GLEAN: Generative Latent Bank for Image Super-Resolution and Beyond

Kelvin C.K. Chan, Xiangyu Xu, Xintao Wang, Jinwei Gu, *Senior Member, IEEE*
Chen Change Loy, *Senior Member, IEEE*

Abstract—We show that pre-trained Generative Adversarial Networks (GANs) such as StyleGAN and BigGAN can be used as a latent bank to improve the performance of image super-resolution. While most existing perceptual-oriented approaches attempt to generate realistic outputs through learning with adversarial loss, our method, **G**enerative **L**at**E**nt b**AN**k (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 for restoration. GLEAN can be easily incorporated in a simple encoder-bank-decoder architecture with multi-resolution skip connections. Employing priors from different generative models allows GLEAN to be applied to diverse categories (*e.g.*, human faces, cats, buildings, and cars). We further present a lightweight version of GLEAN, named LightGLEAN, which retains only the critical components in GLEAN. Notably, LightGLEAN consists of only 21% of parameters and 35% of FLOPs while achieving comparable image quality. We extend our method to different tasks including image colorization and blind image restoration, and extensive experiments show that our proposed models perform favorably in comparison to existing methods. Codes and models are available at https://github.com/open-mmlab/mmediting.

Index Terms—Super-resolution, colorization, restoration, generative adversarial networks, generative prior.

1 Introduction

In this study, we explore a new way to employ GAN [3] for image super-resolution. Since the task of super-resolution is severely underspecified, strong priors are usually required to regularize the restoration process, and the generative prior of GANs has become one of the most widely-used priors thanks to its remarkable abilities to approximate the natural image manifold and synthesize high-quality images.

There are two popular approaches to deploy GANs for super-resolution. The more common paradigm [1], [4], [5] trains a generator to handle the restoration, where adversarial training is performed by using a discriminator to differentiate real images from the upscaled images produced by the generator. Another way to exploit GAN is GAN inversion [2], [6], [7], [8], which needs to 'invert' the generation process of a pre-trained GAN by mapping an image back to the latent space. A restored image can then be reconstructed from the optimal vector in the latent space.

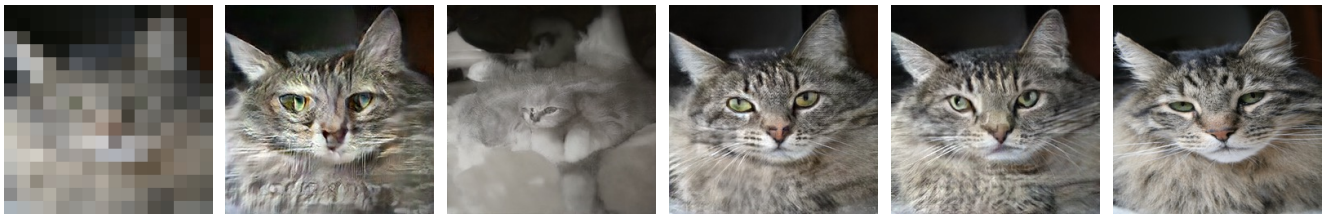
While both methods are capable of generating more realistic results than those approaches that solely rely on pixel-wise loss, they have some inherent shortcomings. The first paradigm typically trains the generator *from scratch* using a combined objective function consisting of an adversarial loss and a fidelity loss. In this setting, the generator is responsible for both capturing natural image characteristics and maintaining fidelity to the ground truth. This inevitably limits the capability of approximating the natural image manifold. As a result, these methods often produce artifacts, such as unnatural textures and colors. As shown in Fig. 1(b), while ESRGAN [1] 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, because the low-dimensional latent codes and the constraints in the image space are insufficient in guiding the restoration process, these methods often generate images with low fidelity. As shown in Fig. 1(c), although PULSE [2] successfully produces a cat-like object, the GAN-inversion-based method fails to recover the structures of the ground-truth faithfully. In addition, since the optimization is usually conducted in an iterative manner for each image at runtime, these approaches are often time consuming.

In this work, we propose a new method to leverage pre-trained GANs such as StyleGAN [9] and BigGAN [10] to provide rich and diverse priors for restoration. This is similar in spirit to the classic notion of dictionary [11], which explicitly constructs a finite image bank. But unlike the conventional method, we use pre-trained GANs as a latent image bank that is practically infinite, hence can serve as a much stronger prior. Compared with 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 perform restoration, therefore is more practical for applications that demand fast response.

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

- *K. C. K. Chan, X. Xu, and C. C. Loy are with S-Lab, Nanyang Technological University (NTU), Singapore (E-mail: chan0899@ntu.edu.sg, xiangyu.xu@ntu.edu.sg, ccloy@ntu.edu.sg).*
- *X. Wang is with Applied Research Center, Tencent PCG (Email: xintao.wang@outlook.com).*
- *J. Gu is with Tetras. AI. and Shanghai AI Laboratory (Email: gujinwei@tetras.ai).*
- *C. C. Loy is the corresponding author.*



(a) Low-Resolution (b) ESRGAN (c) PULSE (d) **GLEAN** (e) **LightGLEAN** (e) Ground-truth

Fig. 1. **Example of 16× super-resolution (SR)**. (a) The low-resolution input. (b) ESRGAN [1] trains the SR generator from scratch, often produces artifacts and unnatural textures. (c) PULSE [2] achieves more realistic textures through GAN inversion but fails to recover ground-truth structures. (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) Our lightweight model, LightGLEAN, achieves comparable performance while having significantly fewer parameters. (f) The ground-truth image.

as a latent bank in a succinct *encoder-bank-decoder* architecture. This novel architecture allows us to lift the burden of simultaneous learning both fidelity and detail generation in a typical encoder-decoder network since the latent bank already captures rich natural image priors. In addition, we show that it is pivotal to condition the bank by passing the convolutional features from the encoder to achieve high-fidelity results. We also design a multi-resolution framework for passing features to strengthen information flow from the latent bank to the decoder, further improving the results. We show the effectiveness of the proposed method in handling images with challenging poses and structures apart from the highly-ill-posed nature of the task. 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 or using more generic priors.

This work is an extension of our earlier conference version [12]. In comparison to the conference version, we have introduced a significant amount of new materials. **1)** Through extensive experiments on our original model GLEAN, we find that some modules can be safely removed without sacrificing the performance. Thus, we redesign GLEAN and propose a lightweight version – *LightGLEAN*. Remarkably, when compared to GLEAN, LightGLEAN consists of only 21% of parameters while achieving comparable performance, as shown in Fig. 1(e). **2)** In addition to 8× to 64× super-resolution, we consider more restoration tasks in this paper. First, we demonstrate the capability of GLEAN on restoring images degraded by complex and unknown real-world degradations. Second, we show that the natural image prior encapsulated in the latent bank is effective in not only super-resolution but also various restoration tasks such as colorization. **3)** We extend GLEAN towards the restoration task of generic images. Different from our conference version that requires different models to restore images of different object classes, we demonstrate the potential of GLEAN on multi-class restorations by employing BigGAN [10] as our generative latent bank, which allows class-conditioned image generation. With the use of the multi-class prior, perceptually convincing images of different objects can be restored with a single model.

2 Related Work

Image Super-Resolution. Many existing SR algorithms [13], [14], [15], [16], [17], [18], [19], [20] directly learn a mapping from low-resolution images to high-resolution images with a pixel-wise constraint (*e.g.*, ℓ_2 loss). While these methods achieve remarkable results in terms of PSNR, training solely with pixel-wise constraints often results in perceptually unconvincing outputs with severe over-smoothing artifacts [2], [4]. To alleviate the problem, GANs [1], [4], [21], [22] are employed to approximate the natural image manifold, yielding more photo-realistic results. For instance, SRGAN [4] adopts adversarial loss and perceptual loss [23] in addition to ℓ_2 loss, improving the visual quality of the outputs. 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 $4\times$) [24], [25], [26], [27]. Dahl *et al.* [24] propose a fully probabilistic pixel recursive network for upsampling extremely coarse images with an resolution 8×8. RFB-ESRGAN [26] builds upon ESRGAN and adopts multi-scale receptive field blocks for $16\times$ SR. VarSR [25] achieves $8\times$ SR by matching the latent distributions of LR and HR images to recover the missing details. Zhang *et al.* [27] 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 specializing 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 [28], facial landmarks [29], [30], and identity [31] have been studied. Our work goes beyond previous works by pushing the SR limit to $64\times$, a large magnification factor that is challenging due to its highly ill-posed nature, and by generalizing to more categories.

Face Restoration with Generative Prior. Several concurrent works adopt generative priors for blind face restoration. Specifically, GFP-GAN [32] employs StyleGAN2 trained on FFHQ dataset to provide facial priors, and incorporate a channel-split SFT [5] and a facial component loss. GPEN [33] also makes use of StyleGAN2 as a prior. The noise map in StyleGAN is replaced by a feature map learned from an encoder. Different from GFP-GAN and GPEN, GLEAN is not confined to face restoration. Instead, we focus on a generic framework that is applicable to a wide range of object categories and tasks. Moreover, unlike GFP-GAN and GPEN, which use the entire StyleGAN as a prior network, we carefully examine the use of StyleGAN in image restoration and find that using the entire StyleGAN

