

GLEAN: Generative Latent Bank for Large-Factor Image Super-Resolution

Kelvin C.K. Chan¹ Xintao Wang² Xiangyu Xu¹ Jinwei Gu³ Chen Change Loy^{1*}

¹Nanyang Technological University, Singapore

²Applied Research Center, Tencent PCG ³SenseBrain

{chan0899, xiangyu.xu, ccloy}@ntu.edu.sg xintao.wang@outlook.com gujinwei@sensebrain.ai



Figure 1: **Example of large-factor super-resolution (16 \times).** (a) The low-resolution input (LR). (b) ESRGAN [35] trains the SR generator from scratch, which often produces artifacts and unnatural textures. (c) PULSE [27] achieves more realistic results by GAN inversion, which, however, cannot faithfully recover the structures of the ground-truth. (d) With the proposed generative latent bank, GLEAN is able to generate output that not only is close to the ground-truth, but also possesses realistic textures. (e) The ground-truth (GT).

Abstract

We show that pre-trained Generative Adversarial Networks (GANs), e.g., StyleGAN, can be used as a latent bank to improve the restoration quality of large-factor image super-resolution (SR). While most existing SR approaches attempt to generate realistic textures through learning with adversarial loss, our method, **G**enerative **L**at**E**nt b**A**NK (GLEAN), goes beyond existing practices by directly leveraging rich and diverse priors encapsulated in a pre-trained GAN. But unlike prevalent GAN inversion methods that require expensive image-specific optimization at runtime, our approach only needs a single forward pass to generate the upscaled image. GLEAN can be easily incorporated in a simple encoder-bank-decoder architecture with multi-resolution skip connections. Switching the bank allows the method to deal with images from diverse categories, e.g., cat, building, human face, and car. Images upscaled by GLEAN show clear improvements in terms of fidelity and texture faithfulness in comparison to existing methods as shown in Fig. 1. A high-resolution version of this paper can be found at <https://ckkelvinchan.github.io/>.

1. Introduction

In this study, we explore a new way to employ GAN [9] for image super-resolution. We are interested in the regime

*Corresponding author

of high magnification factors (8 \times to 64 \times), which typical SR methods fail to handle since most details and textures are lost during downsampling. Since the problem is severely underspecified, informative priors become inevitable in this setting, especially in restoring the textural details. Studying large-factor image SR is meaningful as it can potentially improve the state of the arts in SR, and more generally conditional generative models for images.

The notion of GAN has been extensively used in SR with the aim to enrich texture details in an upscaled image. There are two popular approaches to deploy GANs for this task. The more common paradigm [21, 34, 35] trains a generator to handle the upscaling task, where adversarial training is performed by using a discriminator to differentiate real images from the upscaled images produced by the generator. Another possible way to exploit GAN for the task is by GAN inversion [1, 11, 27, 28]. In this setting, one will need to ‘invert’ the generation process of a pre-trained GAN by mapping a corrupted image back to the latent space. A restored image can then be reconstructed from the optimal vector in the latent space.

While both methods are capable of generating more realistic results than approaches that solely rely on ℓ_2 loss, they have some inherent shortcomings. The first paradigm typically trains the SR generator *from scratch* using a combined objective function consisting of a fidelity term and an adversarial loss. In this setting, the generator is responsible for both capturing the natural image characteristics and main-

taining the fidelity to the ground-truth. This inevitably limits the capability of approximating the natural image manifold. As a result, these methods often produce artifacts and unnatural textures. As shown in Fig. 1, while ESRGAN [35] faithfully recovers the structures (*e.g.* pose, ear shape) of the cat, it struggles to produce realistic textures.

The second paradigm resolves the aforementioned problem by making better use of the latent space of GAN through optimization. However, as the low-dimensional latent codes and the constraints in the image space are insufficient to guide the restoration process, these methods often generate images with low fidelity. As shown in Fig. 1, despite being realistic, the output of a representative method, PULSE [27], fails to recover the structures of the ground-truth faithfully. In addition, as the optimization is usually conducted in an iterative manner for each image at runtime, these approaches are often time-consuming.

In our approach, we leverage pre-trained GANs such as StyleGAN [16] to provide rich and diverse priors for the task. Unlike most GAN inversion methods, which also use pre-trained GANs, our method does not involve image-specific optimization at runtime. Once trained, the model only needs a single forward pass to upscale an image, which is more practical for applications that demand fast response. The idea is partially inspired by the classic notion of dictionary [40]. But unlike conventional approaches that construct a finite and imagery-derived dictionary, we exploit GAN as a more effective way for storing priors.

Conditioning and retrieving from a *GAN-based dictionary* is a new and non-trivial question we need to address in this work. We show that pre-trained GANs can be employed as a latent bank in a succinct *encoder-bank-decoder* architecture. This novel architecture allows us to lift the burden of learning both fidelity and texture generation simultaneously in a typical encoder-decoder network since the latent bank already captures rich texture priors. In addition, we show that it is pivotal to condition the bank by passing both the latent vectors and multi-resolution convolutional features from the encoder to achieve high-fidelity results. Symmetrically, multi-resolution cues need to be passed from the bank to the decoder. We show the effectiveness of the proposed method in handling images with challenging poses and structures apart from the large magnification factor. We also demonstrate how the method can be generalized to different categories, *e.g.*, human faces, cats, buildings, by switching different pre-trained GAN latent banks.

2. Related Work

Image Super-Resolution. Many existing SR algorithms [4, 6, 7, 8, 12, 36, 44, 48] directly learn a mapping from the low-resolution images to high-resolution images with a pixel-wise constraint (*e.g.* ℓ_2 loss). While these meth-

ods achieve remarkable results in terms of PSNR, training solely with pixel-wise constraints often results in perceptually unconvincing outputs with severe over-smoothing artifacts [21, 27]. To alleviate the problem, GANs [21, 29, 35, 37] are employed to approximate the natural image manifold, yielding more photo-realistic results. However, as the generator needs to learn both fidelity and natural image characteristics, unnatural artifacts could still be observed in the outputs, especially if one trains the generator from scratch.

Recent interests have shifted to large-factor SR beyond the typical upscaling factors ($2\times$ or $4\times$) [3, 13, 31, 46]. Dahl *et al.* [3] propose a fully probabilistic pixel recursive network for upsampling extremely coarse images with resolution 8×8 . RFB-ESRGAN [31] builds upon ESRGAN and adopts multi-scale receptive fields blocks for $16\times$ SR. VarSR [13] achieves $8\times$ SR by matching the latent distributions of LR and HR images to recover the missing details. Zhang *et al.* [46] perform $16\times$ reference-based SR on paintings with a non-local matching module and a wavelet texture loss. To handle even larger magnification factors, one would need to rely on stronger priors. SR methods specialized on large magnification factors are typically dedicated to the human face category as one could exploit the strong structural prior of faces. Facial priors including facial attributes [22], facial landmarks [18, 26], and identity [10] have been studied. Our work goes beyond previous works and pushes the limit to $64\times$ and generalizes to more categories. Such a large magnification factor is challenging due to its highly ill-posed nature.

GAN Inversion. Given a degraded image x , GAN inversion-based methods [1, 11, 27, 28] in general produce a natural image best approximating x by optimizing $z^* = \operatorname{argmin}_{z \in \mathcal{Z}} \mathcal{L}(G(z), x)$, where \mathcal{Z} is the latent space and $\mathcal{L}(\cdot, \cdot)$ denotes the task-specific objective function. For instance, PULSE [27] iteratively optimizes the latent code of StyleGAN [16] with a pixel-wise constraint between the input and output. mGANprior [11] optimizes multiple latent codes to increase the expressiveness of the model. DGP [28] further finetunes the generator together with the latent code to reduce the gap between the distributions of the training and testing images. A common issue with GAN inversion is that important spatial information may not be faithfully kept due the low-dimensionality of the latent code. Thus, these methods often generate undesirable results that do not resemble the ground-truth. Different from GAN inversion, GLEAN conditions the pre-trained generator with both the latent codes and multi-resolution convolutional features, providing additional spatial guidance for restoration. In addition, GLEAN does not require iterative optimization during inference.

