Invertible Image Rescaling

Mingqing Xiao[1*], Shuxin Zheng[2], Chang Liu[2], Yaolong Wang[3*], Di He[1], Guolin Ke[2], Jiang Bian[2], Zhouchen Lin[1], and Tie-Yan Liu[2]

[1] Peking University
[2] Microsoft Research Asia
[3] Toronto University
{mingqing_xiao, di_he, zlin}@pku.edu.cn, {shuz, changliu, guoke, jiabia, tyliu}@microsoft.com, yaolong.wang@mail.utoronto.ca

Abstract. High-resolution digital images are usually downscaled to fit various display screens or save the cost of storage and bandwidth, meanwhile the post-upscaling is adpoted to recover the original resolutions or the details in the zoom-in images. However, typical image downscaling is a non-injective mapping due to the loss of high-frequency information, which leads to the ill-posed problem of the inverse upscaling procedure and poses great challenges for recovering details from the downscaled low-resolution images. Simply upscaling with image super-resolution methods results in unsatisfactory recovering performance. In this work, we propose to solve this problem by modeling the downscaling and upscaling processes from a new perspective, i.e. an invertible bijective transformation, which can largely mitigate the ill-posed nature of image upscaling. We develop an Invertible Rescaling Net (IRN) with deliberately designed framework and objectives to produce visually-pleasing low-resolution images and meanwhile capture the distribution of the lost information using a latent variable following a specified distribution in the downscaling process. In this way, upscaling is made tractable by inversely passing a randomly-drawn latent variable with the low-resolution image through the network. Experimental results demonstrate the significant improvement of our model over existing methods in terms of both quantitative and qualitative evaluations of image upscaling reconstruction from downscaled images.

1 Introduction

With exploding amounts of high-resolution (HR) images/videos on the Internet, image downscaling is quite indispensable for storing, transferring and sharing such large-sized data, as the downscaled counterpart can significantly save the storage, efficiently utilize the bandwidth [12,37,54,48,34] and easily fit for screens with different resolution while maintaining visually valid information [26,49]. Meanwhile, many of these downscaling scenarios inevitably raise a great demand for the inverse task, i.e., upscaling the downscaled image to a higher resolution or its original size [56,57,46,19]. However, details are lost and distortions appear when users zoom in or upscale the low-resolution (LR)

* Work done during an internship at Microsoft Research Asia.

images. Such an upscaling task is quite challenging since image downscaling is well-known as a non-injective mapping, meaning that there could exist multiple possible HR images resulting in the same downscaled LR image. Hence, this inverse task is usually considered to be ill-posed [24,55,17].

Many efforts have been made to mitigate this ill-posed problem, but the gains fail to meet the expectation. For example, most of previous works choose super-resolution (SR) methods to upscale the downscaled LR images. However, mainstream SR algorithms [17,36,60,59,14,50] focus only on recovering HR images from LR ones under the guidance of a predefined and non-adjustable downscaling kernel (e.g., Bicubic interpolation), which omits its compatibility to the downscaling operation. Intuitively, as long as the target LR image is pre-downscaled from an HR image, taking the image downscaling method into consideration would be quite invaluable for recovering the high-quality upscaled image.

Instead of simply treating the image downscaling and upscaling as two separate and independent tasks, most recently, there have been efforts [26,34,49] attempting to model image downscaling and upscaling as a united task by an encoder-decoder framework. Specifically, they proposed to use an upscaling-optimal downscaling method as an encoder which is jointly trained with an upscaling decoder [26] or existing SR modules [34,49]. Although such an integrated training approach can significantly improve the quality of the HR images recovered from the corresponding downscaled LR images, neither can we do a perfect reconstruction. These efforts didn't tackle much on the ill-posedness since they link the two processes only through the training objectives and conduct no attempt to capture any feature of the lost information.

In this paper, with inspiration from the reciprocal nature of this pair of image rescaling tasks, we propose a novel method to largely mitigate this ill-posed problem of the image upscaling. According to the Nyquist-Shannon sampling theorem, high-frequency contents are lost during downscaling. Ideally, we hope to keep all lost information to perfectly recover the original HR image, but storing or transferring the high-frequency information is unacceptable. In order to well address this challenge, we develop a novel invertible model called Invertible Rescaling Net (IRN) which captures some knowledge on the lost information in the form of its distribution and embeds it into model's parameters to mitigate the ill-posedness. Given an HR image x, IRN not only downscales it into a visually-pleasing LR image y, but also embed the case-specific high-frequency content into an auxiliary case-agnostic latent variable z, whose marginal distribution obeys a fixed pre-specified distribution (e.g., isotropic Gaussian). Based on this model, we use a randomly drawn sample of z from the pre-specified distribution for the inverse upscaling procedure , which holds the most information that one could have in upscaling.