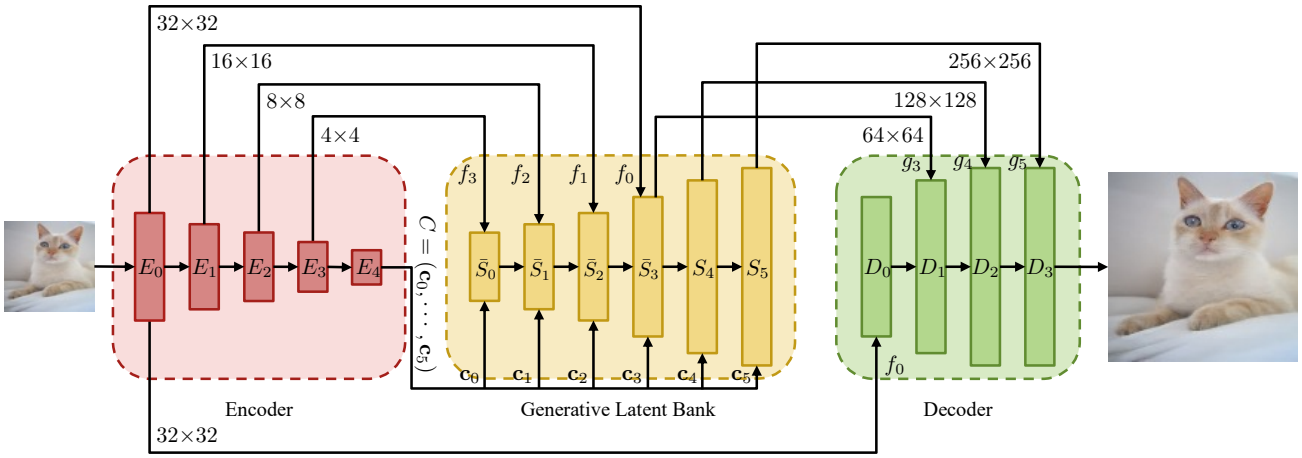


Fig. 2. **Overview of GLEAN**. In addition to the latent vectors $\mathbf{c}_i$, the generator (*i.e.*, the generative latent bank) is also conditioned on the multi-resolution features f_i. With a pre-trained GAN capturing the natural image prior, this encoder-bank-decoder design lifts the burden of 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.

for image restoration is unnecessary. Consequently, we redesign GLEAN by pruning a significant portion of it. Our lightweight model, LightGLEAN, achieves comparable performance while having only 21% of parameters and 35% of FLOPs.

Image Colorization. Existing works [34], [35], [36], [37] generally use pixel-wise loss between the output image and ground-truth image to guide the training of the network. With sufficient training data, learning-based methods are able to achieve promising results. However, it is well-known that pixel-wise losses such as ℓ_2 loss often lead to images with flat color and reduced perceptual quality. Later works [38], [39], [40] explore the possibility of generative adversarial networks to approximate the natural image manifold for better visual quality. Some studies attempt to employ additional priors such as object class [35], [36], [39] to improve the performance. In this work, we demonstrate a new way of exploiting prior information, particularly the prior captured in generative models for improving color restoration.

GAN Inversion. Given a degraded image x, GAN inversion-based methods [2], [6], [7], [8] in general produce a natural image best approximating x by optimizing $z^* = \text{argmin}_{z\in\mathcal{Z}} \mathcal{L}(G(z), x)$, where $\mathcal{Z}$ is the latent space and $\mathcal{L}(\cdot,\cdot)$ denotes the task-specific objective function. For instance, PULSE [2] iteratively optimizes the latent code of StyleGAN [9] with a pixel-wise constraint between the input and output. mGANprior [7] optimizes multiple latent codes to increase the expressiveness of the model. DGP [8] 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 retained 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 on 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.

3 GLEAN

A GAN model that is trained on large-scale natural images captures rich texture, color, and shape priors. Previous studies [2], [6], [7], [8] have shown that such priors can be harvested through GAN inversion to benefit various image restoration tasks. Nonetheless, methods for exploiting these priors without the costly optimization during inversion remain underexplored.

In this study, we devise GLEAN within a novel *encoder-bank-decoder* architecture, allowing one to exploit the generative priors with just a single forward pass. An overview of the architecture is depicted in Fig. 2. Given a degraded 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, further producing 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 the state-of-the-art GAN architectures as the generative latent bank, such as StyleGAN [9], [41] and BigGAN [10], while the specific choice is flexible, depending on different applications.

3.1 Encoder

To generate the latent vectors, we first use an RRDBNet [1] (denoted as E_0) to extract features f_0 from the input LR image. Then, we gradually reduce the resolution of the features by:

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

where E_i, $i \in \{1, \cdots, N\}$, 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:

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

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

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 GAN is originally designed for image generation tasks, it cannot be directly integrated into the proposed encoder-bank-decoder framework. In this work, we adapt the GAN architecture (*e.g.*, StyleGAN and BigGAN) to our framework 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 $C{=}(\mathbf{c}_0, \cdots, \mathbf{c}_{k-1})$ 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 [7], [42], [43].

2) To allow conditioning on the additional features from the encoder, we use an additional convolution in each style block for feature fusion for features whose resolution is smaller than or equal to the input resolution:

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

where S_i and $\bar{S}_i$ denote the original style block and the augmented style block with an additional convolution, respectively. g_i corresponds to the output feature of the i-th 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 the encoder.

Advantages. The use of generative latent bank is reminiscent of the task of reference-based restoration, where external HR information, such as single reference image [44], [45], [46], [47], [48], [49], multiple reference images [50], [51], [52] and learnable dictionary [50], is used. 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 [48] or component detection/selection [50] 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 the reference images/components selection.

3.3 Decoder

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

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

where D_i and d_i denote a 3×3 convolution and its output, respectively. Each convolution is followed by a pixel-shuffle [53] 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 better focus on the texture and detail generation.

3.4 Training

Similar to existing works [1], [4], [5], we adopt the standard MSE loss, perceptual loss [23], and adversarial loss for training. MSE loss is used to guide the fidelity of the output images:

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

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 [23] and adversarial loss [3] to improve the perceptual quality:

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

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

where $V(\cdot)$ denotes the feature embedding space of the VGG16 [54] network, and $\mathcal{D}$ corresponds to the StyleGAN or BigGAN discriminator. The resulting objective function is a weighted mean of the three losses:

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

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

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

To exploit the generative prior, we keep the weights of the latent bank fixed throughout training. This is because the latent bank may eventually be biased to the training distribution and can potentially harm the model generalizatbility. It is worth emphasizing that despite GLEAN being 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 a larger version of ESRGAN, named ESRGAN$^+$.

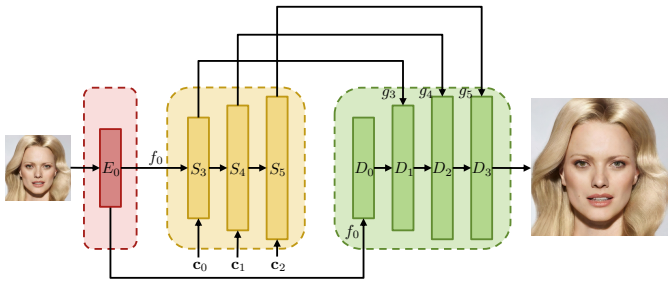


Fig. 3. **Overview of LightGLEAN**. Unlike GLEAN, which generates features with resolution down to 4×4, the latent bank in LightGLEAN directly conditions on the RRDB feature f_0, bypassing the style blocks that corresponds to the coarse resolutions. In addition, a fixed latent code $\mathbf{c}$ is used for all style blocks. In this design, LightGLEAN can be devised with much fewer learnable parameters.

TABLE 1
Complexity comparison between LightGLEAN and GLEAN. LightGLEAN contains 79% fewer parameters. Comparison is performed on the models for 64×64 input and 1024×1024 output.

	GLEAN	LightGLEAN	% Reduction
Encoder	137.3M	23.7M	82.7%
Generator	30.4M	10.9M	63.9%
Decoder	7.9M	1.7M	78.6%
Params	175.6M	36.3M	79.3%
FLOPs	277.5G	98.24G	64.6%

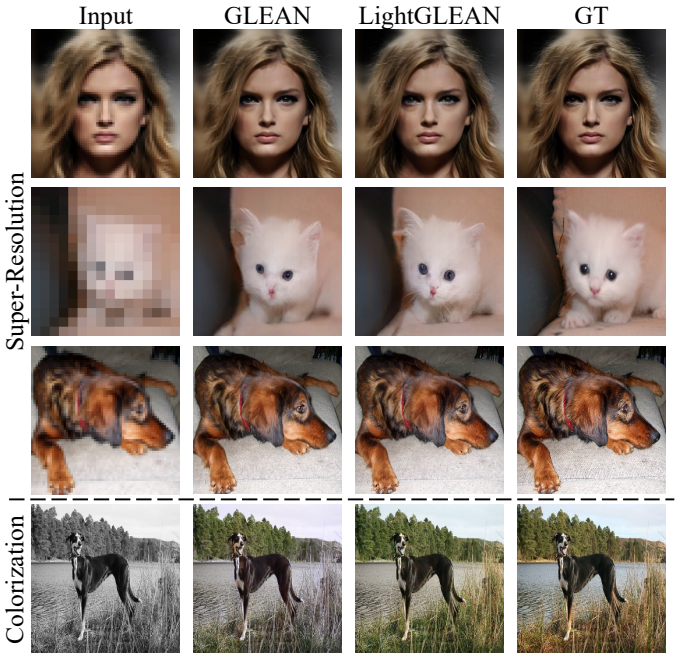


Fig. 4. **Qualitative comparison between GLEAN and LightGLEAN.** Despite being more lightweight than GLEAN, LightGLEAN provides outputs that are comparable to GLEAN. **(Zoom in for best view)**

4 LightGLEAN

Although it achieves remarkable performance, GLEAN has a large model size. To address this issue, we conduct an in-depth analysis of the design of GLEAN, such that nonessential modules can be identified and pruned.

As shown in Fig. 2, GLEAN adopts a multi-resolution structure, and the number of feature channels increases when resolution decreases. As a result, the encoder incurs a significant number of parameters in modules corresponding to the coarse features. While it is a common approach to synthesize high-quality images from a coarse resolution such as a 4×4 input (*e.g.*, StyleGAN and BigGAN), we find that the features with a resolution smaller than that of the input image are less important in the task of image restoration, since the feature f_0 extracted from the input image has already provided rich spatial and structural information for the subsequent process in the generative latent bank.

We propose the following two strategies to simplify the structure of GLEAN, and the resulting lightweight model, LightGLEAN, is shown in Fig. 3.