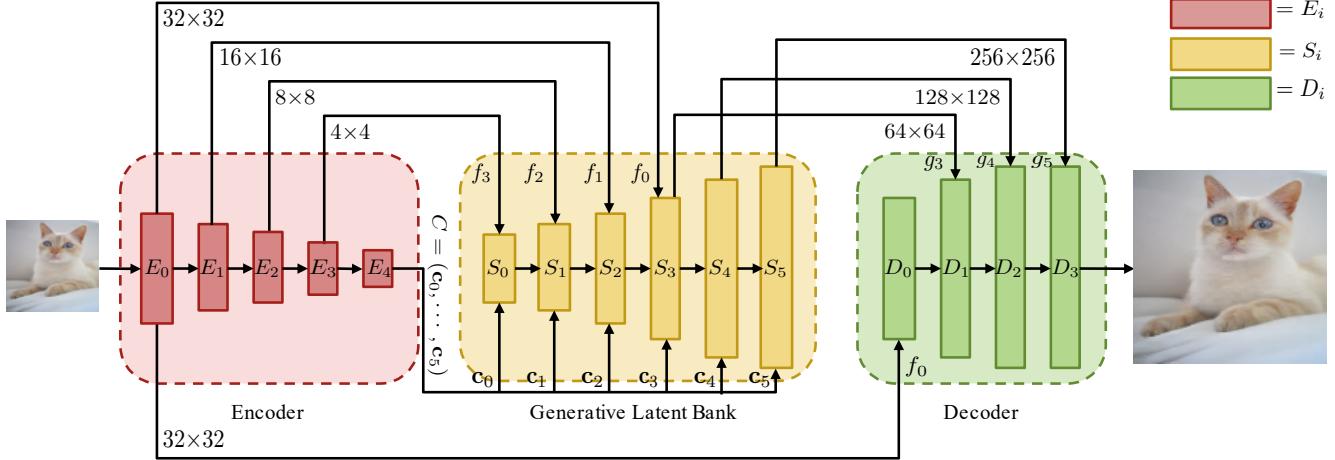


Figure 2: **Overview of GLEAN.** In addition to the latent vectors \mathbf{c}_i , the generator (*i.e.*, the generative latent bank) is also conditioned on the multi-resolution features f_i . With a pre-trained GAN capturing the natural image prior, this encoder-bank-decoder design lifts the burden of learning both fidelity and naturalness in the conventional encoder-decoder architecture. E_i , S_i and D_i denote the encoder blocks, latent bank blocks and decoder blocks, respectively. This example corresponds to an input size of 32×32 and an output size of 256×256 .

3. Methodology

A GAN model that is trained on large-scale natural images captures rich texture and shape priors. Previous studies [1, 11, 27, 28] have shown that such priors can be harvested through GAN inversion to benefit various image restoration tasks. Nonetheless, it remains underexplored how to exploit the priors without the expensive optimization during inversion.

In this study, we devise GLEAN within a novel *encoder-bank-decoder* architecture, which allows one to exploit the generative priors by needing just a single forward pass. An overview of the architecture is depicted in Fig. 2. Given a severely downsampled LR image, GLEAN applies an encoder to extract latent vectors and multi-resolution convolutional features, which capture important high-level cues as well as spatial structure of the LR image. Such cues are used to condition the latent bank, which further produces another set of multi-resolution features for the decoder. Finally, the decoder generates the final output by integrating the features from both the encoder and the latent bank. In this work, we adopt StyleGAN [16, 17] as the generative latent bank due to its exceptional performance. The idea of latent bank can be extended to other generators such as BigGAN [2].

3.1. Encoder

To generate the latent vectors, we first use an RRDB-Net [35] (denoted as E_0) to extract features f_0 from the input LR image. Then, we gradually reduce the resolution of the features by:

$$f_i = E_i(f_{i-1}), \quad i \in \{1, \dots, N\}, \quad (1)$$

where E_i , $i \in \{1, \dots, N\}$, denotes a stack of a stride-2 convolution and a stride-1 convolution. Finally, a convolution and a fully-connected layer are used to generate the latent vectors:

$$C = E_{N+1}(f_N), \quad (2)$$

where C is a matrix whose columns represent the latent vectors for the StyleGAN.

The latent vectors in C capture a compressed representation of the images, providing the generative latent bank with high-level information. To further capture the local structures of the LR image and to provide additional guidance for structure restoration, we also feed multi-resolution convolutional features $\{f_i\}$ into the latent bank.

3.2. Generative Latent Bank

Given the convolutional features $\{f_i\}$ and the latent vectors C , we leverage a pre-trained generator as a latent bank to provide priors for texture and detail generation. As StyleGAN is originally designed for image generation tasks, it cannot be directly integrated into the proposed encoder-bank-decoder framework. In this work, we adapt StyleGAN to our SR network by making three modifications:

1. Instead of taking one single latent vector as the input, each block of the generator takes a different latent vector to improve expressiveness. More specifically, we have $C = (\mathbf{c}_0, \dots, \mathbf{c}_{k-1})$ for k blocks, where each \mathbf{c}_i corresponds to one latent vector. We find that this modification leads to outputs with fewer artifacts. This modification is also seen in previous works [11, 38, 49].
2. To allow conditioning on the additional features from

the encoder, we use an additional convolution in each style block for feature fusion:

$$g_i = \begin{cases} S_0(\mathbf{c}_0, f_N), & \text{if } i = 0, \\ S_i(\mathbf{c}_i, g_{i-1}, f_{N-i}), & \text{otherwise,} \end{cases} \quad (3)$$

where S_i denotes the augmented style block with an additional convolution, and g_i corresponds to the output feature of the i -th augmented style block.

3. Instead of directly generating outputs from the generator, we output the features $\{g_i\}$ and pass them to the decoder to better fuse the features from the latent bank and encoder.

Advantages. The use of generative latent bank is reminiscent of the task of reference-based SR [23, 24, 39, 45, 47], where external HR reference image(s) are employed as an explicit imagery dictionary. While the external HR information leads to marked improvements, the performance is sensitive to the similarity between the inputs and references. This sensitivity may eventually lead to degraded results when the reference images/components are not well selected. Moreover, the size and diversity of those imagery dictionaries are limited by the selected components, impeding the generalization to diverse scenes in practice. In addition, computationally-intensive global matching [47] or component detection/selection [23] is often required to aggregate appropriate information from the references, hindering the applications to scenarios with tight computational constraints. Instead of constructing an imagery dictionary, GLEAN adopts a *GAN-based* dictionary conditioned on a pre-trained GAN. Our dictionary does not depend on any specific components or images. Instead, it captures the distribution of the images and has potentially unlimited size and diversity. Furthermore, GLEAN is computationally efficient without requiring global matching and reference images/components selection.

3.3. Decoder

GLEAN uses an additional decoder with progressive fusion to integrate the features from the encoder and latent bank to generate the output image. It takes the RRDBNet features as inputs and progressively fuse the features with the multi-resolution features from the latent bank:

$$d_i = \begin{cases} D_0(f_0) & \text{if } i = 0, \\ D_i(d_{i-1}, g_{N-2+i}) & \text{otherwise,} \end{cases} \quad (4)$$

where D_i and d_i denote a 3×3 convolution and its output, respectively. Each convolution is followed by a pixel-shuffle [32] layer except the final output layer. With the skip-connection between the encoder and decoder, the information captured by the encoder can be reinforced and hence the latent bank could focus more on the texture and detail generation.

3.4. Training

Similar to existing works [21, 34, 35], we adopt the standard l_2 loss, perceptual loss [14], and adversarial loss for training. More details on the loss function can be found in the appendix. To exploit the generative prior, we keep the weights of the latent bank fixed throughout training. In our preliminary experiments, finetuning the latent bank with the encoder and decoder demonstrates no noticeable improvements. Moreover, it potentially harms the generalizability of the model as the latent bank may eventually bias to the training distribution. It is worth emphasizing that despite GLEAN is trained with similar objectives as in existing works (*e.g.* ESRGAN), the main difference to these methods is that GLEAN leverages a pre-trained generator to directly incorporate the priors into the network, further improving the output quality. We show that the improvement is not due to additional parameters in the generator by comparing GLEAN with ESRGAN⁺, a larger ESRGAN that has similar FLOPs to GLEAN.

4. Experiments

We adopt pre-trained StyleGAN¹ [16] or StyleGAN2² [17] (depending on the availability of pre-trained models) as our latent bank, and use the publicly available codes of existing methods for the comparison in this section. To maintain fairness, we train our model and baselines on the same datasets, including FFHQ [16] and LSUN [41], so that the difference in restoration quality is mainly caused by the algorithms instead of the training distribution. Test set is strictly exclusive from the training. Detailed experimental settings are provided in the appendix.

Qualitative comparison. The qualitative comparison on $16 \times$ SR is shown in Fig. 3. Guided by low-dimensional vectors and constraints in LR space, the outputs of GAN inversion methods are unable to maintain a good fidelity. In particular, PULSE [27] and mGANprior [11] fail to restore a face image with the same identity. In addition, artifacts are observed in their outputs. Through finetuning the generator during optimization, the result of DGP [28] demonstrates significant improvements in both quality and fidelity. However, a slight difference between the identities of the output and ground-truth is still observed. For example, the eyes and lips show noticeable differences.

Methods trained with adversarial loss (SinGAN [30], ESRGAN⁺³ [35]) can preserve the local structures, but fail in synthesizing convincing textures and details. Specifically, SinGAN fails to capture the natural image style, producing a painting-like image. Although ESRGAN⁺ is capable of generating a realistic image, it struggles to synthesize

¹GenForce: <https://github.com/genforce/genforce>

²BasicSR: <https://github.com/xinntao/BasicSR>

³A larger version of ESRGAN with similar FLOPs to GLEAN.