Yet, there are still several great challenges needed to be addressed during the IRN training process. Specifically, it is essential to ensure the quality of reconstructed HR images, obtain visually pleasing downscaled LR ones, and accomplish the upscaling with a case-agnostic z, i.e., $z \sim p(z)$ instead of a case-specific $z \sim p(z|y)$. To this end, we design a novel compact and effective objective function by combining three respective components: an HR reconstruction loss, an LR guidance loss and a distribution matching loss. The last component is for the model to capture the true HR image mani-

fold as well as for enforcing z to be case-agnostic. Neither the conventional adversarial training techniques of generative adversarial nets (GANs) [21] nor the maximum likelihood estimation (MLE) method for existing invertible neural networks [15,16,29,4] could achieve our goal, since the model distribution doesn't exist here, meanwhile these methods don't guide the distribution in the latent space. Instead, we take the pushed-forward empirical distribution of x as the distribution on y, which, in independent company with $p(z)$, is the actually used distribution to inversely pass our model to recover the distribution of x. We thus match this distribution with the empirical distribution of x (the data distribution). Moreover, due to the invertible nature of our model, we show that once this matching task is accomplished, the matching task in the (y, z) space is also solved, and z is made case-agnostic. We minimize the JS divergence to match the distributions, since the alternative sample-based maximum mean discrepancy (MMD) method [3] doesn't generalize well to the high dimension data in our task.

Our contributions are concluded as follows:

- To our best knowledge, the proposed IRN is the first attempt to model image downscaling and upscaling, a pair of mutually-inverse tasks, using an invertible (i.e., bijective) transformation. Powered by the deliberately designed invertibility, our proposed IRN can largely mitigate the ill-posed nature of image upscaling reconstruction from the downscaled LR image.
- We propose a novel model design and efficient training objectives for IRN to enforce the latent variable z, with embedded lost high-frequency information in the downscaling direction, to obey a simple case-agnostic distribution. This enables efficient upscaling based on the valuable samples of z drawn from the certain distribution.
- The proposed IRN can significantly boost the performance of upscaling reconstruction from downscaled LR images compared with state-of-the-art downscaling-SR and encoder-decoder methods. Moreover, the amount of parameters of IRN is significantly reduced, which indicates the light-weight and high-efficiency of the new IRN model.

2 Related Work

2.1 Image Upscaling after Downscaling

Super resolution (SR) is a widely-used image upscaling method and get promising results in low-resolution (LR) image upscaling task. Therefore, SR methods could be used to upscale downscaled images. Since the SR task is inherently ill-posed, previous SR works mainly focus on learning strong prior information by example-based strategy [18,20,46,27] or deep learning models [17,36,60,59,14,50]. However, if the targeted LR image is pre-downscaled from the corresponding high-resolution image, taking the image downscaling method into consideration would significantly help the upscaling reconstruction.

Traditional image downscaling approaches employ frequency-based kernels, such as Bilinear, Bicubic, etc. [41], as a low-pass filter to sub-sample the input HR images into target resolution. Normally, these methods suffer from resulting over-smoothed images since the high-frequency details are suppressed. Therefore, several detail-preserving

or structurally similar downscaling methods [31,42,51,52,38] are proposed recently. Besides those perceptual-oriented downscaling methods, inspired by the potentially mutual reinforcement between downscaling and its inverse task, upscaling, increasing efforts have been focused on the upscaling-optimal downscaling methods, which aim to learn a downscaling model that is optimal to the post-upscaling operation. For instance, Kim *et al.* [26] proposed a task-aware downscaling model based on an auto-encoder framework, in which the encoder and decoder act as the downscaling and upscaling model, respectively, such that the downscaling and upscaling processes are trained jointly as a united task. Similarly, Li *et al.* [34] proposed to use a CNN to estimate downscaled compact-resolution images and leverage a learned or specified SR model for HR image reconstruction. More recently, Sun *et al.* [49] proposed a new content-adaptive-resampler based image downscaling method, which can be jointly trained with any existing differentiable upscaling (SR) models. Although these attempts have an effect of pushing one of downscaling and upscaling to resemble the inverse process of the other, they still suffer from the ill-posed nature of image upscaling problem. In this paper, we propose to model the downscaling and upscaling processes by leveraging the invertible neural networks.