1) We remove the coarse feature connections between the encoder and the generator. Specifically, instead of generating coarse features and performing feature fusion, only E_0 is kept in the encoder of LightGLEAN. The encoder outputs only the RRDB feature f_0, and only f_0 is sent to the generator. For the generator, let i_0 be the style block index corresponding to the input resolution, the modules for the coarse resolutions (*i.e.*, $S_i, i\in\{i=1,\cdots,i_0-1\}$) are bypassed, and the feature fusion modules are removed. The encoder feature f_0 is directly used as an input to the style block for the finer resolutions. Symbolically, our latent bank is modified as follows:

$$g_i = \begin{cases} S_i(\mathbf{c}_i, f_0), & \text{if } i = i_0, \\ S_i(\mathbf{c}_i, g_{i-1}), & \text{if } i > i_0. \end{cases} \tag{10}$$

2) In LightGLEAN, the latent codes $\mathbf{c}_i$ are no longer learned from the encoder. Instead, they are casted as learnable parameters, and the same set of latent codes is applied for all input images. It further reduces the number of parameters as the linear layers are omitted.

By removing the coarse resolution modules that contribute to a significant portion of the parameters, a lightweight architecture is devised. Note that LightGLEAN can be further pruned by reducing the number of RRDBs [1] without significant performance drop[1]. Such exploration is left as our future work.

As shown in Table 1, LightGLEAN has only 21% of parameters when compared to GLEAN. Notably, the encoder of GLEAN consists of 137.3M parameters, making it hard to deploy in practice. In contrast, with our careful pruning, the encoder of LightGLEAN contains only 23.7M parameters, which is 82.7% fewer parameters than that of GLEAN. LightGLEAN achieves a comparable performance to GLEAN with 79% reduction of parameters and 65% reduction of FLOPs. The examples in Fig. 4 show that the output quality of LightGLEAN and GLEAN are comparable. More quantitative comparisons are discussed in the next section.

5 Experiments

5.1 Training Details

We adopt pre-trained StyleGAN[2] [9], StyleGAN2[3,4] [41], or BigGAN[5] [10] as our latent bank using the publicly available models and codes.

1. In the task of 16× face super-resolution, the PSNR merely drops by 0.02 dB when reducing the number of RRDBs from 23 to 10.
2. GenForce: https://github.com/genforce/genforce
3. BasicSR: https://github.com/xinntao/BasicSR
4. MMEditing [56]: https://github.com/open-mmlab/mmediting
5. https://github.com/ajbrock/BigGAN-PyTorch

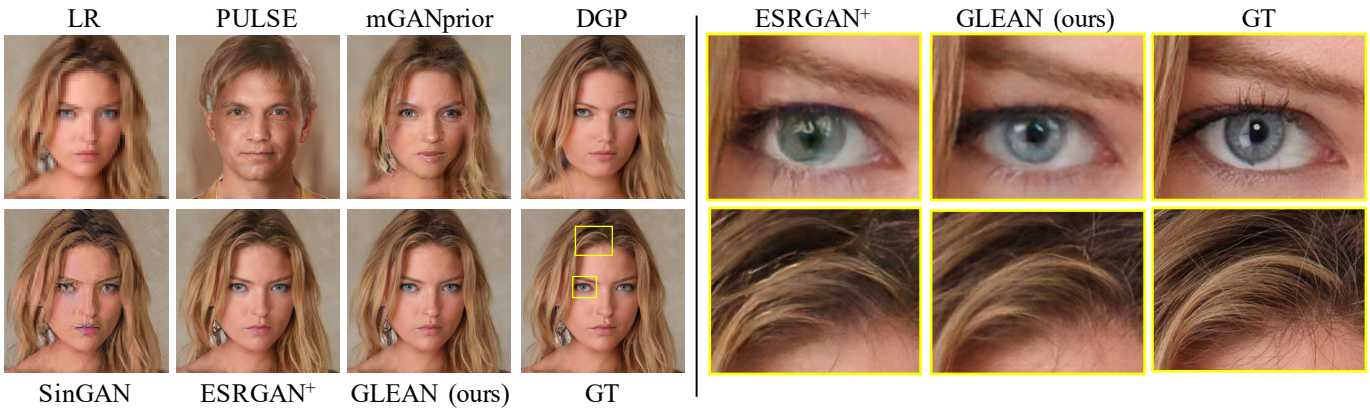


Fig. 5. **Comparisons on 16× SR on CelebA-HQ [55].** Only GLEAN is able to maintain high fidelity while synthesizing realistic textures and details: GAN inversion methods fail to preserve the identity, and adversarial loss methods struggle to synthesize fine details. ESRGAN+ denotes a larger version with similar runtime to GLEAN. **(Zoom in for best view)**

TABLE 2
Datasets used in our experiments.

	Train	Test
Human faces (Bicubic)	FFHQ [9]	CelebA-HQ [55]
Human faces (Blind)	FFHQ [9]	LFW [57], CelebA [58]
Cats	LSUN-train [59]	CAT [60]
Cars	LSUN-train [59]	Cars [61]
Bedrooms	LSUN-train [59]	LSUN-validate [59]
Towers	LSUN-train [59]	LSUN-validate [59]
Multi-class	ImageNet-train [62]	ImageNet-val [62]

We train and test GLEAN on various datasets. The training and test datasets used in our experiments are summarized in Table 2. For fair comparisons, we train the baselines on the same datasets as our model. The test set is strictly exclusive from training. Since StyleGAN and BigGAN produce images with a fixed size, we resize the images in the datasets for our experiments. The specific degradations are described in each section.

We adopt Cosine Annealing scheme [63] and Adam optimizer [64] in training. The number of iterations is 300K and the initial learning rate is 10^{-4}. The batch size is 8 for human faces, 16 for other class-specific training, and 32 for ImageNet training.

5.2 Class-Specific Super-Resolution

In this section, we assume the downsampling kernel is known, and we synthesize training and test data using the same degradation. Specifically, bicubic downsampling is used to synthesize LR-HR pairs.

The qualitative comparison on 16× SR is shown in Fig. 5. Since the GAN inversion methods are only guided by low-dimensional vectors and constraints in LR space, they are unable to maintain a good fidelity of the outputs. In particular, PULSE [2] and mGANprior [7] 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 [8] 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.

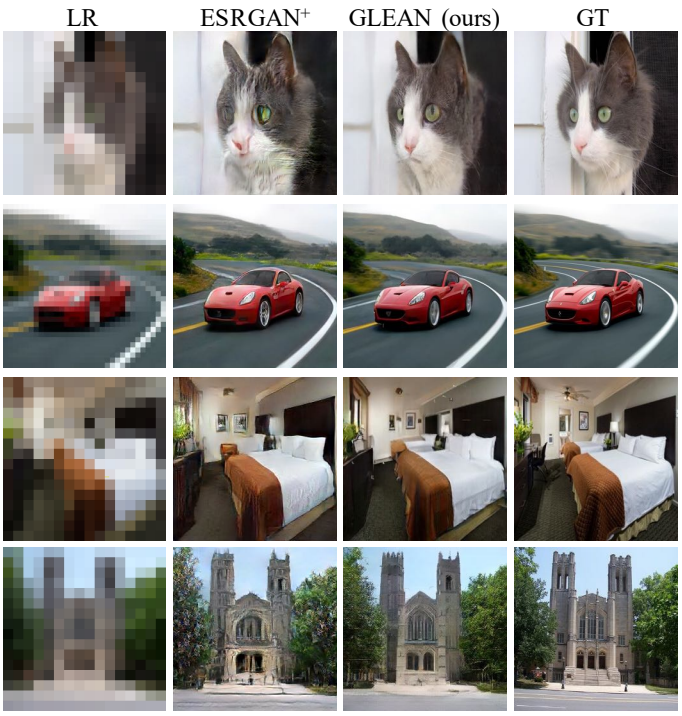


Fig. 6. **Results of 16× SR on other categories.** GLEAN can be applied to various categories by switching between StyleGANs trained on different categories. **(Zoom in for best view)**

Methods trained with adversarial loss (SinGAN [65], ESRGAN+[6] [1]) 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 fine details and introduces unnatural artifacts in detailed regions. It is worth emphasizing that although ESRGAN+ achieves competitive results on human faces, its perceptual quality on other categories such as *cats* and *cars* are less promising (see Fig. 1 and Fig. 6). With the latent bank providing natural image priors, GLEAN succeeds in both

6. A larger version of ESRGAN with similar runtime to GLEAN.

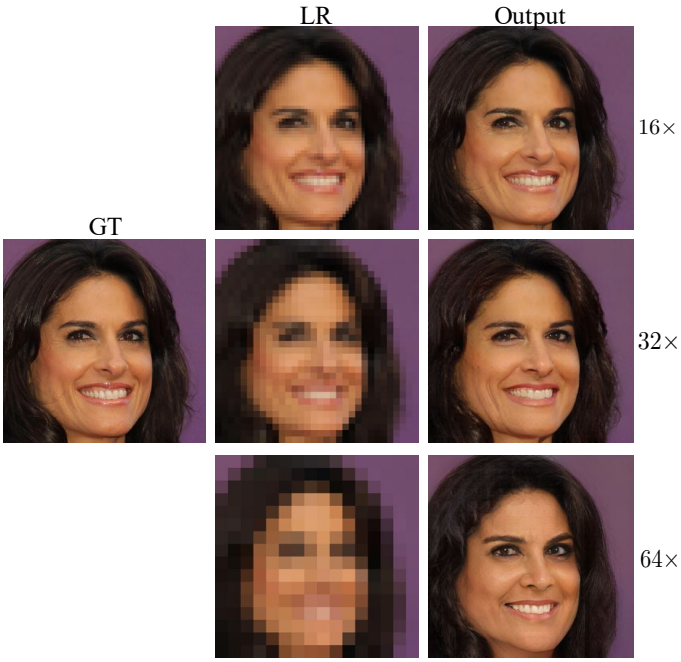


Fig. 7. **Results on larger scale factors.** GLEAN reconstructs realistic images highly similar to the GT for up to 64× upscaling factor. **(Zoom in for best view)**

Fig. 8. **Outputs with diverse poses and contents.** Despite GLEAN being 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 *(inlet image at the bottom left corner)*, but its outputs are significantly different from the GT when viewed in high resolution.

fidelity and naturalness. For example, when compared to ESRGAN$^+$, GLEAN reconstructs eyes with better shape and details. We further extend our method to larger scale factors in Fig. 7. GLEAN successfully generates realistic images resembling the ground truth up to 64× 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. 8, guided by the convolutional features, GLEAN is able to construct realistic non-aligned and non-human face images, even through the model was trained on aligned human faces. In contrast, the outputs of PULSE are biased to aligned human faces.