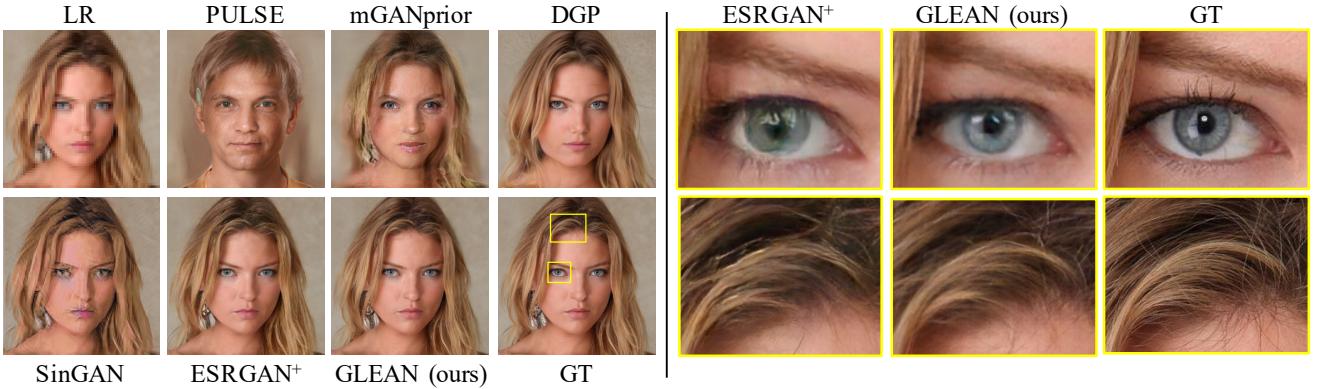


Figure 3: **Comparisons on $16\times$ SR on CelebA-HQ [15].** Only GLEAN is able to maintain high fidelity while synthesizing realistic textures and details: GAN inversion methods fail to preserve the identity, and adversarial loss methods struggle to synthesize fine details. ESRGAN⁺ denotes a larger version with similar FLOPs to GLEAN. (**Zoom-in for best view**)

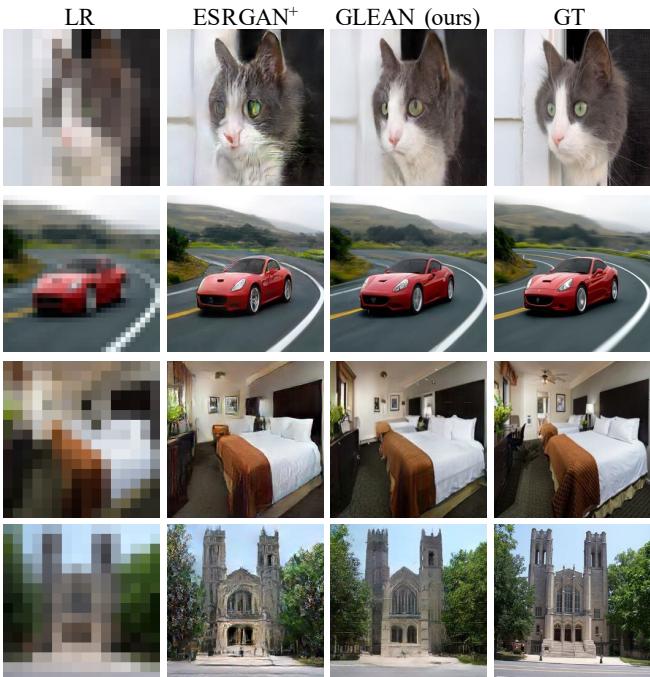


Figure 4: **Results of $16\times$ SR on other categories.** GLEAN can be applied to various categories by switching between StyleGANs trained on different categories. (**Zoom-in for best view**)

fine details and introduces unnatural artifacts in detailed regions. It is worth emphasizing that although ESRGAN⁺ achieves competitive results on human faces, its performances on other categories such as *cats* and *cars* are less promising (see Fig. 1 and Fig. 4). With the latent bank providing natural image priors, GLEAN succeeds in both fidelity and naturalness. For example, when compared to ESRGAN⁺, GLEAN reconstructs eyes with better shape

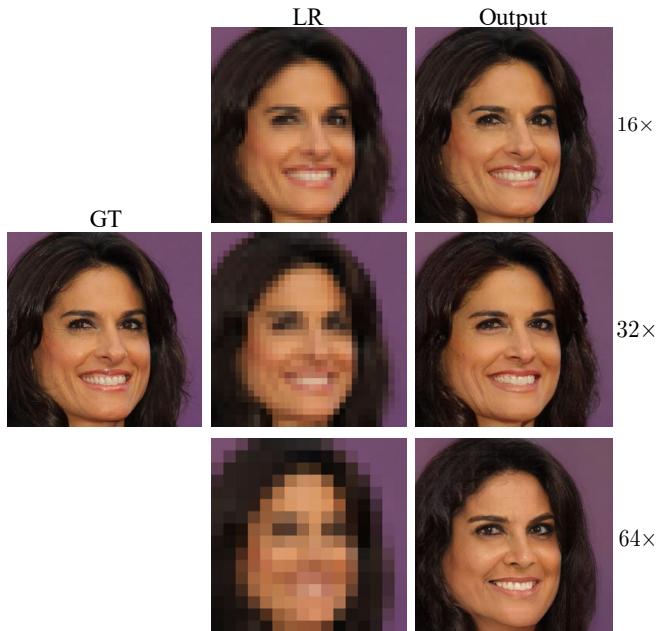


Figure 5: **Results on larger scale factors.** GLEAN reconstructs realistic images highly similar to the GT for up to $64\times$ upscaling factor. (**Zoom-in for best view**)

and details. We further extend our method to larger scale factors in Fig. 5. GLEAN successfully generates perceptually convincing images resembling the ground-truth for up to $64\times$ upscaling.

Robustness to poses and contents. Another appealing property of GLEAN is its robustness to the changes in poses and contents. As shown in Fig. 6, guided by the convolutional features, GLEAN is still able to construct realistic



Figure 6: **Outputs with diverse poses and contents.** Despite GLEAN is trained with aligned human faces, it is able to reconstruct faithful images for non-aligned and non-human faces. PULSE approximates the GT in low resolution (*bottom left*), but its outputs are significantly different from the GT when viewed in high resolution.

Table 1: **Cosine similarity of ArcFace Embeddings [5].** GLEAN achieves a higher similarity than baselines. **Bolded** texts represent the best performance.

	PULSE [27]	mGANprior [11]	DGP [28]
Similarity	0.4047	0.5526	0.7341
	SinGAN [30]	ESRGAN ⁺ [35]	GLEAN
Similarity	0.7718	0.9599	0.9678

Table 2: **Quantitative (PSNR/LPIPS) comparison on 16×SR.** GLEAN outperforms other methods in most categories. ESRGAN⁺ denotes a larger version of ESRGAN [35] having similar FLOPs to GLEAN. **Bolded** texts represent the best performance.

	mGANprior [11]	PULSE [27]	ESRGAN ⁺ [35]	GLEAN
Face [15]	23.66/0.4661	21.83/0.4600	26.76/0.2787	26.84/0.2681
Cat [43]	17.01/0.5556	19.78/0.5241	19.99/0.3482	20.92/0.3215
Car [20]	14.53/0.7228	16.30/0.6491	19.42/0.3006	19.74/0.2830
Bedroom [41]	16.38/0.5439	12.97/0.7131	19.47/0.3291	19.44/0.3310
Tower [41]	15.96/0.4870	13.62/0.7066	17.86/0.3132	18.41/0.2850

images when the images are non-aligned and contain non-human faces despite it is trained on aligned human faces. In contrast, the outputs of PULSE are biased to aligned human faces. Its outputs can only approximate the ground-truths in low resolution. Such robustness enables GLEAN to be applied to diverse categories and scenes such as cats, cars, bedrooms, and towers. Examples are shown in Fig. 4 and more results are provided in the appendix.

Quantitative comparison. To demonstrate the ability of

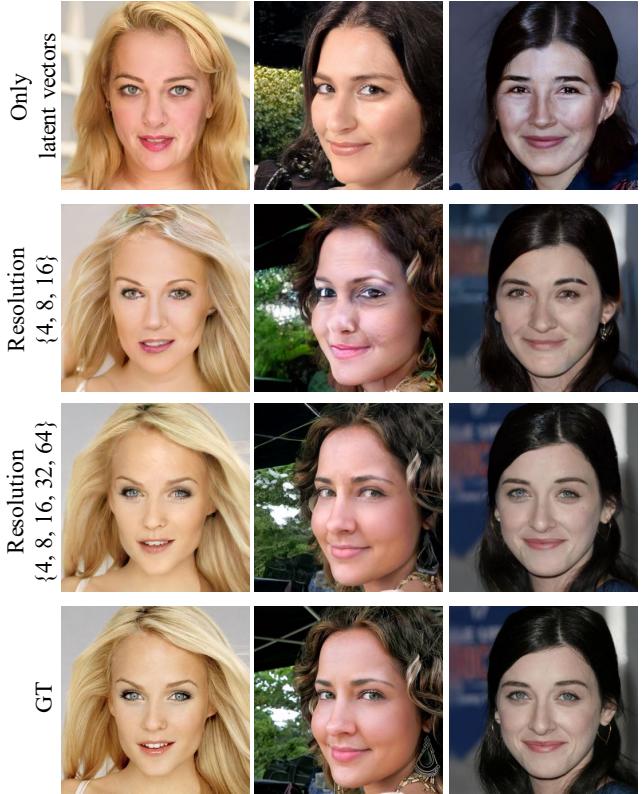


Figure 7: **Effects of the multi-resolution encoder features.** Without the convolutional features, the outputs can only resemble the global attributes (*e.g.* hair color, pose). When adding the encoder features progressively, the network can capture more local structures, better approximating the GT.