Difference from SR. Note that image upscaling is a different task from super-resolution. In our scenario, the ground-truth HR image is available at the beginning but somehow we have to discard it and store/transmit the LR version instead. We hope that we can recover the HR image afterwards using the LR image. While for SR, the real HR is unavailable in applications and the task is to generate new HR images for LR ones.

2.2 Invertible Neural Network

The invertible neural network (INN) [15,16,29,32,22,8,13] is a popular choice for generative models, in which the generative process $x = f_\theta(z)$ given a latent variable z can be specified by an INN architecture f_θ. The direct access to the inverse mapping $z = f_\theta^{-1}(x)$ makes inference much cheaper. As it is possible to compute the density of the model distribution in INN explicitly, one can use the maximum likelihood method for training. Due to such flexibility, INN architectures are also used for many variational inference tasks [44,30,10].

INN is composed of invertible blocks. In this study, we employ the invertible architecture in [16]. For the l-th block, input h^l is split into h_1^l and h_2^l along the channel axis, and they undergo the additive affine transformations [15]:

$$\begin{aligned} h_1^{l+1} &= h_1^l + \phi(h_2^l), \\ h_2^{l+1} &= h_2^l + \eta(h_1^{l+1}), \end{aligned} \tag{1}$$

where ϕ, η are arbitrary functions. The corresponding output is $[h_1^{l+1}, h_2^{l+1}]$. Given the output, its inverse transformation is easily computed:

$$\begin{aligned} h_2^l &= h_2^{l+1} - \eta(h_1^{l+1}), \\ h_1^l &= h_1^{l+1} - \phi(h_2^l), \end{aligned} \tag{2}$$

To enhance the transformation ability, the identity branch is often augmented [16]:

$$\begin{aligned} h_1^{l+1} &= h_1^l \odot \exp(\psi(h_2^l)) + \phi(h_2^l), \\ h_2^{l+1} &= h_2^l \odot \exp(\rho(h_1^{l+1})) + \eta(h_1^{l+1}), \\ h_2^l &= (h_2^{l+1} - \eta(h_1^{l+1})) \odot \exp(-\rho(h_1^{l+1})), \\ h_1^l &= (h_1^{l+1} - \phi(h_2^l)) \odot \exp(-\psi(h_2^l)). \end{aligned} \tag{3}$$

Some prior works studied using INN for paired data (x, y). Ardizzone *et al.* [3] analyzed real-world problems from medicine and astrophysics. Compared to their tasks, image downscaling and upscaling bring more difficulties because of notably larger dimensionality, so that their losses do not work for our task. In addition, the ground-truth LR image y does not exist in our task. Guided image generation and colorization using INN is proposed in [4] where the invertible modeling between x and z is conditioned on a guidance y. The model cannot generate y given x thus is unsuitable for the image upscaling task. INN is also applied to the image-to-image translation task [43] where the paired domain (X, Y) instead of paired data is considered, thus is again not the case of image upscaling.

2.3 Image Compression

Image compression is a type of data compression applied to digital images, to reduce their cost for storage or transmission. Image compression may be lossy (e.g., JPEG, BPG) or lossless (e.g., PNG, BMP). Recently, deep learning based image compression methods [6,45,7,2,40] show promising results on both visual effect and compression ratio. However, the resolution of image won't be changed by compression, which means there is no visually meaningful low-resolution image but only bit-stream after compressing. Thus our task can't be served by image compression methods.

3 Methods

3.1 Model Specification

The sketch of our modeling framework is presented in Fig. 1. As explained in Introduction, we mitigate the ill-posed problem of the upscaling task by modeling the distribution of lost information during downscaling. We note that according to the Nyquist-Shannon sampling theorem [47], the lost information during downscaling an HR image amounts to high-frequency contents. Thus we firstly employ a wavelet transformation to decompose the HR image x into low and high-frequency component, denote as x_L and x_H respectively. Since the case-specific high-frequency information will be lost after downscaling, in order to best recover the original x as possible in the upscaling procedure, we use an invertible neural network to produce the visually-pleasing LR image y meanwhile model the distribution of the lost information by introducing an auxiliary latent variable z. In contrast to the case-specific x_H (i.e., $x_H \sim p(x_H|x_L)$), we force z to be case-agnostic (i.e., $z \sim p(z)$) and obey a simple specified distribution, e.g., an isotropic Gaussian distribution. In this way, there is no further need to preserve either