TABLE 3
Cosine similarity of ArcFace features [66] for 16× SR. GLEAN and LightGLEAN achieve a higher similarity than baselines. **Bolded** texts represent the best performance.

	Bicubic	PULSE [2]	mGANprior [7]	DGP [8]
Similarity	0.8939	0.4047	0.5526	0.7341
	SinGAN [65]	ESRGAN$^+$ [1]	**GLEAN**	**LightGLEAN**
Similarity	0.7718	0.9599	**0.9678**	0.9607

TABLE 4
Quantitative (PSNR/LPIPS) comparison on 16× SR. GLEAN outperforms other methods in most categories. **Bolded** texts represent the best performance.

	mGANprior [7]	PULSE [2]	ESRGAN$^+$ [1]	**GLEAN**	**LightGLEAN**
Face	23.66/0.4661	21.83/0.4600	26.76/0.2787	26.84/**0.2681**	**26.85**/0.2784
Cat	17.01/0.5556	19.78/0.5241	19.99/0.3482	**20.92**/**0.3215**	20.83/0.3243
Car	14.53/0.7228	16.30/0.6491	19.42/0.3006	**19.74**/**0.2830**	19.46/0.2887
Bedroom	16.38/0.5439	12.97/0.7131	**19.47**/**0.3291**	19.44/0.3310	19.37/0.3345
Tower	15.96/0.4870	13.62/0.7066	17.86/0.3132	**18.41**/**0.2850**	18.28/0.2933

TABLE 5
Complexity Comparison. GLEAN and LightGLEAN possess faster speeds with better performance. FLOPs of methods requiring test-time-training are not reported. **Bolded** texts represent the faster speed.

	Bicubic	PULSE [2]	mGANprior [7]	DGP [8]
Runtime	-	5s	7m	25m
Params	-	30.4M	30.4M	30.4M
FLOPs	-	-	-	-
	SinGAN [65]	ESRGAN$^+$ [1]	**GLEAN**	**LightGLEAN**
Runtime	42m	0.13s	0.12s	**0.10s**
Params	**1.03M**	23.97M	175.6M	36.3M
FLOPs	-	225.0G	277.5G	**98.24G**

Its outputs can only approximate the ground truth 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. 6.

Quantitative Comparison. To demonstrate the ability of GLEAN in producing outputs with high fidelity, we extract 100 images from CelebA-HQ [55] and compute the cosine similarity to the ground-truth on the ArcFace embedding space [66]. As shown in Table 3, both GLEAN and LightGLEAN achieve higher similarity than the baseline methods, validating the effectiveness of using generative latent bank in preserving intrinsic structure of the input space.

We additionally provide the quantitative comparison on different categories in Table 4. For each category, we select 100 images and compute their average PSNR and LPIPS [67]. 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. In addition, with our effective pruning, LightGLEAN matches the performance of GLEAN and outperforms existing works. Remarkably, LightGLEAN outperforms ESRGAN$^+$ with about 50% of FLOPs reduction.

Complexity Comparison. To demonstrate the efficiency of GLEAN and LightGLEAN, we compare their runtime, model size, and FLOPs on the task of 16× face super-resolution, using a V100 GPU. As shown Table 5, GLEAN and LightGLEAN have much faster speeds when compared to GAN-inversion methods and SinGAN. They also outperform ESRGAN$^+$ with comparable speed. Although Light-

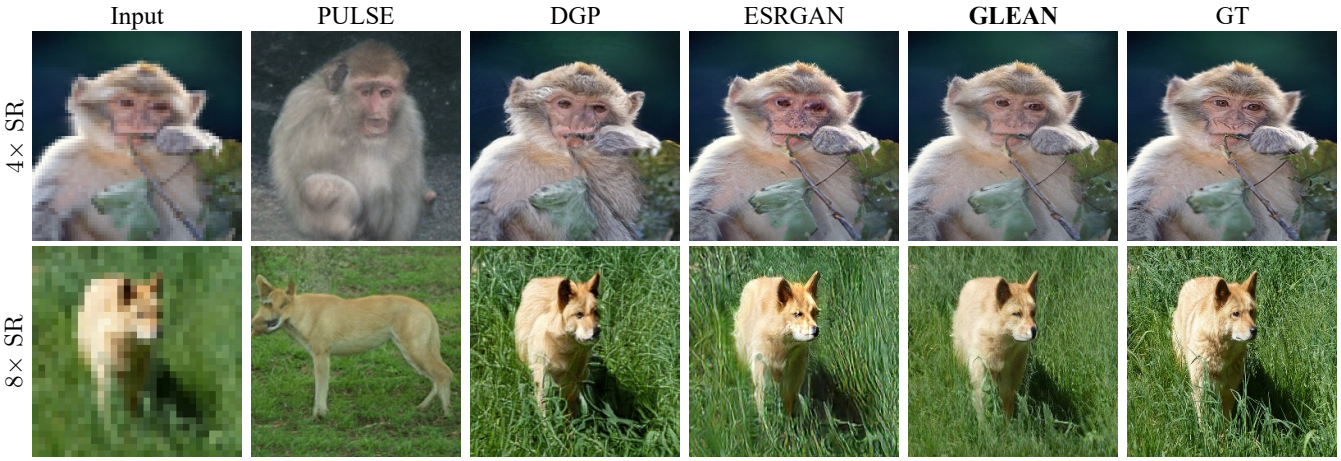


Fig. 9. **Results of Super-Resolution using BigGAN.** By employing the multi-class prior encapulated in BigGAN [10], GLEAN can be applied to multiple classes using a single model. GLEAN outperforms existing works in terms of both fidelity and quality. **(Zoom in for best view)**

Fig. 10. **Results of Image colorization.** In addition to super-resolution, GLEAN can be extended to other restoration tasks such as colorization. By employing the generative priors in BigGAN, GLEAN tends to produce natural color when compared to existing works. GLEAN is also applicable to real-world old photos.

GLEAN possesses a larger model size than existing methods', it has only 44% of FLOPs of ESRGAN's, and it does not require training during test time. Despite the large FLOPs improvement of LightGLEAN over the original GLEAN, we observe a minor improvement in runtime reduction due to the non-optimized code. The speed of LightGLEAN could be further improved with more engineering efforts.

Latent Bank Fintuning. Different from DGP [8] and GPEN [33], our experiments show that finetuning the latent bank does not lead to significant performance difference. The discrepancy results from the differnece in network designs: (1) DGP represents an image with only latent vectors, which possess limited representational power. Therefore, finetuning is necessary to overfit to the structure of the input image. In contrast, GLEAN uses additional convolutional features to guide the structures. (2) The encoder features of GPEN are used to replace the *random noise* in the original StyleGAN. Since the noise controls only fine-grained attributes before finetuning, it is expected that finetuning is needed to alter the contribution of the noise. In contrast, GLEAN did not attempt to modify the inputs to StyleGAN.

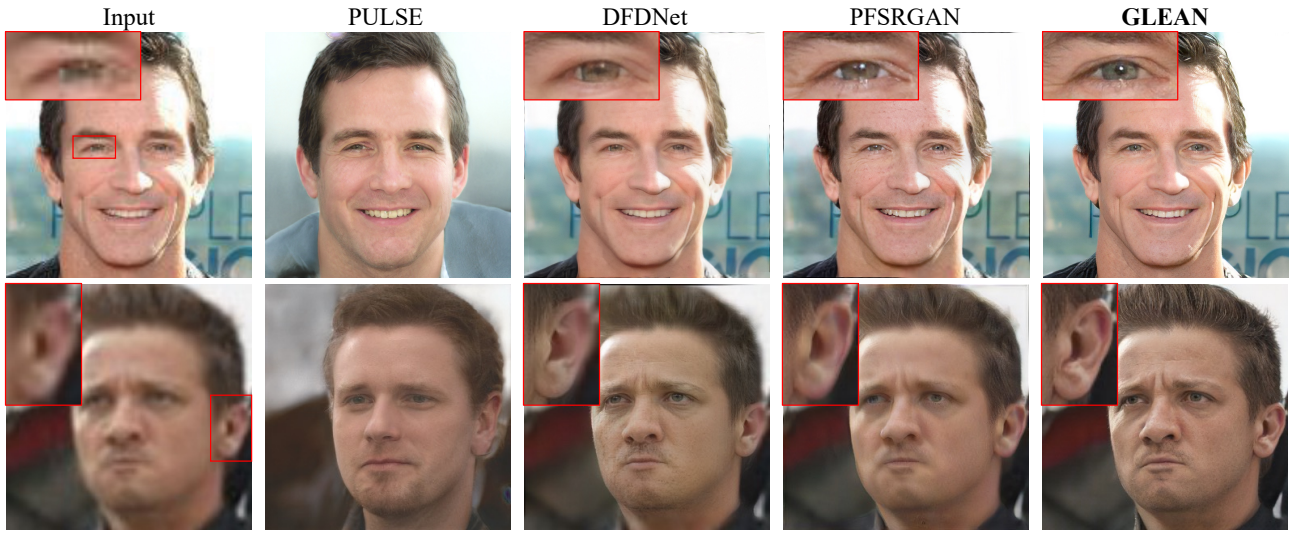


Fig. 11. **Results of real-world image restoration.** GLEAN can be extended to unknown degradations by applying various degradations during training. With the natural image prior encapsulated in our latent bank, GLEAN is able to produce results with high quality and natural textures, whereas existing methods produce outputs either with low fidelity, noticeable artifacts, or blurry details. **(Zoom in for best view)**