GLEAN in producing outputs with high fidelity, we extract 100 images from CelebA-HQ [15] and compute the cosine similarity to the ground-truth on the ArcFace embedding space [5]. As shown in Table 1, GLEAN achieves higher similarity than the baseline methods, validating the superiority of GLEAN.

We additionally provide the quantitative comparison on different categories in Table 2. For each category, we select 100 images and compute their average PSNR and LPIPS [42]. It is observed that mGANprior and PULSE perform significantly worse as they fail to restore the original objects. GLEAN outperforms these methods in most categories, suggesting its effectiveness in generating images with high quality and fidelity.

5. Ablation Studies

Importance of multi-resolution encoder features. We demonstrate how the convolutional features generated from the encoder assist in the restoration of fine details and local structures. We start with only the latent vectors and observe the transition when features are gradually introduced to the

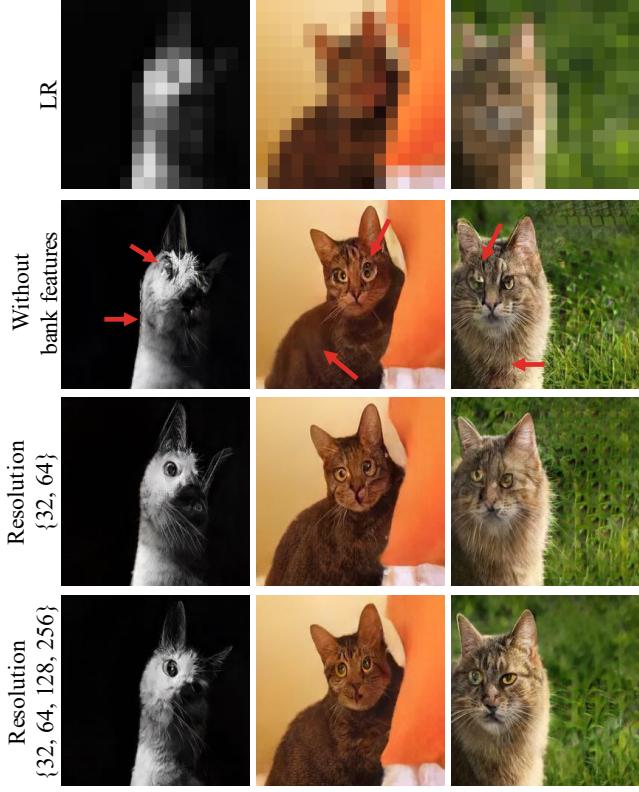


Figure 8: **Effects of the latent bank features.** The rich texture priors captured in the generator lift the burden of the encoder in texture generation. Improvements on both texture and structures are observed when finer features are inserted into the decoder. (Zoom-in for best view)

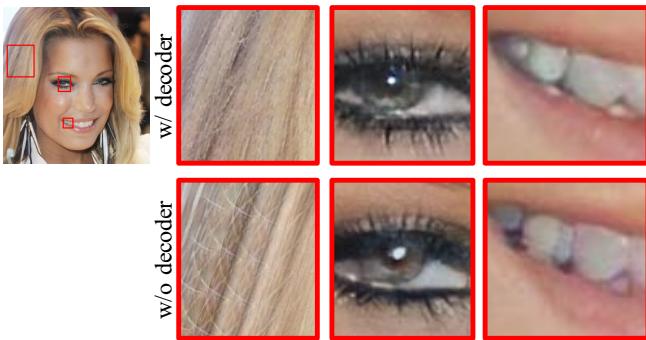


Figure 9: **Contributions of the decoder.** The decoder reinforces the spatial information captured in the encoder features and aggregate them in a coarse-to-fine manner, leading to enhanced quality.

latent bank as conditions. To discard the effects brought by the decoder, we test with a variant of GLEAN where the generator directly produces the output images. The comparison is depicted in Fig. 7.

When all convolutional features are discarded, GLEAN resembles the typical GAN inversion methods that learn

only the latent vectors. Similar to those methods, the network is able to synthesize realistic images given the latent vectors. However, guided only by low-dimensional vectors, in which spatial information is not well-preserved, the network restores only the global attributes such as hair color and poses, but fails to preserve finer details. When providing coarse (from 4×4 to 16×16) convolutional features to the latent bank, more details are recovered and the outputs are better approximating the ground-truths. Further improvements in both quality and fidelity are observed when finer features are passed to the latent bank. The above observations corroborate our hypothesis that the convolutional features are pivotal in guiding the restoration of fine details and local structures, which cannot be reconstructed with only the latent vectors.

Effects of latent bank features. To understand the contributions of the latent bank, we investigate the effects brought by the latent bank features. We start by discarding all the latent bank features, and progressively pass the features to the decoder. The comparison is shown in Fig. 8. Lacking appropriate prior information, the network is responsible for both generating realistic details and maintaining fidelity to the ground-truths. Such a demanding objective eventually leads to outputs that contain flaws in both structure restoration and texture generation. With the latent bank, the burden of texture and details generation is reduced as the generator already captures rich image priors. Therefore, improvements in both structures and textures are observed when passing finer features to the decoder.

Importance of decoder. As shown in Fig. 9, without the decoder, despite being perceptually convincing overall, the output image contains unpleasant artifacts when zoomed in. The decoder allows the network to aggregate the information in a coarse-to-fine manner, leading to more natural details. In addition, the multi-scale skip-connections between the encoder and decoder reinforce the spatial information captured in the encoder features so that the latent bank could focus more on detail generation, further enhancing the output quality.

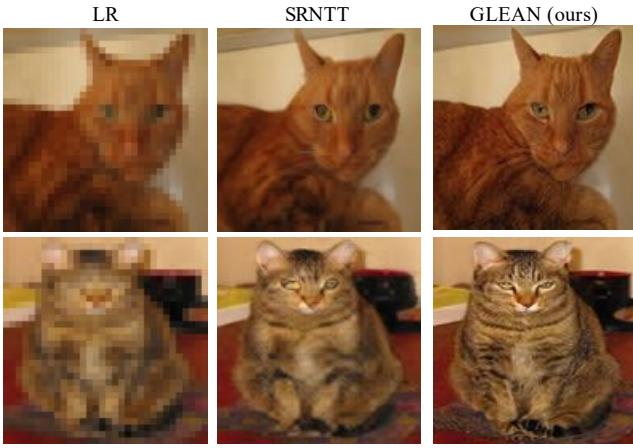
Comparisons with reference-based methods. We assess the efficacy of the new notion of GAN-based dictionary by comparing GLEAN with two representative methods adopting an imagery dictionary for SR – SRNTT [47] and DFDNet [23]. Examples are shown in Fig. 10.

For DFDNet, we evaluate the performance on LR images with unknown degradations⁴. Through pre-constructing a dictionary of facial components (*e.g.* eyes, lips), DFDNet shows remarkable performance on face restoration. However, it cannot produce faithful results on parts absent in the dictionary, such as skin and hair. Therefore, significant in-

⁴We further downsample the LR images to 64×64 to match the input size of GLEAN.



(a) Comparison with DFDNet [23]



(b) Comparison with SRNTT [47]

Figure 10: Comparison to imagery dictionary. (a) DFDNet fails to restore components absent in the dictionary (*e.g.* skin, hair), leading to incoherent outputs. (b) SRNTT is unable to produce faithful fur textures.

coherence is observed in the outputs. Despite GLEAN is trained on the bicubic kernel, it is still capable of producing appealing outputs. More importantly, GLEAN is not confined to improving the visual quality of specific components. Instead, the entire image is super-resolved, leading to coherent and pleasing results. The performance of GLEAN could be further improved by employing multiple degradations during training.

For SRNTT, we follow the same settings and downsample the ground-truth images using the bicubic kernel. With such low-resolution images (32×32), global matching becomes prohibitive, and hence SRNTT fails to transfer the textures from HR reference images. As a result, SRNTT tends to provide blurry textures. By capturing the distribution instead of specific imagery clues, GLEAN does not rely on any explicit textural transferal procedure. This enables

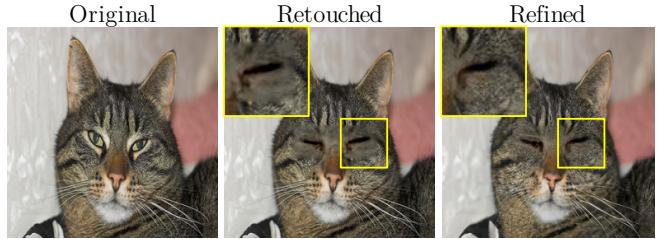


Figure 11: Image Retouching. GLEAN can be used to eliminate unnatural artifacts introduced by amateur retouching. (Zoom-in for best view)

the applicability to large-factor SR, where image matching is extremely difficult. More importantly, with no external images employed, GLEAN does not require any global matching to search for suitable textures/details. This allows GLEAN to be applied to images with larger resolutions, where global matching is computationally prohibitive.