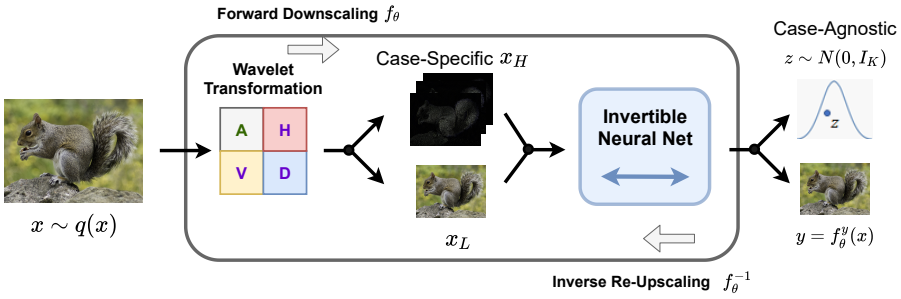


Fig. 1. Illustration of the problem formulation. In the forward downscaling procedure, HR image x is transformed to visually pleasing LR image y and case-agnostic latent variable z through a parameterized invertible function $f_\theta(\cdot)$; in the inverse upscaling procedure, a randomly drawn z combined with LR image y are transformed to HR image through the inverse function $f_\theta^{-1}(\cdot)$.

x_H or z after downscaling, and z can be randomly sampled in the upscaling procedure, which is used to reconstruct x combined with LR image y by inversely passing the model.

3.2 Invertible Architecture

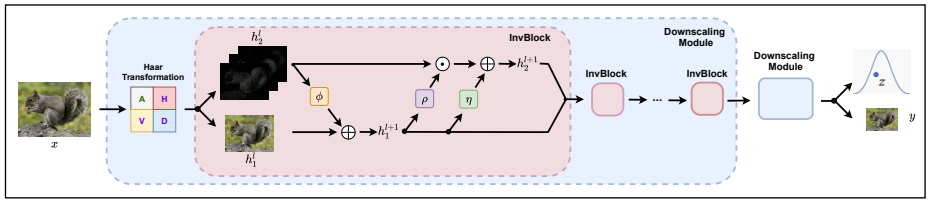


Fig. 2. Illustration of our framework. The invertible architecture is composed of Downscaling Modules, in which InvBlocks are stacked after a Haar Transformation. Each Downscaling Module reduces the spatial resolution by $2\times$. The $\exp(\cdot)$ of ρ is omit.

The general architecture of our proposed IRN is composed of stacked *Downscaling Modules*, each of which contains one *Haar Transformation* block and several invertible neural network blocks (*InvBlocks*), as illustrated in Fig. 2. We will show later that both of them are invertible, and thus the entire IRN model is invertible accordingly.

The Haar Transformation We design the model to contain certain inductive bias, which can efficiently learn to decompose x into the downscaled image y and case-agnostic high-frequency information embedded in z. To achieve this, we apply the Haar Transformation as the first layer in each downscaling module, which can explicitly decompose the input images into an approximate low-pass representation, and three directions of high-frequency coefficients [53][35][4]. More concretely, the Haar Transformation transforms the input raw images or a group of feature maps with height H, width W and channel C into a tensor of shape $(\frac{1}{2}H, \frac{1}{2}W, 4C)$. The first C slices of the

output tensor are effectively produced by an average pooling, which is approximately a low-pass representation equivalent to the Bilinear interpolation downsampling. The rest three groups of C slices contain residual components in the vertical, horizontal and diagonal directions respectively, which are the high-frequency information in the original HR image. By such a transformation, the low and high-frequency information are effectively separated and will be fed into the following InvBlocks.

InvBlock Taking the feature maps after the Haar Transformation as input, a stack of InvBlocks is used to further abstract the LR and latent representations. We leverage the general coupling layer architecture proposed in [15,16], i.e. Eqs. (1,3).

