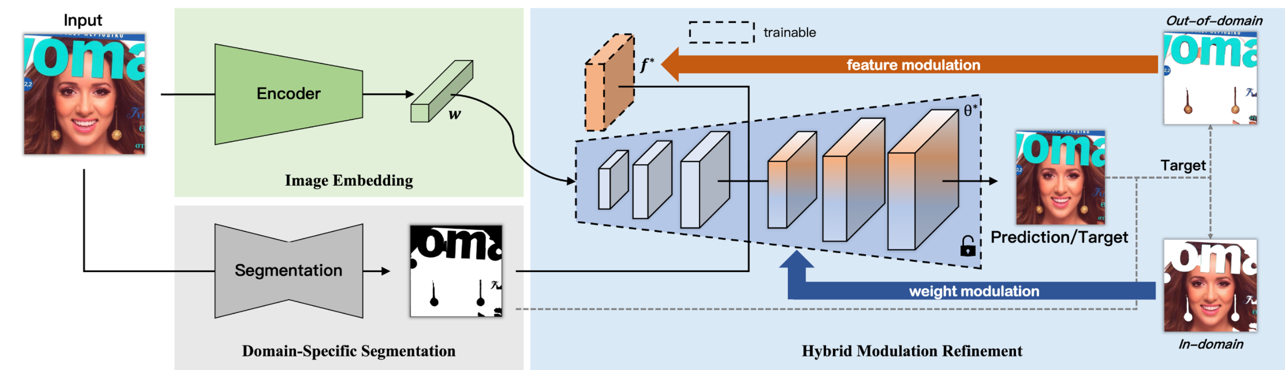

Figure 4. **Overview of our Domain-Specific Hybrid Refinement framework.** We use an off-the-shelf image encoding mechanism and introduce Domain-Specific Segmentation and Hybrid Modulation Refinement. The former segments the input images with two domains: *in-domain* and *out-of-domain*. They are refined by weight modulation and feature modulation in the latter method.

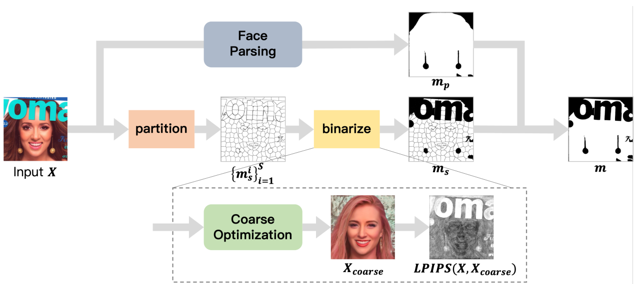

Figure 5. **Illustration of Domain-Specific Segmentation module.** We use a parsing model and superpixel algorithm with coarse optimization to segment input images into *in-domain* (white areas) and *out-of-domain* (black areas) parts.

3.3. Domain-Specific Segmentation

We propose a Domain-Specific Segmentation module to segment images into two domains: *in-domain* and *out-of-domain*. However, training an end-to-end learning-based domain segmentation model requires a large and annotated dataset, which can be costly. Although previous work [23] trains an invertibility prediction model by self-supervision, results are inaccurate in some complex areas, as we demonstrate in the appendix. To address this issue and achieve robust results on real-world images without requiring data annotation, we design an automatic segmentation pipeline consisting of two steps: "partition" and "binarize", which are combined with a parsing model. The pipeline is shown in Figure 5.

We utilize a superpixel algorithm [3] to partition the input image into multiple areas. Each partition is represented as $\{m_s^i\}_{i=1}^S$. Categorizing each partition into *in-domain* and *out-of-domain* without manual annotation is a crucial challenge. However, categorizing each partition into *in-domain* and *out-of-domain* without any manual annotation poses a significant challenge. To address this issue, we employ a coarse optimization in the W space. Here, we initialize the latent codes with mean values and optimize them for a few steps. Since *in-domain* are those easy-to-invert areas, the coarse inversion result X_{coarse} could reconstruct *in-domain* areas effectively. We compute the perceptual loss $L(X, X_{coarse})$ between the coarse reconstruction image and the input image, as shown at the bottom of Figure 5. The white area indicates a higher loss value, while the black area represents a lower loss value. As can be observed, the loss value of the occluded area is significantly higher than that of the face area. We then calculate the average loss of each partition as follows:

$$v_i = \frac{\mathcal{L} \odot m_s^i}{||m_s^i||}$$

and binarize them by threshold τ and attain m_s.