6. Application – Image Retouching

In this section, we present one interesting application of GLEAN. In interactive image retouching, users can manually edit the images based on their preference. However, a perfect output requires tedious and precise retouching. As a result, artifacts are common in the outputs, especially those from amateur retouching. As a powerful super-resolver, GLEAN can be used as an image retouching tool to eliminate unpleasant artifacts.

As shown in Fig. 11, the blending operation in the interactive editing software produces a blurry and incoherent output. Thanks to the capability of GLEAN in producing high quality and fidelity images, GLEAN is able to eliminate the blurry region and generate a coherent output with natural textures. Furthermore, with only a single forward pass for generation, it can be easily incorporated into existing interactive editing software. More examples will be shown in the appendix.

7. Conclusion

We have presented a new way to exploit pre-trained GANs for the task of large-scale super-resolution, up to $64 \times$ upscaling factor. We have shown that a pre-trained GAN can be used as a generative latent bank in an encoder-bank-decoder architecture. Reconstructing photorealistic HR images requires just a single forward pass, thanks to effective ways in conditioning and retrieving rich priors from the bank. The generality of the notion of GAN-based dictionary allows GLEAN to be potentially extended to not only diverse architectures but also various imaging tasks, such as image denoising, inpainting and colorization.

Acknowledgement. This research was conducted in collaboration with SenseTime and supported by the Singapore Government through the Industry Alignment Fund - Industry Collaboration Projects Grant. It is also partially supported by Singapore MOE AcRF Tier 1 (2018-T1-002-056) and NTU SUG.

References

- [1] David Bau, Hendrik Strobelt, William Peebles, Bolei Zhou, Jun-Yan Zhu, Antonio Torralba, et al. Semantic photo manipulation with a generative image prior. *TOG*, 2020. [1](#), [2](#), [3](#)
- [2] Andrew Brock, Jeff Donahue, and Karen Simonyan. Large scale gan training for high fidelity natural image synthesis. *arXiv preprint arXiv:1809.11096*, 2018. [3](#)
- [3] Ryan Dahl, Mohammad Norouzi, and Jonathon Shlens. Pixel recursive super resolution. In *ICCV*, 2017. [2](#)
- [4] Tao Dai, Jianrui Cai, Yongbing Zhang, Shu-Tao Xia, and Lei Zhang. Second-order attention network for single image super-resolution. In *CVPR*, 2019. [2](#)
- [5] Jiankang Deng, Jia Guo, Niannan Xue, and Stefanos Zafeiriou. ArcFace: Additive angular margin loss for deep face ... In *CVPR*, 2019. [6](#)
- [6] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, 2014. [2](#)
- [7] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *TPAMI*, 38(2):295–307, 2016. [2](#)
- [8] Chao Dong, Chen Change Loy, and Xiaoou Tang. Accelerating the super-resolution convolutional neural network. In *ECCV*, pages 391–407. Springer, 2016. [2](#)
- [9] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *NeurIPS*, 2014. [1](#), [11](#)
- [10] Klemen Grm, Walter J Scheirer, and Vitomír Štruc. Face hallucination using cascaded super-resolution and identity priors. *TIP*, 2019. [2](#)
- [11] Jinjin Gu, Yujun Shen, and Bolei Zhou. Image processing using multi-code GAN prior. In *CVPR*, 2020. [1](#), [2](#), [3](#), [4](#), [6](#)
- [12] Xiangyu He, Zitao Mo, Peisong Wang, Yang Liu, Mingyuan Yang, and Jian Cheng. Ode-inspired network design for single image super-resolution. In *CVPR*, 2019. [2](#)
- [13] Sangeek Hyun and Jae-Pil Heo. VarSR: Variational super-resolution network for very low resolution images. In *ECCV*, 2020. [2](#)
- [14] Justin Johnson, Alexandre Alahi, and Fei-Fei Li. Perceptual losses for real-time style transfer and super-resolution. In *ECCV*, pages 694–711. Springer, 2017. [4](#), [11](#)
- [15] Tero Karras, Timo Aila, Samuli Laine, and Jaakko Lehtinen. Progressive growing of GANs for improved quality, stability, and variation. In *ICLR*, 2018. [5](#), [6](#), [11](#), [12](#)
- [16] Tero Karras, Samuli Laine, and Timo Aila. A style-based generator architecture for generative adversarial networks. In *CVPR*, 2019. [2](#), [3](#), [4](#), [11](#)
- [17] Tero Karras, Samuli Laine, Miika Aittala, Janne Hellsten, Jaakko Lehtinen, and Timo Aila. Analyzing and improving the image quality of StyleGAN. *arXiv preprint arXiv:1912.04958*, 2019. [3](#), [4](#), [11](#)
- [18] Deokyun Kim, Minseon Kim, Gihyun Kwon, and Dae-Shik Kim. Progressive face super-resolution via attention to facial landmark. In *BMVC*, 2019. [2](#)
- [19] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. [11](#)
- [20] Jonathan Krause, Michael Stark, Jia Deng, and Fei-Fei Li. 3D object representations for fine-grained categorization. In *ICCV*, 2013. [6](#), [11](#)
- [21] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. In *CVPR*, 2017. [1](#), [2](#), [4](#)
- [22] Mengyan Li, Yuechuan Sun, Zhaoyu Zhang, Haonian Xie, and Jun Yu. Deep learning face hallucination via attributes transfer and enhancement. In *ICME*, 2019. [2](#)
- [23] Xiaoming Li, Chaofeng Chen, Shangchen Zhou, Xianhui Lin, Wangmeng Zuo, and Lei Zhang. Blind face restoration via deep multi-scale component dictionaries. In *ECCV*, 2020. [4](#), [7](#), [8](#)
- [24] Xiaoming Li, Wenyu Li, Dongwei Ren, Hongzhi Zhang, Meng Wang, and Wangmeng Zuo. Enhanced blind face restoration with multi-exemplar images and adaptive spatial feature fusion. In *CVPR*, 2020. [4](#)
- [25] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016. [11](#)
- [26] Cheng Ma, Zhenyu Jiang, Yongming Rao, Jiwen Lu, and Jie Zhou. Deep face super-resolution with iterative collaboration between attentive recovery and landmark estimation. In *CVPR*, 2020. [2](#)
- [27] Sachit Menon, Alexandru Damian, Shijia Hu, Nikhil Ravi, and Cynthia Rudin. PULSE: Self-supervised photo upscaling via latent space exploration of generative models. In *CVPR*, 2020. [1](#), [2](#), [3](#), [4](#), [6](#), [11](#), [12](#)
- [28] Xingang Pan, Xiaohang Zhan, Bo Dai, Dahua Lin, Chen Change Loy, and Ping Luo. Exploiting deep generative prior for versatile image restoration and manipulation. In *ECCV*, 2020. [1](#), [2](#), [3](#), [4](#), [6](#)
- [29] Mehdi S M Sajjadi, Bernhard Schölkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *ICCV*, pages 4501–4510, 2017. [2](#)
- [30] Tamar Rott Shaham, Tali Dekel, and Tomer Michaeli. SiGAN: Learning a generative model from a single natural image. In *ICCV*, 2019. [4](#), [6](#)
- [31] Tai Zhang Shang, Qiuju Dai, Shengchen Zhu, Tong Yang, and Yandong Guo. Perceptual extreme super resolution network with receptive field block. In *CVPRW*, 2020. [2](#)
- [32] Wenzhe Shi, Jose Caballero, Ferenc Huszár, Johannes Totz, Andrew P Aitken, Rob Bishop, Daniel Rueckert, and Zehan Wang. Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network. In *CVPR*, pages 1874–1883, 2016. [4](#)

- [33] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *ICLR*, 2015. 11
- [34] Xintao Wang, Ke Yu, Chao Dong, and Chen Change Loy. Recovering realistic texture in image super-resolution by deep spatial feature transform. In *CVPR*, 2018. 1, 4, 11
- [35] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaou Tang. ESRGAN: Enhanced super-resolution generative adversarial networks. In *ECCVW*, 2018. 1, 2, 3, 4, 6, 11
- [36] Xiangyu Xu, Yongrui Ma, and Wenxiu Sun. Towards real scene super-resolution with raw images. In *CVPR*, 2019. 2
- [37] Xiangyu Xu, Deqing Sun, Jinshan Pan, Yujin Zhang, Hanspeter Pfister, and Ming-Hsuan Yang. Learning to super-resolve blurry face and text images. In *ICCV*, 2017. 2
- [38] Yinghao Xu, Yujun Shen, Jiapeng Zhu, Ceyuan Yang, and Bolei Zhou. Generative hierarchical features from synthesizing images. *arXiv preprint arXiv:2007.10379*, 2020. 3
- [39] Xu Yan, Weibing Zhao, Kun Yuan, Ruimao Zhang, Zhen Li, and Shuguang Cui. Towards content-independent multi-reference super-resolution: Adaptive pattern matching and feature aggregation. In *ECCV*, 2020. 4
- [40] Jianchao Yang, John Wright, Thomas S Huang, and Yi Ma. Image super-resolution via sparse representation. *TIP*, 19(11):20861–2873, 2010. 2
- [41] Fisher Yu, Ari Seff, Yinda Zhang, Shuran Song, Thomas Funkhouser, and Jianxiong Xiao. LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop. *arXiv preprint arXiv:1506.03365*, 2015. 4, 6, 11
- [42] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *arXiv preprint arXiv:1801.03924*, 2018. 6
- [43] Weiwei Zhang, Jian Sun, and Xiaou Tang. Cat head detection - how to effectively exploit shape and texture features. In *ECCV*, 2008. 6, 11
- [44] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, 2018. 2
- [45] Yang Zhang, Ivor W Tsang, Yawei Luo, Changhui Hu, Xiaobo Lu, and Xin Yu. Copy and Paste GAN: Face hallucination from shaded thumbnails. In *CVPR*, 2020. 4
- [46] Yulun Zhang, Zhifei Zhang, Stephen DiVerdi, Zhaowen Wang, Jose Echevarria, and Yun Fu. Texture hallucination for large-scale painting super-resolution. In *ECCV*, 2020. 2
- [47] Zhifei Zhang, Zhaowen Wang, Zhe Lin, and Hairong Qi. Image super-resolution by neural texture transfer. In *CVPR*, 2019. 4, 7, 8
- [48] Shangchen Zhou, Jiawei Zhang, Wangmeng Zuo, and Chen Change Loy. Cross-scale internal graph neural network for image super-resolution. In *NeurIPS*, 2020. 2
- [49] Jiapeng Zhu, Yujun Shen, Deli Zhao, and Bolei Zhou. In-domain GAN inversion for real image editing. In *ECCV*, 2020. 3