Utilizing the coupling layer is based on our considerations that (1) the input has already been split into low and high-frequency components by the Haar transformation; (2) we want the two branches of the output of a coupling layer to further polish the low and high-frequency inputs for a suitable LR image appearance and an independent and properly distributed latent representation of the high-frequency contents. So we match the low and high-frequency components respectively to the split of h_1^l, h_2^l in Eq. (1). Furthermore, as the shortcut connection is proved to be important in the image scaling tasks [36,50], we employ the additive transformation (Eq. 1) for the low-frequency part h_1^l, and the enhanced affine transformation (Eq. 3) for the high-frequency part h_2^l to increase the model capacity, as shown in Fig. 2.

Note that the transformation functions $\phi(\cdot), \eta(\cdot), \rho(\cdot)$ in Fig. 2 can be arbitrary. Here we employ a densely connected convolutional block, which is referred as Dense Block in [50] and demonstrated for its effectiveness of image upscaling task. Function $\rho(\cdot)$ is further followed by a centered sigmoid function and a scale term to prevent numerical explosion due to the $\exp(\cdot)$ function. Note that Figure 2 omits the $\exp(\cdot)$ in function ρ.

Quantization To save the output images of IRN as common image storage format such as RGB (8 bits for each R, G and B color channels), a quantization module is adopted which converts floating-point values of produced LR images to 8-bit unsigned int. We simply use rounding operation as the quantization module, store our output LR images by PNG format and use it in the upscaling procedure. There is one obstacle should be noted that the quantization module is nondifferentiable. To ensure that IRN can be optimized during training, we use Straight-Through Estimator [9] on the quantization module when calculating the gradients.

3.3 Training Objectives

Based on Section 3.1, our approach for invertible downscaling constructs a model that specifies a correspondence between HR image x and LR image y, as well as a case-agnostic distribution $p(z)$ of z. The goal of training is to drive these modeled relations and quantities to match our desiderata and HR image data $\{x^{(n)}\}_{n=1}^N$. This includes three specific goals, as detailed below.

LR Guidance Although the invertible downscaling task does not pose direct requirements on the produced LR images, we do hope that they are valid visually pleasing LR images. To achieve this, we utilize the widely acknowledged Bicubic method [41] to guide the downscaling process of our model. Let $y_{\text{guide}}^{(n)}$ be the LR image corresponding to $x^{(n)}$ that is produced by the Bicubic method. To make our model follow the guidance,

we drive the model-produced LR image $f_\theta^y(x^{(n)})$ to resemble $y_{\text{guide}}^{(n)}$:

$$L_{\text{guide}}(\theta) := \sum_{n=1}^{N} \ell_{\mathcal{Y}}(y_{\text{guide}}^{(n)}, f_\theta^y(x^{(n)})), \tag{4}$$

where $\ell_{\mathcal{Y}}$ is a difference metric on $\mathcal{Y}$, e.g., the L_1 or L_2 loss. We call it the LR guidance loss. This practice has also been adopted in the literature [26,49].

HR Reconstruction Although f_θ is invertible, it is not for the correspondence between x and y when z is not transmitted. We hope that for a specific downscaled LR image y, the original HR image can be restored by the model using any sample of z from the case-agnostic $p(z)$. Inversely, this also encourages the forward process to produce a disentangled representation of z from y. As described in Section 3.1, given a HR image $x^{(n)}$, the model-downscaled LR image $f_\theta^y(x^{(n)})$ is to be upscaled by the model as $f_\theta^{-1}(f_\theta^y(x^{(n)}), z)$ with a randomly drawn $z \sim p(z)$. The reconstructed HR image should match the original one $x^{(n)}$, so we minimize the expected difference and traverse over all the HR images:

$$L_{\text{recon}}(\theta) := \sum_{n=1}^{N} \mathbb{E}_{p(z)}[\ell_{\mathcal{X}}(x^{(n)}, f_\theta^{-1}(f_\theta^y(x^{(n)}), z))], \tag{5}$$

where $\ell_{\mathcal{X}}$ measures the difference between the original image and the reconstructed one. We call $L_{\text{recon}}(\theta)$ the HR reconstruction loss. For practical minimization, we estimate the expectation w.r.t. z by one random draw from $p(z)$ for each evaluation.