To compensate for the missed segmentation of small areas by superpixel, we also utilize a parsing model. The parsing model categorizes face components like eye, mouth, and background [46]. We obtain parsing results m_p, which we manually categorize as *in-domain* or *out-of-domain* based on the face components. Finally, domain-specific segmentation results are combined by parsing results and superpixel results: $m = m_p \times m_s$. Our Domain-Specific Segmentation module gains fine segmentation results without data annotation and is more effective for real-world images.

3.4. Hybrid Modulation Refinement

To maintain the editing capability of a pre-trained GAN and faithfully recover image details, we introduce a Hybrid Modulation Refinement module. This module includes two refinement aspects: weight modulation and feature modulation. Weight modulation aims to minimize *in-domain* reconstruction error by tuning the generator's parameters, while feature modulation is applied to *out-of-domain* areas

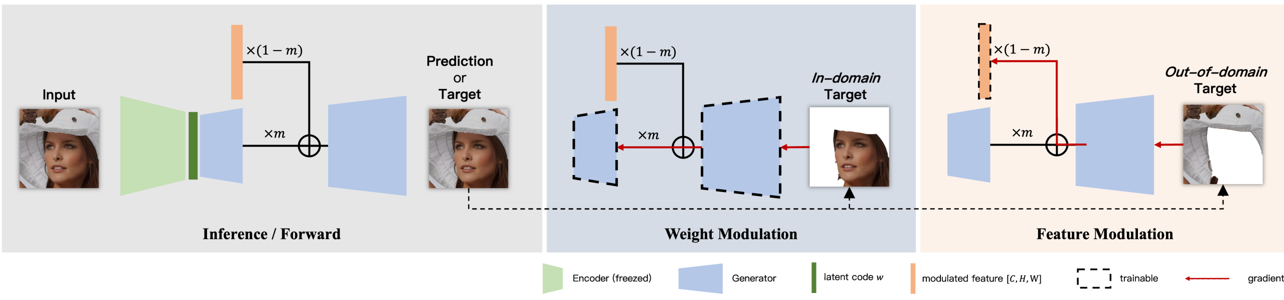

Figure 6. **Illustration of Hybrid Modulation Refinement module.** We refine *in-domain* areas and *out-of-domain* areas by weight and feature modulation, respectively. Black lines indicate forward flows, and red arrows represent gradient flows.

by optimizing the intermediate feature. The forward and backward processes are shown in Figure 6.

For the lth layer of the total k stages in the generator, we denote the original generator's feature as $f_l = G_l(w; \theta)$. An additional modulated feature is marked as f, which is initialized by f_l. When the latent codes are fixed, the original feature f_l is only relevant to θ. We formulate the forward process given the segmentation result m as follows:

$$f' = f_l \odot m + f \odot (1 - m) \tag{4}$$

Then f' represents the output of the first l layers and generates the final images.

For the backward process, we update weight and feature in a parallel optimization process to make them focus on the corresponding domains. We set *in-domain* and *out-of-domain* areas as targets for them, respectively. We use mean square error $\mathcal{L}_2$ and perceptual loss $\mathcal{L}_{lpips}$ as objectives. Calculating the reconstruction errors, we backward loss with segmentation result m:

$$\mathcal{L} = \mathcal{L}_2 + \lambda \mathcal{L}_{lpips} \tag{5}$$

$$\nabla f = \frac{\partial}{\partial f}\left(\frac{\mathcal{L} \odot (1 - m)}{||1 - m||}\right) \tag{6}$$

$$\nabla \theta = \frac{\partial}{\partial \theta}\left(\frac{\mathcal{L} \odot m}{||m||}\right) \tag{7}$$

where λ is a hyper-parameter. The parallel optimization mechanism constrains the impact from different domains. As *in-domain* areas suffer small deviation, $\nabla\theta$ is lower, resulting in less effect on the generator manifold. This significantly retains its highly semantic property and preserves editing capability.