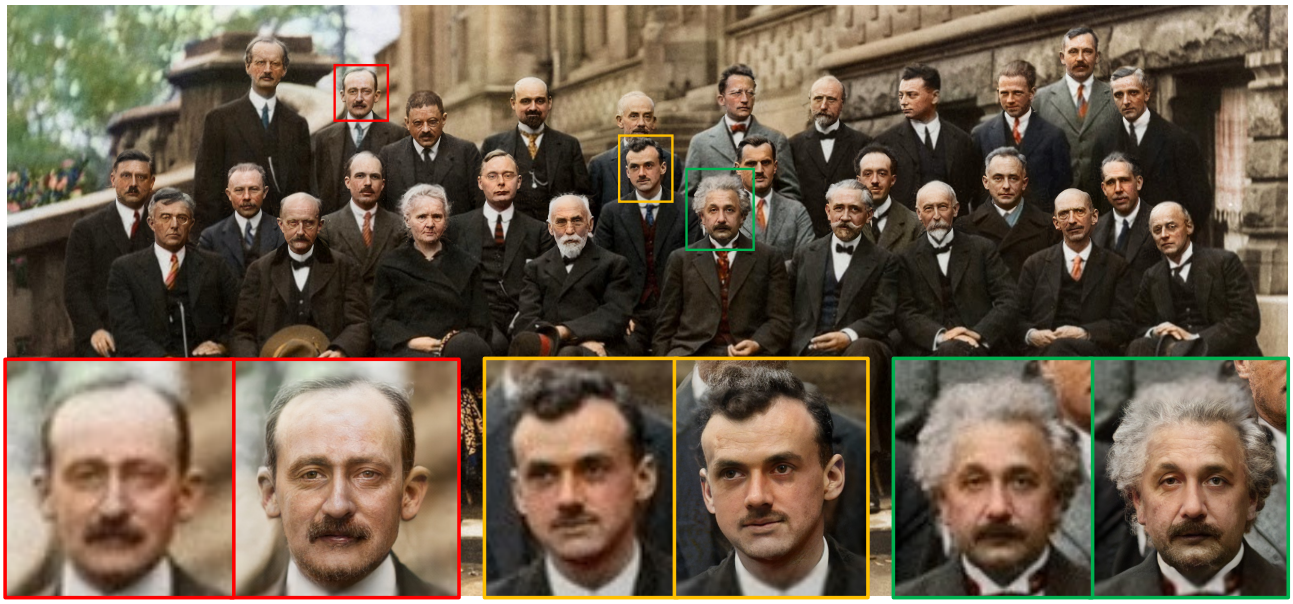


Fig. 12. **Results of real-world image restoration.** GLEAN is able to restore the fine details that are not captured in the input image. Image taken at the Solvay conference. Faces in the background image are replaced with restored outputs generated by GLEAN. **(Zoom in to view other faces in the group photo)**

5.3 Multi-Class Image Super-Resolution

In this work, we take one step towards generic image super-resolution by demonstrating the possiblity of leveraging multi-class prior in generative models for multi-class image super-resolution. We employ BigGAN in place of StyleGAN as the latent bank for a multi-class prior.

The quantitative comparisons to existing state of the arts are shown in Table 6, where we obtain the same conclusion in the class-specific image super-resolution. On the one hand, we see that GLEAN and LightGLEAN significantly outperform GAN-inversion methods including PULSE and DGP. On the other hand, by employing the generative prior, our two models also perform favorably against the feedforward model – ESRGAN. The performance gain is also reflected in the qualitative results depicted in Fig. 9, in which GLEAN produces results with much less artifacts and textures with much better quality. Notably, we observe a larger performance gap between ESRGAN$^+$ and GLEANs in the case of $8\times$ super-resolution, underscoring the significance of our generative prior.

5.4 Image Colorization

In addition to texture and shape priors that are useful for super-resolution, we hypothesize that various information such as color is captured in the generative model. Therefore, in this work, we extend our notion to a task orthogonal to super-resolution – colorization, to demonstrate the versatility of GLEAN. For this task, we also use the multi-class prior in BigGAN. The training scheme remains the same, and the network architecture is modified as follows:

TABLE 6
Quantitative (PSNR/LPIPS) comparison on ImageNet, using BigGAN as the latent bank. GLEAN outperforms other methods in most categories. **Bolded** texts represent the best performance.

	4x SR	8x SR
PULSE [2]	14.88/0.6923	14.80/0.6808
DGP [8]	20.56/0.2818	18.69/0.3904
ESRGAN+ [1]	22.89/0.1442	20.04/0.2628
GLEAN	23.12/**0.1239**	**20.62**/**0.2388**
LightGLEAN	**23.13**/0.1312	20.52/0.2541

	Colorization
DGP [8]	21.34/0.1950
InstColor. [35]	22.88/0.1807
DeOldify [38]	23.12/0.1609
GLEAN	**23.48**/**0.1469**
LightGLEAN	23.24/0.1534

TABLE 7
Quantitative (NIQE/FID/Identity similarity) comparison on real-world face image restoration. GLEAN outperforms existing state of the arts, producing outputs of high quality and fidelity. **Bolded** texts represent the best performance.

	LFW_a [57]	Celeb-A [58]
PULSE [2]	5.018/82.10/0.2509	3.709/83.02/0.3084
DFDNet [50]	8.878/82.35/0.8584	10.152/73.01/0.8780
PSFR-GAN [68]	5.999/69.87/0.8466	5.989/65.83/0.8945
GLEAN	3.943/**67.71**/**0.8949**	3.921/**61.65**/**0.9192**
LightGLEAN	**3.578**/68.67/0.8596	**3.635**/62.35/0.8774

1) The input is the luminance channel in the *Lab* color space, and the input channel of the first convolution is modified from 3 to 1.
2) The decoder is replaced with four convolutional layers, and the output channel of the last convolutional layer is 2. The output is then concatenated to the input luminance image.

More details about the architecture is provided in the supplementary material.

Performance. We compare the performance of the proposed method with representative methods including DGP [8], InstColorization [35], and DeOldify [38]. Table 6 and Fig. 10 demonstrate the superiority of GLEAN in comparison to existing methods. Following existing works [35], we employ PSNR and LPIPS as the evaluation metrics. From Table 6 we see that GLEAN achieves significant gains over existing works, and LightGLEAN is slightly inferior to GLEAN but achieves the second best performance. The qualitative comparison is shown in Fig. 10, in which GLEAN outperforms other methods, highlighting the model's capability in capturing color prior. More examples are provided in the supplementary material.

Discussion. Both existing works and GLEANs attempt to use various forms of priors to improve the perceptual quality of colorization results. In particular, DGP and DeOldify both adopt adversarial loss to better approximate the natural image manifold. InstColorization employs an off-the-shelf detection network to provide object prior. With the knowledge of the object category, the search space of the color could be reduced. Our method explores another use of prior. Through training for image generation, the network is required to understand the color distribution of the respective category in order to synthesize realistic objects. GLEAN exploits this information to synthesize realistic color for various objects.

5.5 Real-World Face Image Restoration

In this section, we demonstrate the capability of GLEAN for real-world face image restoration. In the original GLEAN, bicubic downsampling is added during training. As a result, GLEAN cannot be applied to real-world images with unknown degradations. To further improve the generalizability of GLEAN, we apply random degradations during training to mimic the complex degradations in reality. In this task, the ill-poseness increases significantly and the use of priors becomes more critical. We first describe the training and test settings, followed by the comparison with existing state of the arts. The training scheme remains unchanged.

Training Data. We train GLEAN on synthetic data that approximate real low-quality images. We follow the degradation process adopted in [50] for synthesizing the training data:

$$y = [(x \circledast k_\sigma)_{\downarrow_r} + n_\delta]_{\texttt{JPEG}_q}. \tag{11}$$

In other words, the high quality image x is first convolved with a Gaussian blur kernel k_σ followed by a downsampling operation with a scale factor r. After that, additive white Gaussian noise n_δ is added to the image, and finally the noisy image is compressed by JPEG with quality factor q. For each training pair, we randomly sample values of σ, r, δ, and q from the intervals $[0.2, 10]$, $[1, 8]$, $[0, 25]$, and $[5, 50]$, respectively.

Test Data and Metrics. We use two existing face image datasets for testing. LFW_a is a subset of the LFW dataset [57] whose elements have a surname starting with the letter "A". In addition, we select the first 500 images (images with IDs from 1 to 500) from Celeb-A [58] for test. Since no ground-truth images are available, we use NIQE [69], FID [70], and ArcFace similarity [66] as the evaluation metrics.

Qualitative Comparison. The qualitative comparison is shown in Fig. 11. By optimizing only the low-dimensional vectors, PULSE [2] successfully reconstructs natural images, but the outputs are dissimilar to the ground truths. When compared with DFDNet [50] and PFSRGAN [68], which are the current state-of-the-art blind face image restoration methods, GLEAN is able to produce more details and hence images with better quality.

Quantitative Comparison. The quantitative comparison is shown in Table 7. First, we compare the image naturalness using NIQE and FID. GLEAN and LightGLEAN outperform existing methods in these metrics, demonstrating the superiority of these two methods in the generation of high-quality images. Second, we compare the image fidelity using the ArcFace similarity. We observe that the outputs of PULSE has a significantly worse similarity, indicating that PULSE severely alters the identity of the inputs. Our models achieve either similar to or better similarity in both datasets, confirming its effectiveness in preserving identity.

To further demonstrate its effectiveness on real-world face restoration, we apply GLEAN on a group photo taken at the Solvay conference. As illustrated in Fig. 12, GLEAN successfully restores the fine details that are not captured in the original images.

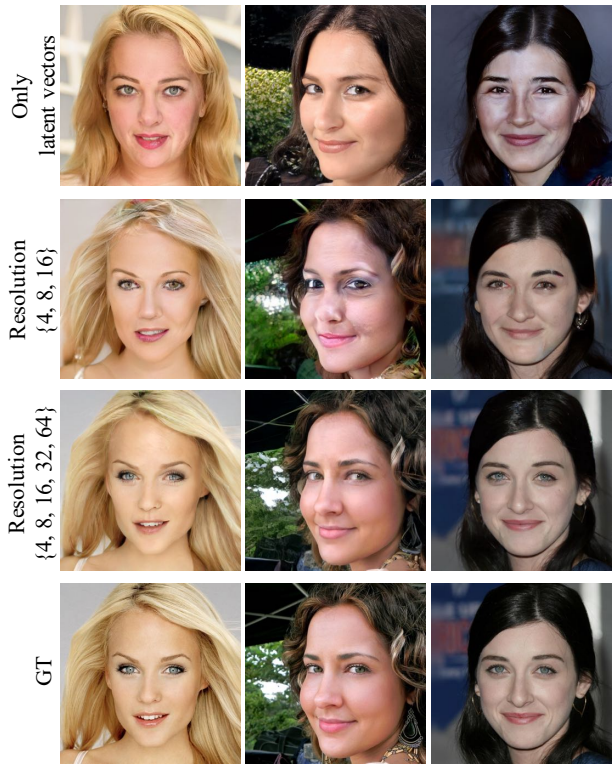


Fig. 13. **Effects of the multi-resolution encoder features.** Without the convolutional features, the outputs can only resemble the global attributes (*e.g.*, hair color, pose). When adding the encoder features progressively, the network can capture more local structures, better approximating the GT.