Appendix

We first provide the implementation details of GLEAN in Sec. A. We then provide additional qualitative results on various categories and scale factors in Sec. B.1. Finally, we demonstrate the application of GELAN to the task of image retouching in Sec. B.2.

A. Training Details of GLEAN

We adopt pre-trained StyleGAN⁵ [16] or StyleGAN2⁶ [17] as our generative latent bank. In this section, we assume the latent bank is pre-trained and present the training details of GLEAN (*i.e.* the encoder-bank-decoder network). Note that the weights of the latent bank are fixed when training GLEAN to better employ the generative prior and to avoid biasing to the training distribution.

We train GLEAN on five categories including human faces, cats, cars, towers, and bedrooms. The training and test datasets used in our experiments are summarized in Table 3. Since StyleGAN produces images with fixed size, we resize the images in the datasets for our experiments.

Table 3: Datasets used in our experiments.

	Train	Test
Human faces	FFHQ [16]	CelebA-HQ [15]
Cats	LSUN-train [41]	CAT [43]
Cars	LSUN-train [41]	Cars [20]
Bedrooms	LSUN-train [41]	LSUN-validate [41]
Towers	LSUN-train [41]	LSUN-validate [41]

Following previous works [34, 35], the objective function for GLEAN consists of three terms. MSE loss is used to guide the fidelity of the output images:

$$\mathcal{L}_{mse} = \frac{1}{N} \|\hat{y} - y\|_2^2, \quad (5)$$

where N , \hat{y} , and y denote the number of pixels, the output image, and the ground-truth image, respectively. We further incorporate perceptual loss [14] and adversarial loss [9] to improve the perceptual quality:

$$\mathcal{L}_{percep} = \frac{1}{N} \|f(\hat{y}) - f(y)\|_2^2, \quad (6)$$

$$\mathcal{L}_{gen} = \log(1 - D(\hat{y})), \quad (7)$$

where $f(\cdot)$ denotes the feature embedding space of the VGG16 [33] network, and D corresponds to the StyleGAN discriminator. The resulting objective function is a weighted mean of the three losses:

$$\mathcal{L}_g = \mathcal{L}_{mse} + \alpha_{percep} \cdot \mathcal{L}_{percep} + \alpha_{gen} \cdot \mathcal{L}_{gen}. \quad (8)$$

⁵GenForce: <https://github.com/genforce/genforce>

⁶BasicSR: <https://github.com/xinntao/BasicSR>

In all our experiments, we set $\alpha_{percep}=\alpha_{gen}=10^{-2}$. For the discriminator, we maximize

$$\mathcal{L}_d = \log(1 - D(\hat{y})) + \log D(y). \quad (9)$$

We adopt Cosine Annealing Scheme [25] and Adam optimizer [19] in training. The number of iterations is 300K and the initial learning rate is 10^{-4} . The batch size is 8 for human faces and 16 for other categories. We train our models using two Nvidia V100 GPUs.

B. Qualitative Results

B.1. Super-Resolution

Randomly-Selected Examples. In Fig. 12, we show the results of randomly-selected examples from CelebA-HQ [15]. By optimizing only the latent codes, PULSE [27] produces outputs with low-fidelity. In contrast, guided by the encoder features and our generative latent bank, GLEAN achieves remarkable quality and fidelity, demonstrating the effectiveness of our designs.

Scale Factors and Categories. GLEAN is extensible to various scale factors (from $8\times$ to $64\times$) and categories (*e.g.* faces, cats, cars, bedrooms, towers). From Fig. 13 to Fig. 18, we see that GLEAN outperforms DGP and ESRGAN⁺ in both fidelity and quality. It is noteworthy that the performance of DGP and ESRGAN⁺ are less promising on categories other than human faces.

B.2. Image Retouching

In interactive image retouching, users can manually edit the images based on their preference. For instance, users can change the facial expression of an object and perform geometric transformations for enlarging eyes. However, a perfect output requires tedious and precise retouching. As a result, artifacts are common in the outputs from amateur retouching.

GLEAN allows the possibility of performing realistic refinement of imperfect retouching. More specifically, given a retouched image, we can first downsample the image to a smaller resolution, where the artifacts vanished. We can then upsample it back to the original resolution. With GLEAN as a powerful super-resolver, we can obtain an output with unnatural artifacts suppressed.

As shown in Fig. 19, GLEAN is able to correct the unnatural artifacts introduced by amateur retouching while being similar to the retouched images, realistic, and coherent with the unaltered regions. In addition, since GLEAN requires only a single forward pass, it can be used in interactive image editing software to allow a more flexible retouching.

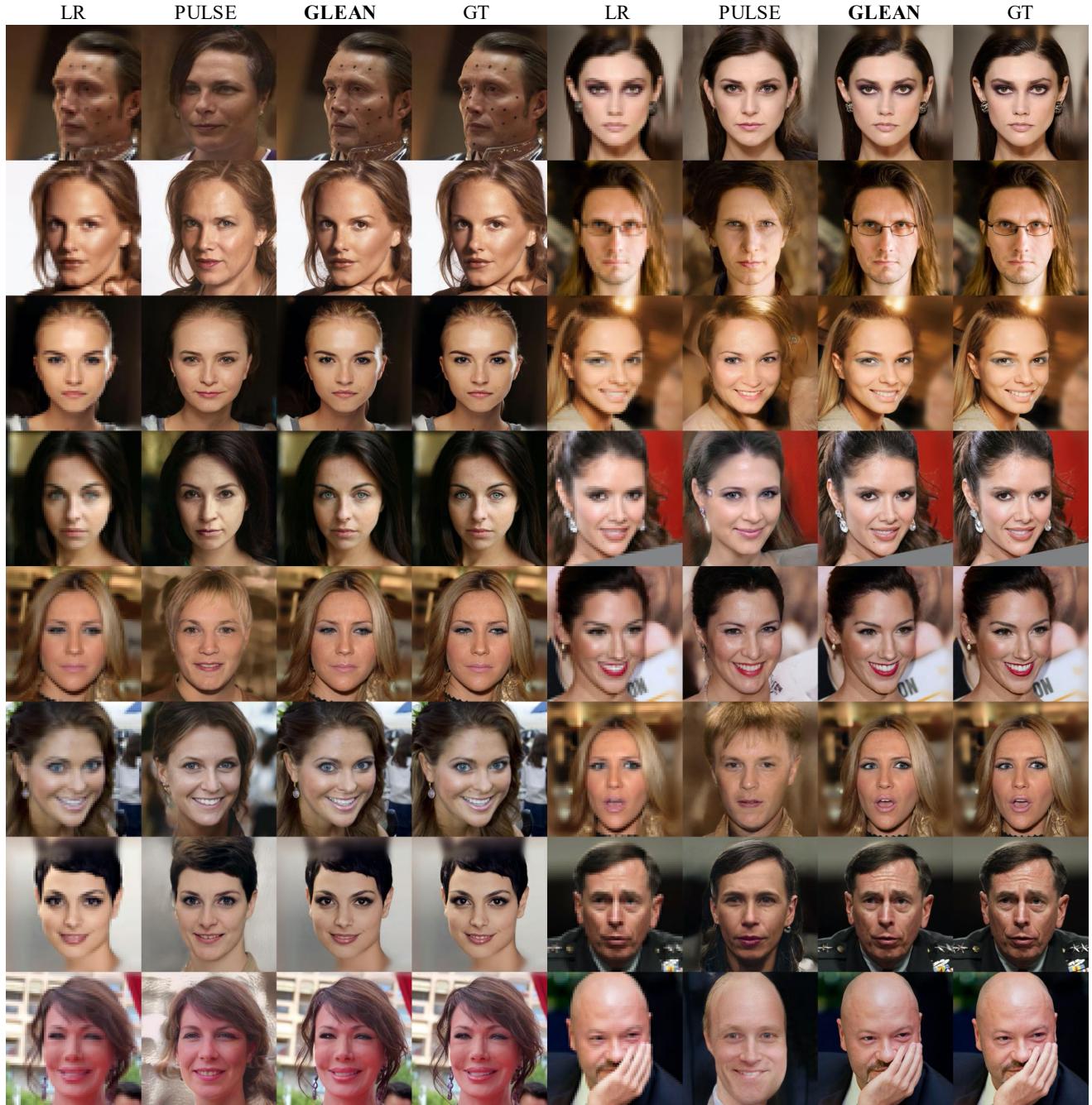


Figure 12: **Comparison to PULSE on randomly-selected examples from CelebA-HQ [15].** By optimizing only the latent vectors, the outputs of PULSE [27] differ significantly from the ground-truths. With our novel designs, GLEAN produces outputs highly similar to the ground-truths.