4. Experiments

4.1. Experimental Settings

Datasets. We evaluate all methods on the CelebA-HQ [16, 22] test set (2,824 images). Encoders and the generator are trained on FFHQ [19] (70,000 images).

Baselines. We compare our model to previous state-of-the-art refinement methods, i.e., ReStyle [4], HFGI [34], SAM [23], and PTI [27]. We use pSp [26], e4e [31] and LSAP [25] as encoders. Moreover, the performance of encoder-based methods is also reported. All of weights of encoders come from their official release.

Metrics. We evaluate all methods at two respects: inversion and editing. For inversion ability, we conduct MSE, LPIPS [45], and identity similarity calculated by a face recognition model [14]. MSE straightforwardly measures the image distortion, and LPIPS evaluates the visual discrepancy. Identity similarity further compares the identity consistency during inversion. Moreover, we perform user studies to evaluate the perceptual performance of inversion and editing.

4.2. Main Results

Quantitative results. We first evaluate the reconstruction ability quantitatively. The results are reported in Table 1. We compare our method with four previous refinement methods and employ two encoders. For PTI, we conduct experiments with encoders and their proposed W_{pivot}. As one can see, DHR achieves the best performance on all metrics. Employed with e4e, it gains 0.0036 MSE, which is 7.5% of the vanilla e4e, 8.3% of ReStyle, 17% of HFGI, 25% of SAM, and 48% of PTI. For LPIPS and identity similarity, it demonstrates similar superiority and surpasses other methods by a large margin.

This is because our hybrid refinement approach attains an extraordinary trade-off between fidelity and editability. The previous feature modulation methods (i.e., HFGI and SAM) constrain the effects of the additional feature information by elaborate fusion module or feature regularization to preserve editing capability, and Restyle and PTI encode spatial details in low-rate latent codes or global convolutional weights (hard to recover spatial visual details). All of them employ stronger limitations on high-rate information to maintain editing capability. We will then demonstrate that our DHR not only gains better quantitative reconstruction results but also attain better editing results without any

Figure 7. **Inversion and editing results.** The second column is the segmentation results of *in-domain* and *out-of-domain* areas. Our method restores almost all image details.

regularization.

Qualitative results. We illustrate the inversion and editing results of DHR in Figure 7. The second column is the results from the Domain-Specific Segmentation module, and the third is our inversion results. Some dedicated areas are categorized into *out-of-domain* domain (black area), such as the braid area in the first image. We edit them in four directions, i.e., smile, young, exposure, and lipstick. All image details are preserved in both inversion and editing results, such as hairstyle (the first row), earrings (the fifth row), and hats.

We further conduct a qualitative comparison with other methods, shown in Figure 8. Although inversion results are reasonable, editing capability degradation occurs in baselines, e.g., decorations and occlusion blur. Our editing results are more faithful and photorealistic.

Figure 8. **Comparisons to previous methods.** We compare the inversion and editing results with PTI [4], HFGI [34], and SAM [23]. Although HFGI and SAM reach reasonable inversion results, image distortion and details loss occur in editing results. Our method attains the best fidelity and editing performance. Image details are reserved in both phases, and our results are the most natural.