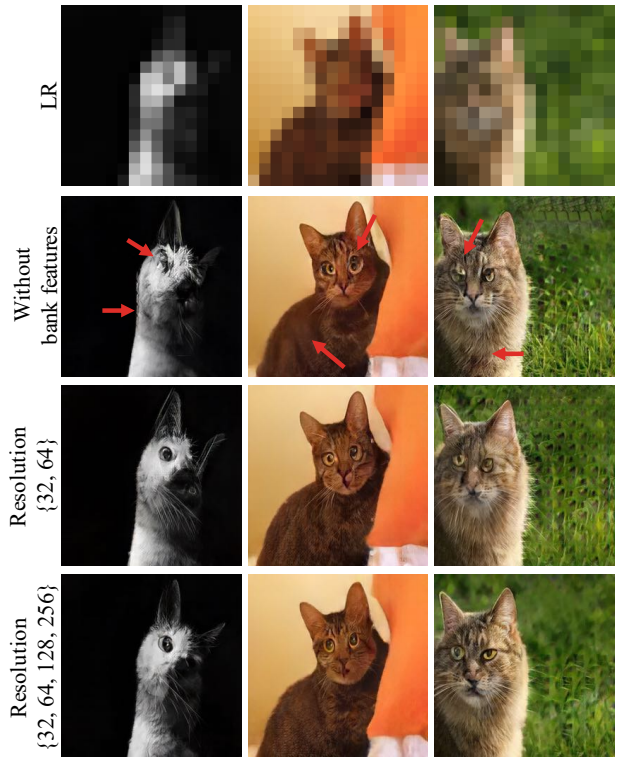


Fig. 14. **Effects of the latent bank features.** The rich texture priors captured in the generator lift the burden of the encoder in texture generation. Improvements on both texture and structures are observed when finer features are inserted into the decoder. **(Zoom in for best view)**

6 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 latent bank as conditions. To remove the effects brought on by the decoder, we test with a variant of GLEAN where the generator directly produces the output images. The comparison is depicted in Fig. 13.

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, 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 approximations of 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 on 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. 14. 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.

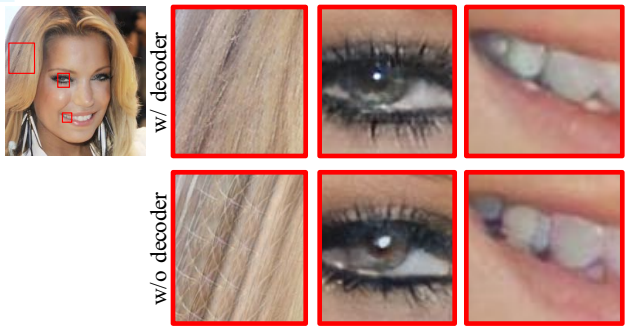
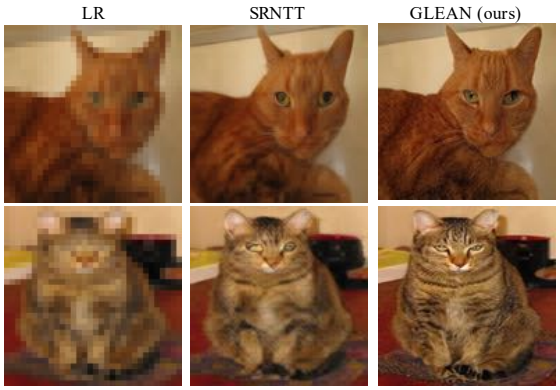


Fig. 15. **Contributions of the decoder.** The decoder reinforces the spatial information captured in the encoder features and aggregate them in a coarse-to-fine manner, resulting in an enhanced quality.

Importance of Decoder. As shown in Fig. 15, without the decoder, despite being perceptually convincing overall, the

(a) Comparison with DFDNet [50]



(b) Comparison with SRNTT [48]

Fig. 16. **Comparison with 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 faithfully produce fur textures.

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 that adopt an imagery dictionary for SR – DFDNet [50] and SRNTT [48]. Examples are shown in Fig. 16.

For DFDNet, we evaluate the performance on LR images with unknown degradations[7]. Through pre-constructing a dictionary of facial components (*e.g.*, eyes, lips), DFDNet shows remarkable performance on face restoration. However, it cannot faithfully produce results on parts absent in the dictionary, such as skin and hair. Therefore, significant incoherence is observed in the outputs. Despite GLEAN being trained on the bicubic kernel, it is still capable of producing visually appealing outputs. More importantly, GLEAN is not confined to improving the visual quality of specific components. Instead, the entire image is super-

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

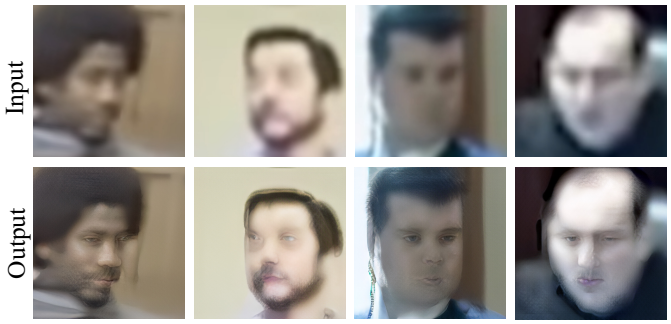


Fig. 17. **Failure cases on extreme degradations.** In extreme cases that contain severe degrdations, GLEAN cannot restore extreme cases of severe degradations. This problem can potentially be solved by adopting heavier degradations.

resolved, leading to coherent and attractive results. We foresee GLEAN to achieve even better performance 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 the specific imagery clues, GLEAN does not rely on explicit textural transferal procedure. This enables its 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 higher resolutions, where global matching is computationally prohibitive.

7 Discussion and Conclusion

We have presented a new way to exploit pre-trained GANs for various image restoration tasks including super-resolution, colorization, and hybrid restoration. We have shown that a pre-trained GAN can be used as a generative latent bank in an encoder-bank-decoder architecture. We have also presented a lightweight version of GLEAN, named LightGLEAN, and demonstrated the potential of these two methods on generic images by employing multi-class prior. Both GLEAN and LightGLEAN outperform existing state of the arts in terms of fidelity and quality.

Despite obtaining satisfactory results in various class domains, there are a few limitations in GLEANs. First, in spite of the efforts made in this work towards generic restoration, the existing version of GLEANs are still confined to *finite classes* and *fixed resolution* encompassed by the GAN prior, due to the lack of powerful generic image synthesizer. We believe that with a stronger generative model that can synthesize realistic images on more diverse scenes, the performance of GLEANs and relevant ideas could be markedly improved. Second, in the task of real-world face restoration, although they work well on cases of mild to moderate degradations, they fail to produce pleasant outputs in cases of heavy degradations, as shown in Fig. 17. To adapt to such complex and heavy degradations, we believe that incorporating heavier degradations during training could

partially remedy the situation. In addition, more sophisticated data augmentation schemes, such as second-order degradations [71] and degradation-shuffling [72], could be taken into consideration. In the future, with more sophisticated generative models, we believe that the notion of generative latent bank can be used for more restoration tasks as well as more diverse scenes. This idea can potentially be extended to various forms of priors such as language priors.

Acknowledgement. This study is supported under the RIE2020 Industry Alignment Fund Industry Collaboration Projects (IAF-ICP) Funding Initiative, as well as cash and in-kind contribution from the industry partner(s). It is also partly supported by the NTU NAP grant.

REFERENCES

[1] X. Wang, K. Yu, S. Wu, J. Gu, Y. Liu, C. Dong, C. C. Loy, Y. Qiao, and X. Tang, "ESRGAN: Enhanced super-resolution generative adversarial networks," in *ECCVW*, 2018. 1, 2, 3, 4, 5, 6, 7, 10

[2] S. Menon, A. Damian, S. Hu, N. Ravi, and C. Rudin, "PULSE: Self-supervised photo upsampling via latent space exploration of generative models," in *CVPR*, 2020. 1, 2, 3, 6, 7, 10

[3] I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio, "Generative adversarial nets," in *NeurIPS*, 2014. 1, 4

[4] C. Ledig, L. Theis, F. Huszár, J. Caballero, A. Cunningham, A. Acosta, A. P. Aitken, A. Tejani, J. Totz, Z. Wang *et al.*, "Photo-realistic single image super-resolution using a generative adversarial network," in *CVPR*, 2017. 1, 2, 4

[5] X. Wang, K. Yu, C. Dong, and C. C. Loy, "Recovering realistic texture in image super-resolution by deep spatial feature transform," in *CVPR*, 2018. 1, 2, 4

[6] D. Bau, H. Strobelt, W. Peebles, B. Zhou, J.-Y. Zhu, A. Torralba *et al.*, "Semantic photo manipulation with a generative image prior," *TOG*, 2020. 1, 3

[7] J. Gu, Y. Shen, and B. Zhou, "Image processing using multi-code GAN prior," *CVPR*, 2020. 1, 3, 4, 6, 7

[8] X. Pan, X. Zhan, B. Dai, D. Lin, C. C. Loy, and P. Luo, "Exploiting deep generative prior for versatile image restoration and manipulation," in *ECCV*, 2020. 1, 3, 6, 7, 8, 10

[9] T. Karras, S. Laine, and T. Aila, "A style-based generator architecture for generative adversarial networks," in *CVPR*, 2019. 1, 3, 5, 6