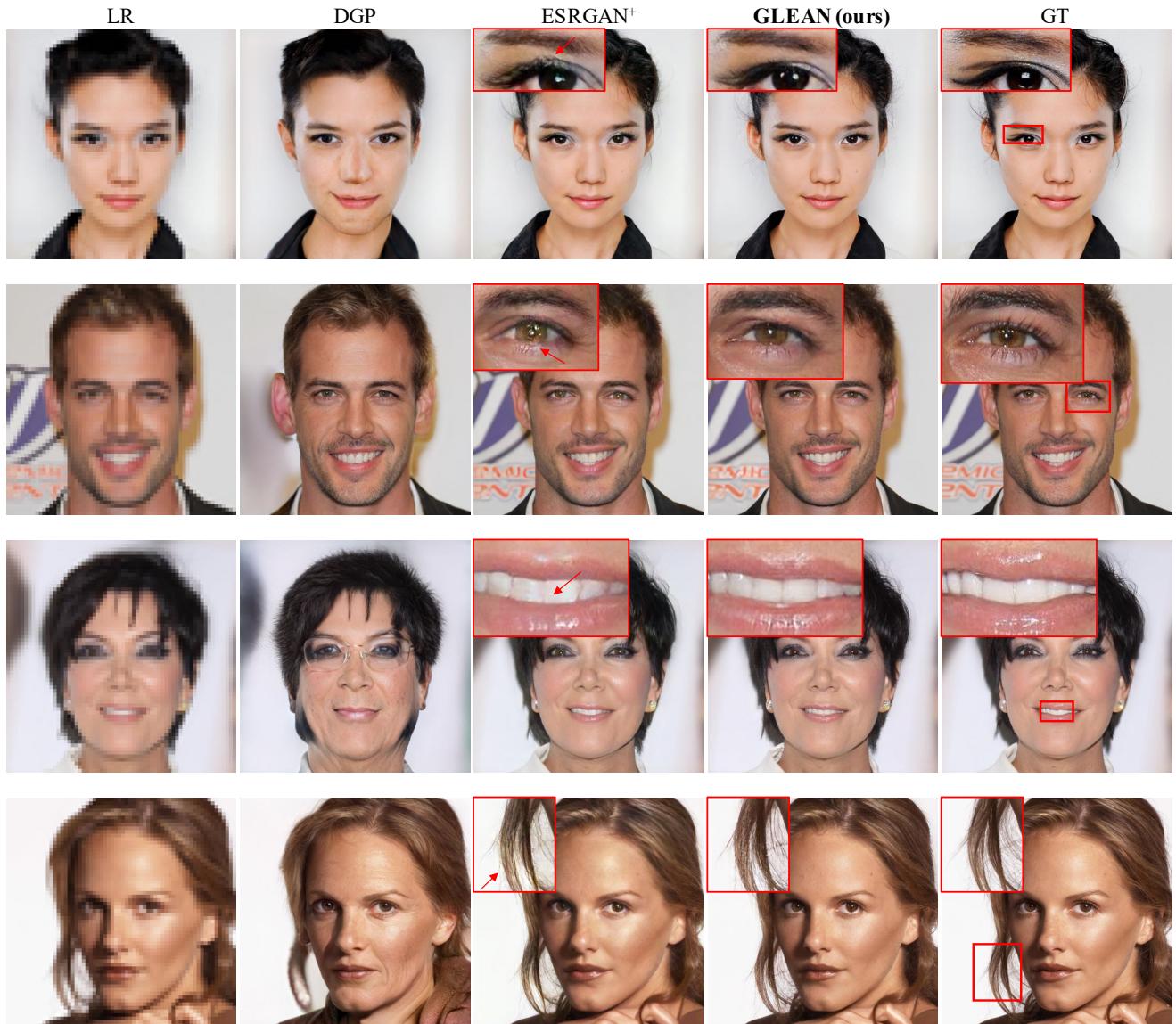


Figure 13: **Comparison with DGP and ESRGAN⁺.** The outputs of DGP show noticeable identity differences to the ground-truths. ESRGAN⁺ shows unpleasant artifacts for the fine details. (**Zoom-in for best view**)

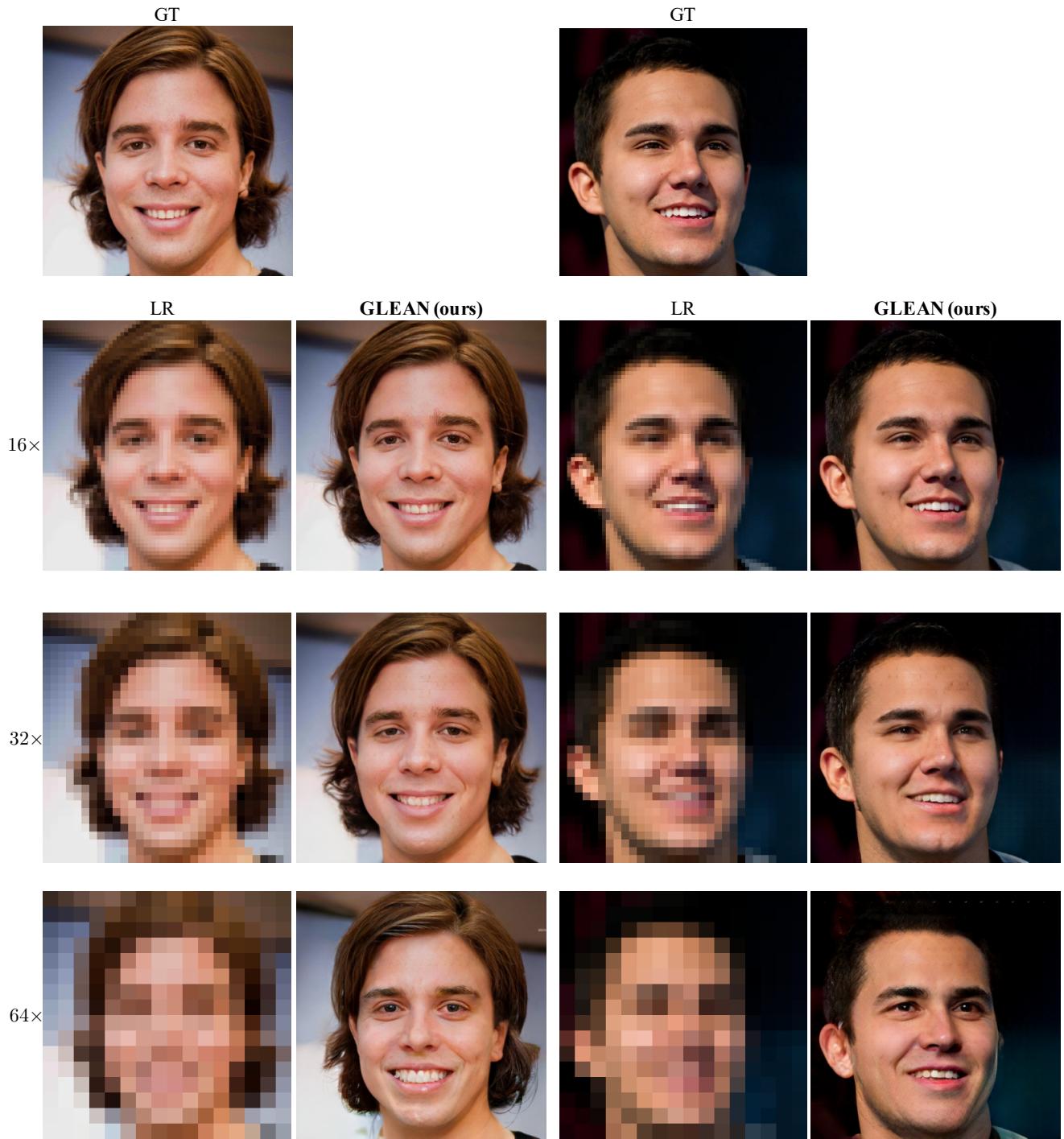


Figure 14: **Performance of GLEAN on 16x, 32x, and 64x SR.** GLEAN is able to synthesize images well resembling the ground-truths for up to 64x upsampling.