Method	**Encoder**	**MSE ↓**	**LPIPS ↓**	**Similarity ↑**
ReStyle [4]	pSp	0.0276	0.1298	0.5816
	e4e	0.0429	0.1904	0.5062
HFGI [34]	e4e	0.0210	0.1172	0.6816
	LSAP	0.0210	0.0945	0.7405
SAM [23]	e4e	0.0143	0.1104	0.5568
	LSAP	0.0117	0.0939	0.6184
PTI [27]	e4e	0.0074	0.0750	0.8633
	LSAP	0.0067	0.0666	0.8696
	W_{pivot}	0.0084	0.0845	0.8402
DHR (ours)	e4e	0.0036	0.0455	0.8704
	LSAP	**0.0035**	**0.0436**	**0.8780**
Encoder-Only	pSp [26]	0.0351	0.1628	0.5591
	e4e [31]	0.0475	0.1991	0.4966
	LSAP [25]	0.0397	0.1766	0.5305

Table 1. **Fidelity results on face domain.** We compare DHR to three previous refinement methods with two powerful encoders. The results of these encoders are also presented at the bottom.

Method	Inversion	Editing			
		Smile	Young	Exposure	Lipstick
ReStyle [4]	94%	100%	100%	90%	94%
HFGI [34]	94%	84%	86%	100%	96%
SAM [23]	100%	92%	94%	100%	100%
PTI [27]	96%	100%	100%	84%	88%

Table 2. **User study.** We conduct user studies on inversion and editing tasks. The values in the table indicate the percentage of images where users prefer our results. Results show our method is more faithful and photorealistic.

4.3. User Study

We conduct user studies to demonstrate the performance of inversion and editing. Results are shown in Table 2. We randomly select 50 different images and invert and edit them by HFGI [34], SAM [23], PTI [27], and our method. We then ask three users to make a preference for each pair of images. A higher value implies users prefer our results. As can be seen, our results are highly preferred by users, which are most all above 90%, compared to previous state-of-the-art methods. It illustrates that our method decreases image distortion and attains better photorealism of reconstruction and manipulation results.

5. Conclusion

In this work, we present a novel inversion approach, Domain-Specific Hybrid Refinement, aimed at improving GAN inversion and editing capability. We investigate the causes of editing ability degradation in the refinement process and introduce a "divide and conquer" strategy to address this issue. Our method consists of two main components: Domain-Specific Segmentation and Hybrid Modulation Refinement. The former segments images into *in-domain* and *out-of-domain* parts without data annotation, while the latter refines them by weight and feature modulation, respectively. Our method achieves promising results in both inversion and editing tasks, with significant improvements over existing approaches.

Acknowledgements This work was supported by the National Key Research and Development Program of China (Grant No. 2022YFC3302200), China Postdoctoral Science Foundation (2022M710467), National Key Research and Development Program of China (Grant No. 2021YFF0500900), and Intelligent Logistics Interdisciplinary Team Project of BUPT.

References

[1] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan: How to embed images into the stylegan latent space? In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 4432–4441, 2019. 2

[2] Rameen Abdal, Yipeng Qin, and Peter Wonka. Image2stylegan++: How to edit the embedded images? In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8296–8305, 2020. 2

[3] Radhakrishna Achanta, Appu Shaji, Kevin Smith, Aurelien Lucchi, Pascal Fua, and Sabine Süsstrunk. Slic superpixels compared to state-of-the-art superpixel methods. *IEEE transactions on pattern analysis and machine intelligence*, 34(11):2274–2282, 2012. 5

[4] Yuval Alaluf, Or Patashnik, and Daniel Cohen-Or. Restyle: A residual-based stylegan encoder via iterative refinement. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6711–6720, 2021. 6, 8

[5] Yuval Alaluf, Omer Tov, Ron Mokady, Rinon Gal, and Amit Bermano. Hyperstyle: Stylegan inversion with hypernetworks for real image editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18511–18521, 2022. 2, 3

[6] Antonia Creswell and Anil Anthony Bharath. Inverting the generator of a generative adversarial network. *IEEE transactions on neural networks and learning systems*, 30(7):1967–1974, 2018. 2

[7] Emily Denton, Ben Hutchinson, Margaret Mitchell, and Timnit Gebru. Detecting bias with generative counterfactual face attribute augmentation. 2019. 3

[8] Tan M Dinh, Anh Tuan Tran, Rang Nguyen, and Binh-Son Hua. Hyperinverter: Improving stylegan inversion via hypernetwork. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11389–11398, 2022. 3