[10] A. Brock, J. Donahue, and K. Simonyan, "Large scale GAN training for high fidelity natural image synthesis," in *ICLR*, 2018. 1, 2, 3, 5, 8

[11] J. Yang, J. Wright, T. S. Huang, and Y. Ma, "Image super-resolution via sparse representation," *TIP*, 2010. 1

[12] K. C. Chan, X. Wang, X. Xu, J. Gu, and C. C. Loy, "GLEAN: Generative latent bank for large-factor image super-resolution," in *CVPR*, 2021. 2

[13] T. Dai, J. Cai, Y. Zhang, S.-T. Xia, and L. Zhang, "Second-order attention network for single image super-resolution," in *CVPR*, 2019. 2

[14] C. Dong, C. C. Loy, K. He, and X. Tang, "Learning a deep convolutional network for image super-resolution," in *ECCV*, 2014. 2

[15] ——, "Image super-resolution using deep convolutional networks," *TPAMI*, vol. 38, no. 2, pp. 295–307, 2016. 2

[16] C. Dong, C. C. Loy, and X. Tang, "Accelerating the super-resolution convolutional neural network," in *ECCV*. Springer, 2016, pp. 391–407. 2

[17] X. He, Z. Mo, P. Wang, Y. Liu, M. Yang, and J. Cheng, "ODE-inspired network design for single image super-resolution," in *CVPR*, 2019. 2

[18] X. Xu, Y. Ma, and W. Sun, "Towards real scene super-resolution with raw images," in *CVPR*, 2019. 2

[19] Y. Zhang, K. Li, K. Li, L. Wang, B. Zhong, and Y. Fu, "Image super-resolution using very deep residual channel attention networks," in *ECCV*, 2018. 2

[20] S. Zhou, J. Zhang, W. Zuo, and C. C. Loy, "Cross-scale internal graph neural network for image super-resolution," in *NeurIPS*, 2020. 2

[21] M. S. Sajjadi, B. Schölkopf, and M. Hirsch, "EnhanceNet: Single image super-resolution through automated texture synthesis," in *ICCV*, 2017. 2

[22] X. Xu, D. Sun, J. Pan, Y. Zhang, H. Pfister, and M.-H. Yang, "Learning to super-resolve blurry face and text images," in *ICCV*, 2017. 2

[23] J. Johnson, A. Alahi, and F.-F. Li, "Perceptual losses for real-time style transfer and super-resolution," in *ECCV*. Springer, 2017. 2, 4

[24] R. Dahl, M. Norouzi, and J. Shlens, "Pixel recursive super resolution," in *ICCV*, 2017. 2

[25] S. Hyun and J.-P. Heo, "VarSR: Variational super-resolution network for very low resolution images," in *ECCV*, 2020. 2

[26] T. Shang, Q. Dai, S. Zhu, T. Yang, and Y. Guo, "Perceptual extreme super resolution network with receptive field block," in *CVPRW*, 2020. 2

[27] Y. Zhang, Z. Zhang, S. DiVerdi, Z. Wang, J. Echevarria, and Y. Fu, "Texture hallucination for large-scale painting super-resolution," in *ECCV*, 2020. 2

[28] M. Li, Y. Sun, Z. Zhang, H. Xie, and J. Yu, "Deep learning face hallucination via attributes transfer and enhancement," in *ICME*, 2019. 2

[29] D. Kim, M. Kim, G. Kwon, and D.-S. Kim, "Progressive face super-resolution via attention to facial landmark," in *BMVC*, 2019. 2

[30] C. Ma, Z. Jiang, Y. Rao, J. Lu, and J. Zhou, "Deep face super-resolution with iterative collaboration between attentive recovery and landmark estimation," in *CVPR*, 2020. 2

[31] K. Grm, W. J. Scheirer, and V. Štruc, "Face hallucination using cascaded super-resolution and identity priors," *TIP*, 2019. 2

[32] X. Wang, Y. Li, H. Zhang, and Y. Shan, "Towards real-world blind face restoration with generative facial prior," in *CVPR*, 2021. 2

[33] T. Yang, P. Ren, X. Xie, and L. Zhang, "GAN prior embedded network for blind face restoration in the wild," in *CVPR*, 2021. 2, 8

[34] Z. Cheng, Q. Yang, and B. Sheng, "Deep colorization," in *ICCV*, 2015, pp. 415–423. 3

[35] J.-W. Su, H.-K. Chu, and J.-B. Huang, "Instance-aware image colorization," in *CVPR*, 2020. 3, 10

[36] S. Iizuka, E. Simo-Serra, and H. Ishikawa, "Let there be color! joint end-to-end learning of global and local image priors for automatic image colorization with simultaneous classification," in *TOG*, 2016. 3

[37] R. Zhang, P. Isola, and A. A. Efros, "Colorful image colorization," in *ECCV*, 2016. 3

[38] J. Antic, "Deoldify: https://github.com/jantic/deoldify." 3, 10

[39] P. Vitoria, L. Raad, and C. Ballester, "ChromaGAN: Adversarial picture colorization with semantic class distribution," in *WACV*. 3

[40] P. Isola, J.-Y. Zhu, T. Zhou, and A. A. Efros, "Image-to-image translation with conditional adversarial networks," *arXiv preprint arXiv:1611.07004*, 2017. 3

[41] T. Karras, S. Laine, M. Aittala, J. Hellsten, J. Lehtinen, and T. Aila, "Analyzing and improving the image quality of StyleGAN," in *CVPR*, 2020. 3, 5

[42] Y. Xu, Y. Shen, J. Zhu, C. Yang, and B. Zhou, "Generative hierarchical features from synthesizing images," in *CVPR*, 2021. 4

[43] J. Zhu, Y. Shen, D. Zhao, and B. Zhou, "In-domain GAN inversion for real image editing," in *ECCV*, 2020. 4

[44] G. Shim, J. Park, and I. S. Kweon, "Robust reference-based super-resolution with similarity-aware deformable convolution," in *CVPR*, 2020. 4

[45] Y. Xie, J. Xiao, M. Sun, C. Yao, and K. Huang, "Feature representation matters: End-to-end learning for reference-based image super-resolution," in *ECCV*, 2020. 4

[46] F. Yang, H. Yang, J. Fu, H. Lu, and B. Guo, "Learning texture transformer network for image super-resolution," in *CVPR*, 2020. 4

[47] Y. Zhang, I. W. Tsang, Y. Luo, C. Hu, X. Lu, and X. Yu, "Copy and Paste GAN: Face hallucination from shaded thumbnails," in *CVPR*, 2020. 4

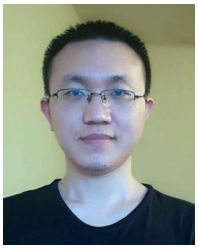
[48] Z. Zhang, Z. Wang, Z. Lin, and H. Qi, "Image super-resolution by neural texture transfer," in *CVPR*, 2019. 4, 12

[49] H. Zheng, M. Ji, H. Wang, Y. Liu, and L. Fang, "CrossNet: An end-to-end reference-based super resolution network using cross-scale warping," in *ECCV*, 2018. 4

[50] X. Li, C. Chen, S. Zhou, X. Lin, W. Zuo, and L. Zhang, "Blind face restoration via deep multi-scale component dictionaries," in *ECCV*, 2020. 4, 10, 12

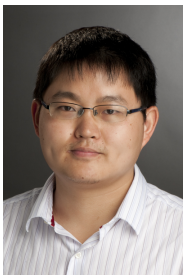
[51] X. Li, W. Li, D. Ren, H. Zhang, M. Wang, and W. Zuo, "Enhanced blind face restoration with multi-exemplar images and adaptive spatial feature fusion," in *CVPR*, 2020. 4
[52] X. Yan, W. Zhao, K. Yuan, R. Zhang, Z. Li, and S. Cui, "Towards content-independent multi-reference super-resolution: Adaptive pattern matching and feature aggregation," in *ECCV*, 2020. 4
[53] W. Shi, J. Caballero, F. Huszár, J. Totz, A. P. Aitken, R. Bishop, D. Rueckert, and Z. Wang, "Real-time single image and video super-resolution using an efficient sub-pixel convolutional neural network," in *CVPR*, 2016. 4
[54] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *ICLR*, 2015. 4
[55] T. Karras, T. Ailo, S. Laine, and J. Lehtinen, "Progressive growing of GANs for improved quality, stability, and variation," in *ICLR*, 2018. 6, 7
[56] MMEditing Contributors, "MMEditing: OpenMMLab Image and Video Editing Toolbox," 2022. [Online]. Available: https://github.com/open-mmlab/mmediting 5
[57] G. B. Huang, M. Mattar, T. Berg, and E. Learned-Miller, "Labeled faces in the wild: A database forstudying face recognition in unconstrained environments," in *Workshop on Faces in 'Real-Life' Images: Detection, Alignment, and Recognition*, 2008. 6, 10
[58] Z. Liu, P. Luo, X. Wang, and X. Tang, "Deep learning face attributes in the wild," in *ICCV*, 2015. 6, 10
[59] F. Yu, A. Seff, Y. Zhang, S. Song, T. Funkhouser, and J. Xiao, "LSUN: Construction of a large-scale image dataset using deep learning with humans in the loop," *arXiv preprint arXiv:1506.03365*, 2015. 6
[60] W. Zhang, J. Sun, and X. Tang, "Cat head detection - how to effectively exploit shape and texture features," in *ECCV*, 2008. 6
[61] J. Krause, M. Stark, J. Deng, and F.-F. Li, "3D object representations for fine-grained categorization," in *ICCV*, 2013. 6
[62] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and F.-F. Li, "ImageNet: A large-scale hierarchical image database," in *CVPR*, 2009. 6
[63] I. Loshchilov and F. Hutter, "SGDR: Stochastic gradient descent with warm restarts," in *ICLR*, 2017. 6
[64] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015. 6
[65] T. R. Shaham, T. Dekel, and T. Michaeli, "SinGAN: Learning a generative model from a single natural image," in *ICCV*, 2019. 6, 7
[66] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, "ArcFace: Additive angular margin loss for deep face ..." in *CVPR*, 2019. 7, 10
[67] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang, "The unreasonable effectiveness of deep features as a perceptual metric," in *CVPR*, 2018. 7
[68] C. Chen, X. Li, L. Yang, X. Lin, L. Zhang, and K.-Y. K. Wong, "Progressive semantic-aware style transformation for blind face restoration," in *CVPR*, 2021. 10
[69] A. Mittal, R. Soundararajan, and A. C. Bovik, "Making a "completely blind" image quality analyzer," *IEEE Signal Processing Letters*, 2013. 10
[70] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs trained by a two time-scale update rule converge to a local nash equilibrium," in *NeurIPS*, 2017. 10
[71] X. Wang, L. Xie, C. Dong, and Y. Shan, "Real-ESRGAN: Training real-world blind super-resolutionwith pure synthetic data," *arXiv preprint arXiv:2107.10833*, 2021. 13
[72] K. Zhang, J. Liang, L. Van Gool, and R. Timofte, "Designing a practical degradation model for deep blind image super-resolution," in *ICCV*, 2021. 13

Kelvin C.K. Chan is currently a fourth-year PhD student at S-Lab, Nanyang Technological University. He received his MPhil degree in Mathematics as well as his BSc and BEng degrees from The Chinese University of Hong Kong. He was awarded the Google PhD Fellowship in 2021. He won the first place in multiple international challenges including NTIRE2019 and NTIRE2021, and was selected as an outstanding reviewer in ICCV 2021. His research interests include low-level vision, especially image and video restoration.