Distribution Matching The third part of the training goal is to encourage the model to catch the data distribution $q(x)$ of HR images, demonstrated by its sample cloud $\{x^{(n)}\}_{n=1}^N$. Recall that the model reconstructs a HR image $x^{(n)}$ by $f_\theta^{-1}(y^{(n)}, z^{(n)})$, where $y^{(n)} := f_\theta^y(x^{(n)})$ is the model-downscaled LR image, and $z^{(n)} \sim p(z)$ is the randomly drawn latent variable. When traversing over the sample cloud of true HR images $\{x^{(n)}\}_{n=1}^N$, $\{y^{(n)}\}_{n=1}^N$ also form a sample cloud of a distribution. We denote this distribution with the push-forward notation as ${f_\theta^y}_\#[q(x)]$, which represents the distribution of the transformed random variable $f_\theta^y(x)$ where the original random variable x obeys distribution $q(x)$, $x \sim q(x)$. Similarly, the sample cloud $\{f_\theta^{-1}(y^{(n)}, z^{(n)})\}_{n=1}^N$ represents the distribution of model-reconstructed HR images, and we denote it as ${f_\theta^{-1}}_\#\big[{f_\theta^y}_\#[q(x)]\ p(z)\big]$ since $(y^{(n)}, z^{(n)}) \sim {f_\theta^y}_\#[q(x)]\ p(z)$ (note that $y^{(n)}$ and $z^{(n)}$ are independent due to the generation process). The desideratum of distribution matching is to drive the model-reconstructed distribution towards data distribution, which can be achieved by minimizing their difference measured by some metric of distributions:

$$L_{\text{distr}}(\theta) := L_{\mathcal{P}}\big({f_\theta^{-1}}_\#\big[{f_\theta^y}_\#[q(x)]\ p(z)\big], q(x)\big). \tag{6}$$

The distribution matching loss pushes the model-reconstructed HR images to lie on the manifold of true HR images so as to make the recovered images appear more realistic. It also drives the case-independence of z from y in the forward process. To see this, we note that if f_θ is invertible, then in the asymptotic case, the two distributions match on $\mathcal{X}$, i.e., ${f_\theta^{-1}}_\#\big[{f_\theta^y}_\#[q(x)]\ p(z)\big] = q(x)$, if and only if they match on $\mathcal{Y} \times \mathcal{Z}$, i.e., ${f_\theta^y}_\#[q(x)]\ p(z) = {f_\theta}_\#[q(x)]$. The loss thus drives the coupled distribution

$f_{\theta\,\#}[q(x)] = (f_\theta^y, f_\theta^z)_\#[q(x)]$ of (y, z) from the forward process towards the decoupled distribution ${f_\theta^y}_\#[q(x)]\ p(z)$. Neither effect can be fully guaranteed by the reconstruction and guidance losses.

As mentioned in Introduction, the minimization is generally hard since both distributions are high-dimensional and have unknown density function. We employ the JS divergence as the probability metric $L_\mathcal{P}$, and our distribution matching loss can be estimated in the following way:

$$\begin{aligned} & L_{\text{distr}}(\theta) = \text{JS}({f_\theta^{-1}}_\#\big[{f_\theta^y}_\#[q(x)]\ p(z)\big], q(x)) \\ \approx\ & \frac{1}{2N} \max_T \sum_n \Big\{ \log \sigma(T(x^{(n)})) \\ & + \log\Big(1 - \sigma\big[T\big(f_\theta^{-1}(f_\theta^y(x^{(n)}), z^{(n)})\big)\big]\Big)\Big\} + \log 2, \end{aligned} \tag{7}$$

where $\{z^{(n)}\}_{n=1}^N$ are i.i.d. samples from $p(z)$, σ is the sigmoid function, $T : \mathcal{X} \to \mathbb{R}$ is a function on $\mathcal{X}$ ($\sigma(T(\cdot))$ is regarded as a discriminator in GAN literatures), and "$\approx$" is due to Monte Carlo estimation. The appendix provides the details. For practical computation, the function T is parameterized as a neural network T_ϕ and $\max_T$ amounts to $\max_\phi$. The expression (7) is also suitable for estimating its gradient w.r.t. θ and ϕ, thus optimization is made practical.

Total Loss We optimize our IRN model by minimizing the compact loss $L_{\text{total}}(\theta)$ with the combination of HR reconstruction loss $L_{\text{recon}}(\theta)$, LR guidance loss $L_{\text{guide}}(\theta)$ and distribution matching loss $L_{\text{distr}}(\theta)$:

$$L_{\text{total}} := \lambda_1 L_{\text{recon}} + \lambda_2 L_{\text{guide}} + \lambda_3 L_{\text{distr}}, \tag{8}$$

where $\lambda_1, \lambda_2, \lambda_3$ are coefficients for balancing different loss terms.