[9] Qianli Feng, Viraj Shah, Raghudeep Gadde, Pietro Perona, and Aleix Martinez. Near perfect gan inversion. *arXiv preprint arXiv:2202.11833*, 2022. 2, 3

[10] Lore Goetschalckx, Alex Andonian, Aude Oliva, and Phillip Isola. Ganalyze: Toward visual definitions of cognitive image properties. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 5744–5753, 2019. 3

[11] Shanyan Guan, Ying Tai, Bingbing Ni, Feida Zhu, Feiyue Huang, and Xiaokang Yang. Collaborative learning for faster stylegan embedding. *arXiv preprint arXiv:2007.01758*, 2020. 2

[12] David Ha, Andrew Dai, and Quoc V Le. Hypernetworks. *arXiv preprint arXiv:1609.09106*, 2016. 2, 3

[13] Erik Härkönen, Aaron Hertzmann, Jaakko Lehtinen, and Sylvain Paris. Ganspace: Discovering interpretable gan controls. *Advances in Neural Information Processing Systems*, 33:9841–9850, 2020. 3

[14] Yuge Huang, Yuhan Wang, Ying Tai, Xiaoming Liu, Pengcheng Shen, Shaoxin Li, Jilin Li, and Feiyue Huang. Curricularface: adaptive curriculum learning loss for deep face recognition. In *proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5901–5910, 2020. 6

[15] Ali Jahanian, Lucy Chai, and Phillip Isola. On the" steerability" of generative adversarial networks. *arXiv preprint arXiv:1907.07171*, 2019. 3

[16] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of gans for improved quality, stability, and variation. *arXiv preprint arXiv:1710.10196*, 2017. 1, 6

[17] Tero Karras, Miika Aittala, Janne Hellsten, Samuli Laine, Jaakko Lehtinen, and Timo Aila. Training generative adversarial networks with limited data. In *Proc. NeurIPS*, 2020. 3

[18] Tero Karras, Miika Aittala, Samuli Laine, Erik Härkönen, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Alias-free generative adversarial networks. *Advances in Neural Information Processing Systems*, 34:852–863, 2021. 1

[19] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 4401–4410, 2019. 1, 6

[20] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of stylegan. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 8110–8119, 2020. 1, 2, 3

[21] Zuoxin Li, Fuqiang Zhou, Lu Yang, Xiaojie Li, and Juan Li. Accelerate neural style transfer with super-resolution. *Multimedia Tools and Applications*, 79(7):4347–4364, 2020. 1

[22] Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Deep learning face attributes in the wild. In *Proceedings of the IEEE international conference on computer vision*, pages 3730–3738, 2015. 6

[23] Gaurav Parmar, Yijun Li, Jingwan Lu, Richard Zhang, Jun-Yan Zhu, and Krishna Kumar Singh. Spatially-adaptive multilayer selection for gan inversion and editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11399–11409, 2022. 2, 3, 4, 5, 6, 8

[24] Antoine Plumerault, Hervé Le Borgne, and Céline Hudelot. Controlling generative models with continuous factors of variations. *arXiv preprint arXiv:2001.10238*, 2020. 3

[25] Cao Pu, Lu Yang, Dongxv Liu, Zhiwei Liu, Wenguan Wang, Shan Li, and Qing Song. Lsap: Rethinking inversion fidelity, perception and editability in gan latent space. *arXiv preprint arXiv:2209.12746*, 2022. 2, 4, 6, 8

[26] Elad Richardson, Yuval Alaluf, Or Patashnik, Yotam Nitzan, Yaniv Azar, Stav Shapiro, and Daniel Cohen-Or. Encoding in style: a stylegan encoder for image-to-image translation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2287–2296, 2021. 2, 3, 6, 8

[27] Daniel Roich, Ron Mokady, Amit H Bermano, and Daniel Cohen-Or. Pivotal tuning for latent-based editing of real images. *ACM Trans. Graph.*, 2021. 2, 3, 4, 6, 8