Xiangyu Xu is a research fellow at S-Lab, Nanyang Technological University. He was a postdoc fellow in the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA from 2019 to 2020, and a research scientist at SenseTime, Beijing, China from 2018 to 2019. He got the Ph.D. degree in the Department of Electronic Engineering, Tsinghua University in 2018. Before that, he received the B.Eng degree in the Department of Electronic Engineering, Tsinghua University in 2013, and the B.Ec degree in the National School of Development, Peking University in 2015. He was a visiting Ph.D. student at University of California, Merced and Harvard University. His research interest includes image processing, computer vision, and machine learning.

Xintao Wang is currently a researcher in Applied Research Center (ARC), Tencent PCG. He received his Ph.D. degree in the Department of Information Engineering, The Chinese University of Hong Kong, in 2020. He was selected as an outstanding reviewer in CVPR 2019 and an outstanding reviewer (honorable mention) in BMVC 2019. He won the first place in several international super-resolution challenges including NTIRE2019, NTIRE2018, and PIRM2018. His research interests focus on low-level vision problems, including super-resolution, image and video restoration.



Jinwei Gu (Senior Member, IEEE) is the R&D Executive Director of SenseTime USA. His current research focuses on low-level computer vision, computational photography, smart visual sensing and perception, and robotics. He obtained his PhD degree in 2010 from Columbia University, and the B.S and M.S. from Tsinghua University, in 2002 and 2005 respectively. Before joining SenseTime, he was a senior research scientist in NVIDIA Research from 2015 to 2018. Prior to that, he was an assistant professor in Rochester Institute of Technology from 2010 to 2013, and a senior researcher in the media lab of Futurewei Technologies from 2013 to 2015. He serves as an associate editor for IEEE Transactions on Computational Imaging and IEEE Transactions on Pattern Analysis and Machine Intelligence. He is an IEEE Senior Member since 2018.

Chen Change Loy (Senior Member, IEEE) is an Associate Professor with the School of Computer Science and Engineering, Nanyang Technological University, Singapore. He is also an Adjunct Associate Professor at The Chinese University of Hong Kong. He received his Ph.D. (2010) in Computer Science from the Queen Mary University of London. Prior to joining NTU, he served as a Research Assistant Professor at the MMLab of The Chinese University of Hong Kong, from 2013 to 2018. He was a postdoctoral researcher at Queen Mary University of London and Vision Semantics Limited, from 2010 to 2013. He serves as an Associate Editor of the IEEE Transactions on Pattern Analysis and Machine Intelligence and International Journal of Computer Vision. He also serves/served as an Area Chair of major conferences such as ICCV 2021, CVPR 2021, CVPR 2019, and ECCV 2018. He is a senior member of IEEE. His research interests include image/video restoration and enhancement, generative tasks, and representation learning.

APPENDIX

We will first discuss the change of the architecture we have made for the task of colorization in Sec. A. We will then provide additional qualitative results of GLEAN and LightGLEAN in Sec. B.

APPENDIX A
ARCHITECTURE FOR COLORIZATION

Unlike super-resolution, in the task of colorization, the resolution of the input image is the same as that of the output image. Therefore, we made minimal modifications to the architecture so that GLEANs can be adapted to the case when input and output have the same resolution. The network architecture is modified as follows:

1) The input is the luminance channel in the *Lab* color space, and the input channel of the first convolution is modified from 3 to 1.
2) The decoder is replaced with four convolution layers, and the output channel of the last convolutional layer is 2. The output is then concatenated to the input luminance image.

We find that this simple modification suffices to achieve good colorization results. Overall, the architectures for different tasks are highly similar, demonstrating the versatility of GLEAN and LightGLEAN.

APPENDIX B
QUALITATIVE RESULTS

In this section, we demonstrate additional qualitative results on 1) multi-class image super-resolution, 2) image colorization, and 3) real-world face image restoration. From Fig. 18 to Fig. 22 we observed that with the generative priors encapsulated in our latent bank, our methods are able to produce faithful results despite the highly ill-posed nature of the tasks.

Fig. 18. More results of GLEAN and LightGLEAN on $4\times$ multi-class image super-resolution. With the powerful priors, GLEAN and LightGLEAN are able to produce realistic details. **(Zoom in for best view)**

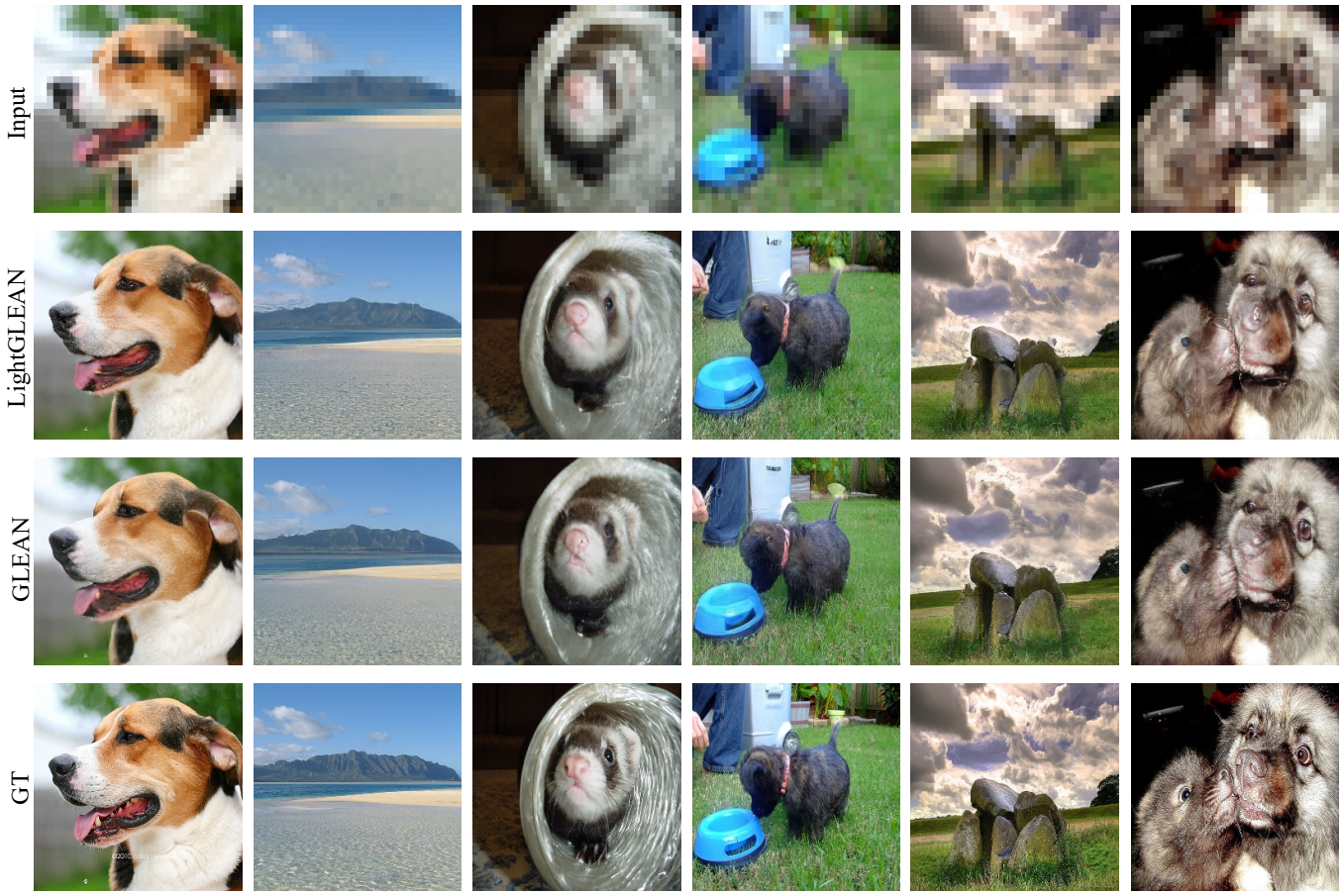


Fig. 19. More results of GLEAN and LightGLEAN on $8\times$ multi-class image super-resolution. Despite the input image is only 32×32, our proposed models are able to synthesize realistic textures. **(Zoom in for best view)**

Fig. 20. More results of GLEAN and LightGLEAN on multi-class image colorization. Although the color is sometimes dissimilar to the ground-truths, our models are able to produce natural color.

Fig. 21. More results of GLEAN and LightGLEAN on multi-class image colorization. Although the color is sometimes dissimilar to the ground-truths, our models are able to produce natural color.

Fig. 22. More results of GLEAN and LightGLEAN on real-world face image restoration. Our models are able to restore real-world face images, which contain diverse and unknown degradations. Note that the ground-truth is unavailable for real-world images. **(Zoom in for best view)**