Loss Minimization in Practice As an issue in practice, we find that directly minimizing the total loss $L_{\text{total}}(\theta)$ is difficult to train, due to the unstable training process of GANs [5]. We propose a pre-training stage that adopts a weakened but more stable surrogate of the distribution matching loss. Recall that the distribution matching loss $L_\mathcal{P}\big({f_\theta^{-1}}_\#\big[{f_\theta^y}_\#[q(x)]\ \ p(z)\big], q(x)\big)$ on $\mathcal{X}$ has the same asymptotic effect as the loss $L_\mathcal{P}({f_\theta^y}_\#[q(x)]\ p(z), (f_\theta^y, f_\theta^z)_\#[q(x)])$ on $\mathcal{Y} \times \mathcal{Z}$. The surrogate considers partial distribution matching on $\mathcal{Z}$, i.e., $L_\mathcal{P}(p(z), {f_\theta^z}_\#[q(x)])$. Since the density function of one of the distributions, $p(z)$, is now made available, we can choose more stable distribution metrics for minimization, such as the cross entropy (CE):

$$\begin{aligned} & L'_{\text{distr}}(\theta) := \text{CE}({f_\theta^z}_\#[q(x)], p(z)) \\ =\ & -\mathbb{E}_{{f_\theta^z}_\#[q(x)]}[\log p(z)] = -\mathbb{E}_{q(x)}[\log p(z = f_\theta^z(x))]. \end{aligned} \tag{9}$$

A related training method is the maximum likelihood estimation (MLE), i.e., $\max_\theta \mathbb{E}_{q(x)}[\log {f_\theta^{-1}}_\#[p(y, z)]]$, which is widely adopted by prevalent flow-based generative models [15,16,29,4]. It is equivalent to minimizing the Kullback-Leibler (KL) divergence $\text{KL}(q(x), {f_\theta^{-1}}_\#[p(y, z)])$. The mentioned models explicitly specify the density function of $p(y, z)$, thus the density function of ${f_\theta^{-1}}_\#[p(y, z)]$ is made available

together with the tractable Jacobian determinant computation of f_θ. However, the same objective cannot be leveraged for our model since we do not have the density function for ${f_\theta^y}_\#[q(x)]\ \ p(z)$; only that of $p(z)$ is known[†]. The invertible neural network (INN) [3] meets the same problem and cannot use MLE either.

We call IRN as our model trained by minimizing the following total objective:

$$L_{\mathrm{IRN}} := \lambda_1 L_{\mathrm{recon}} + \lambda_2 L_{\mathrm{guide}} + \lambda_3 L'_{\mathrm{distr}}. \tag{10}$$

After the pre-training stage, we restore the full distribution matching loss L_{distr} in the objective in place of L'_{distr}. Additionally, we also employ a perceptual loss [25] L_{percp} on $\mathcal{X}$, which measures the difference of two images via their semantic features extracted by benchmarking models. It enhances the perceptual similarity between generated and true images thus helps to produce more realistic images. The perceptual loss has several slightly modified variants which mainly differ in the position of the objective features [33][50]. We adopt the variant proposed in [50]. We call IRN+ as our model trained by minimizing the following total objective:

$$L_{\mathrm{IRN+}} := \lambda_1 L_{\mathrm{recon}} + \lambda_2 L_{\mathrm{guide}} + \lambda_3 L_{\mathrm{distr}} + \lambda_4 L_{\mathrm{percp}}.$$

Difference with GAN On one hand, although the JS divergence is adopted to instead of MLE as distribution matching loss for optimizing IRN, there is one thing should be noted that our model is totally different from typical GAN models: besides the latent variable z which has a prior, there exists y in IRN model which is subject to some distributional constraints, and our model does not have a standalone distribution on x. Therefore, the conventional way to use adversarial loss simply cannot be applied, and we match towards the data distribution with an essentially different distribution from the GAN model distribution. On the other hand, except for JS divergence, a CE loss for L'_{distr} is also adopted as distribution matching loss of IRN. In general, the distribution matching loss reflects the essential idea of IRN, which is totally different from GAN.