[28] Yujun Shen, Ceyuan Yang, Xiaoou Tang, and Bolei Zhou. Interfacegan: Interpreting the disentangled face representation learned by gans. *IEEE transactions on pattern analysis and machine intelligence*, 2020. 3

[29] Yujun Shen and Bolei Zhou. Closed-form factorization of latent semantics in gans. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1532–1540, 2021. 3

[30] Nurit Spingarn-Eliezer, Ron Banner, and Tomer Michaeli. Gan" steerability" without optimization. *arXiv preprint arXiv:2012.05328*, 2020. 3

[31] Omer Tov, Yuval Alaluf, Yotam Nitzan, Or Patashnik, and Daniel Cohen-Or. Designing an encoder for stylegan image manipulation. *ACM Transactions on Graphics (TOG)*, 40(4):1–14, 2021. 2, 3, 4, 6, 8

[32] Andrey Voynov and Artem Babenko. Unsupervised discovery of interpretable directions in the gan latent space. In *International conference on machine learning*, pages 9786–9796. PMLR, 2020. 3

[33] Binxu Wang and Carlos R Ponce. The geometry of deep generative image models and its applications. *arXiv preprint arXiv:2101.06006*, 2021. 3

[34] Tengfei Wang, Yong Zhang, Yanbo Fan, Jue Wang, and Qifeng Chen. High-fidelity gan inversion for image attribute editing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11379–11388, 2022. 2, 3, 6, 8

[35] Xintao Wang, Yu Li, Honglun Zhang, and Ying Shan. Towards real-world blind face restoration with generative facial prior. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9168–9178, 2021. 1

[36] Tianyi Wei, Dongdong Chen, Wenbo Zhou, Jing Liao, Weiming Zhang, Lu Yuan, Gang Hua, and Nenghai Yu. E2style: Improve the efficiency and effectiveness of stylegan inversion. *IEEE Transactions on Image Processing*, 31:3267–3280, 2022. 2

[37] Lu Yang, Wenhe Jia, Shan Li, and Qing Song. Deep learning technique for human parsing: A survey and outlook. *arXiv preprint arXiv:2301.00394*, 2023. 1

[38] Lu Yang, Zhiwei Liu, Tianfei Zhou, and Qing Song. Part decomposition and refinement network for human parsing. *IEEE/CAA Journal of Automatica Sinica*, 9(6):1111–1114, 2022. 1

[39] Lu Yang, Qing Song, Zhihui Wang, Mengjie Hu, Chun Liu, Xueshi Xin, Wenhe Jia, and Songcen Xu. Renovating parsing r-cnn for accurate multiple human parsing. In *European Conference on Computer Vision*, pages 421–437. Springer, 2020. 1

[40] Lu Yang, Qing Song, Zhihui Wang, and Ming Jiang. Parsing r-cnn for instance-level human analysis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 364–373, 2019. 1

[41] Lu Yang, Qing Song, Zhihui Wang, Zhiwei Liu, Songcen Xu, and Zhihao Li. Quality-aware network for human parsing. *IEEE Transactions on Multimedia*, 2022. 1

[42] Lu Yang, Qing Song, and Yingqi Wu. Attacks on state-of-the-art face recognition using attentional adversarial attack generative network. *Multimedia tools and applications*, 80(1):855–875, 2021. 1

[43] Lu Yang, Qing Song, Xueshi Xin, and Zhiwei Liu. Quality-aware network for face parsing. *arXiv preprint arXiv:2106.07368*, 2021. 1

[44] Shuai Yang, Liming Jiang, Ziwei Liu, and Chen Change Loy. Pastiche master: Exemplar-based high-resolution portrait style transfer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 7693–7702, 2022. 1

[45] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 586–595, 2018. 6

[46] Yinglin Zheng, Hao Yang, Ting Zhang, Jianmin Bao, Dongdong Chen, Yangyu Huang, Lu Yuan, Dong Chen, Ming Zeng, and Fang Wen. General facial representation learning in a visual-linguistic manner. *arXiv preprint arXiv:2112.03109*, 2021. 5