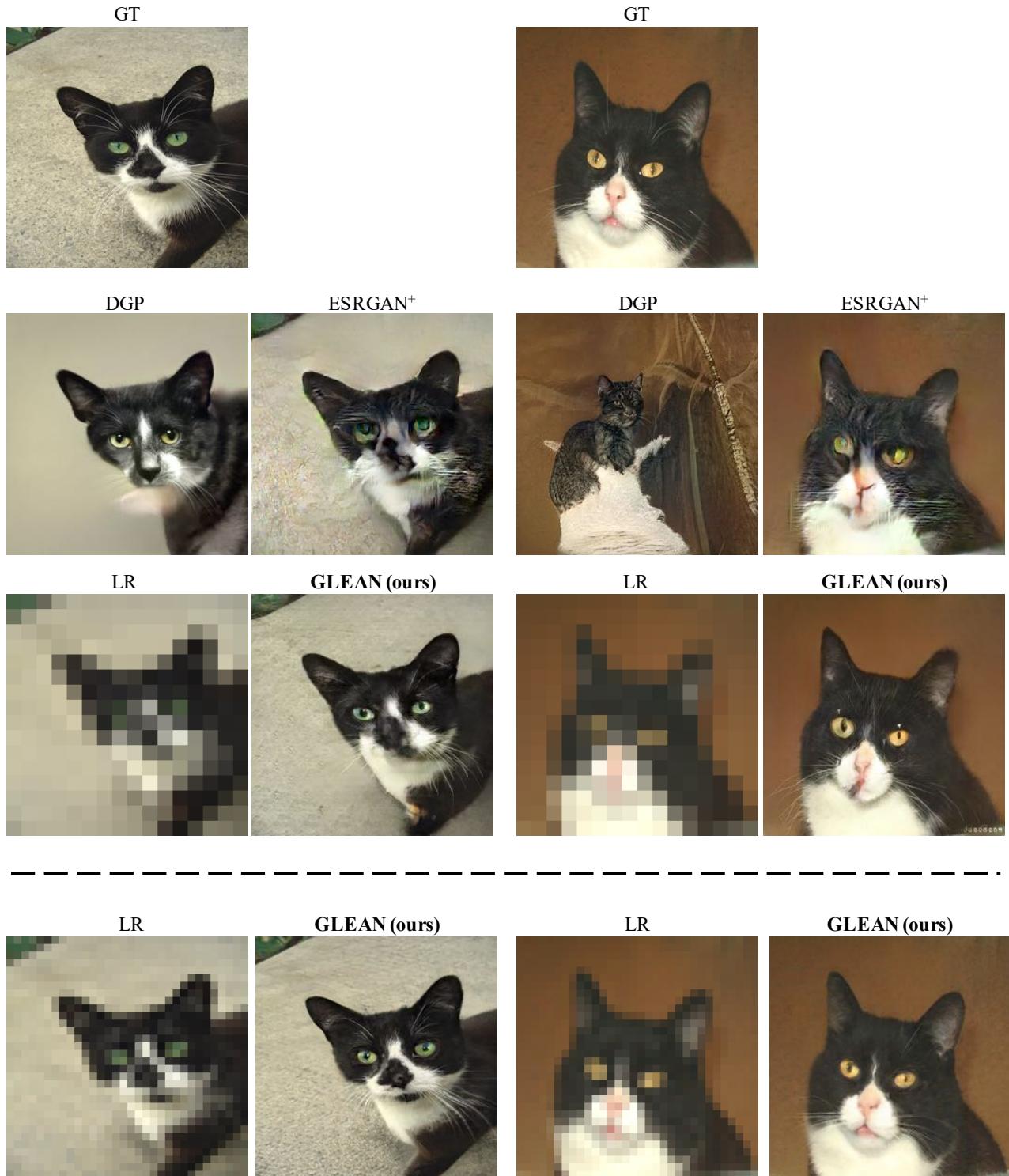


Figure 15: **(Top)** Comparison with DGP and ESRGAN⁺ on *Cats*. DGP produces outputs with low fidelity; ESRGAN⁺ fails to synthesize realistic textures. **(Bottom)** Performance of GLEAN on 8× SR. GLEAN produces realistic outputs that are highly similar to the ground-truths. **(Zoom-in for best view)**



Figure 16: **(Top)** Comparison with DGP and ESRGAN⁺ on *Cars*. DGP produces outputs with low fidelity; ESRGAN⁺ fails to synthesize realistic textures. **(Bottom)** Performance of GLEAN on 8× SR. GLEAN produces realistic outputs that are highly similar to the ground-truths. **(Zoom-in for best view)**



Figure 17: **(Top)** Comparison with DGP and ESRGAN⁺ on *Bedrooms*. DGP produces outputs with low fidelity; ESRGAN⁺ fails to synthesize realistic textures. **(Bottom)** Performance of GLEAN on 8× SR. GLEAN produces realistic outputs that are highly similar to the ground-truths. **(Zoom-in for best view)**



Figure 18: **(Top)** Comparison with DGP and ESRGAN⁺ on *Towers*. DGP produces outputs with low fidelity; ESRGAN⁺ fails to synthesize realistic textures. **(Bottom)** Performance of GLEAN on 8× SR. GLEAN produces realistic outputs that are highly similar to the ground-truths. **(Zoom-in for best view)**

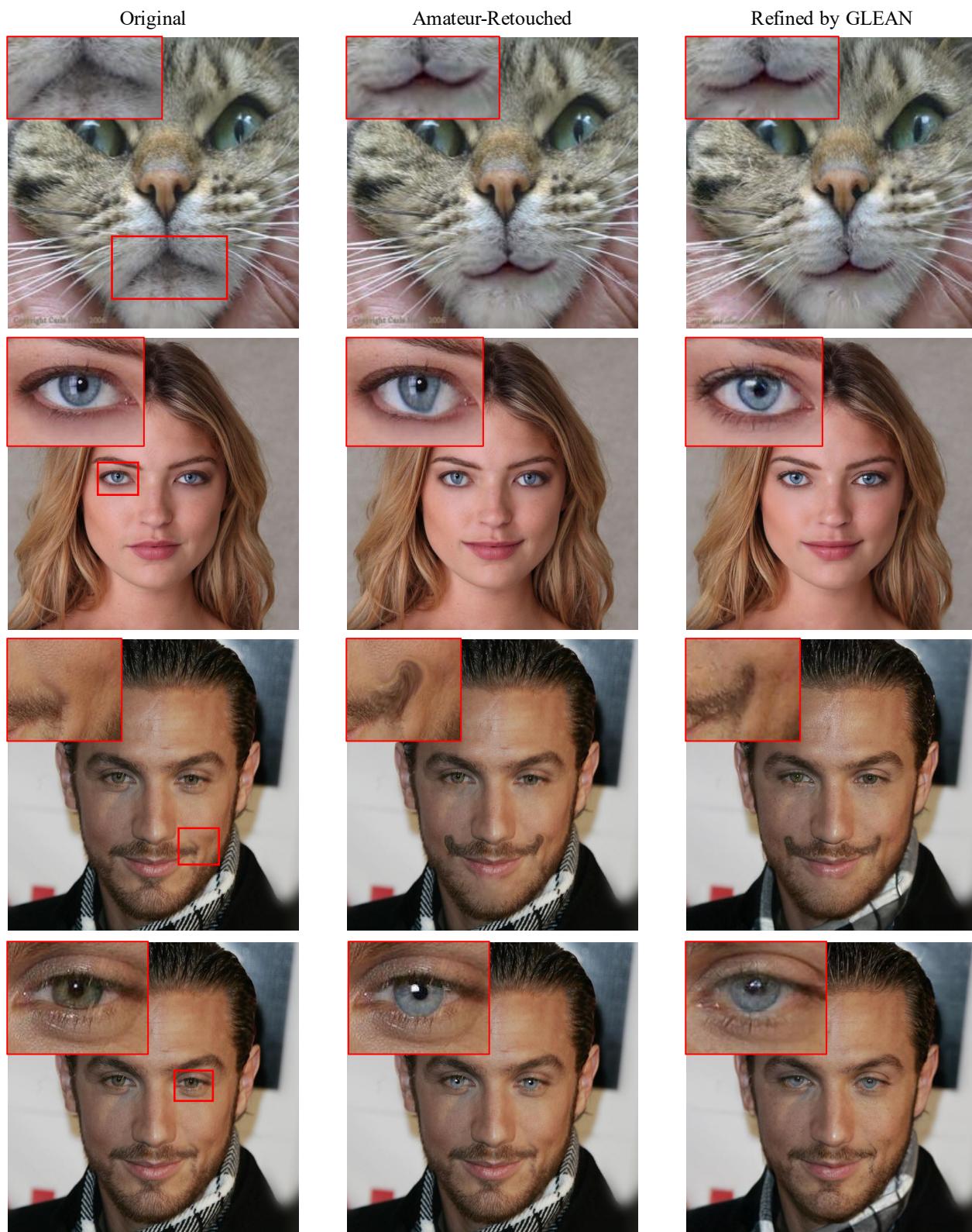


Figure 19: **Results on image retouching.** GLEAN can be used to correct unpleasant artifacts introduced by amateur retouching. (**Zoom-in for best view**)