4 Experiments

4.1 Dataset and Settings

We employ the widely used DIV2K [1] image restoration dataset to train our model, which contains 800 high-quality 2K resolution images in the training set, and 100 in the validation set. Besides, we evaluate our model on 4 additional standard datasets, i.e. the Set5 [11], Set14 [58], BSD100 [39], and Urban100 [23]. Following the setting in [36], we quantitatively evaluate the peak noise-signal ratio (PSNR) and SSIM [51] on the Y channel of images represented in the YCbCr (Y, Cb, Cr) color space. Due to space constraint, we leave training strategy details in the appendix.

[†] MLEs corresponding to minimizing $\mathrm{KL}(q(x|y), {f_\theta^{-1}(y,\cdot)}_\#[p(z)])$ or $\mathrm{KL}\Big(q(x), \Big({\mathbb{E}_{f_\theta^y{}_\#[q(x)]}[f_\theta^{-1}(y,\cdot)]}\Big)_\#[p(z)]\Big)$ are also impossible, since the pushed-forward distributions have a.e. zero density in $\mathcal{X}$ so the KL is a.e. infinite.

4.2 Evaluation on Reconstructed HR Images

This section reports the quantitative and qualitative performance of HR image reconstruction with different downscaling and upscaling methods. We consider two kinds of reconstruction methods as our baselines: (1) downscaling with Bicubic interpolation and upscaling with state-of-the-art SR models [17,36,60,59,50,14]; (2) downscaling with upscaling-optimal models [26,34,49] and upscaling with SR models. For the method of [50], we denote ESRGAN as their pre-trained model, and ESRGAN+ as their GAN-based model. We further investigate the influence of different z samples on the reconstructed image x. Finally, we empirically study the effectiveness of the different types of loss in the pre-training stage.

Table 1. Quantitative evaluation results (PSNR / SSIM) of different downscaling and upscaling methods for image reconstruction on benchmark datasets: Set5, Set14, BSD100, Urban100, and DIV2K validation set. For our method, differences on average PSNR / SSIM from different z samples are less than 0.02. We report the mean result over 5 draws.

Downscaling & Upscaling	Scale	Param	Set5	Set14	BSD100	Urban100	DIV2K
Bicubic & Bicubic	2×	/	33.66 / 0.9299	30.24 / 0.8688	29.56 / 0.8431	26.88 / 0.8403	31.01 / 0.9393
Bicubic & SRCNN [17]	2×	57.3K	36.66 / 0.9542	32.45 / 0.9067	31.36 / 0.8879	29.50 / 0.8946	–
Bicubic & EDSR [36]	2×	40.7M	38.20 / 0.9606	34.02 / 0.9204	32.37 / 0.9018	33.10 / 0.9363	35.12 / 0.9699
Bicubic & RDN [60]	2×	22.1M	38.24 / 0.9614	34.01 / 0.9212	32.34 / 0.9017	32.89 / 0.9353	–
Bicubic & RCAN [59]	2×	15.4M	38.27 / 0.9614	34.12 / 0.9216	32.41 / 0.9027	33.34 / 0.9384	–
Bicubic & SAN [14]	2×	15.7M	38.31 / 0.9620	34.07 / 0.9213	32.42 / 0.9028	33.10 / 0.9370	–
TAD & TAU [26]	2×	–	38.46 / –	35.52 / –	36.68 / –	35.03 / –	39.01 / –
CNN-CR & CNN-SR [34]	2×	–	38.88 / –	35.40 / –	33.92 / –	33.68 / –	–
CAR & EDSR [49]	2×	51.1M	38.94 / 0.9658	35.61 / 0.9404	33.83 / 0.9262	35.24 / 0.9572	38.26 / 0.9599
IRN (ours)	2×	1.66M	43.99 / 0.9871	40.79 / 0.9778	41.32 / 0.9876	39.92 / 0.9865	44.32 / 0.9908
Bicubic & Bicubic	4×	/	28.42 / 0.8104	26.00 / 0.7027	25.96 / 0.6675	23.14 / 0.6577	26.66 / 0.8521
Bicubic & SRCNN [17]	4×	57.3K	30.48 / 0.8628	27.50 / 0.7513	26.90 / 0.7101	24.52 / 0.7221	–
Bicubic & EDSR [36]	4×	43.1M	32.62 / 0.8984	28.94 / 0.7901	27.79 / 0.7437	26.86 / 0.8080	29.38 / 0.9032
Bicubic & RDN [60]	4×	22.3M	32.47 / 0.8990	28.81 / 0.7871	27.72 / 0.7419	26.61 / 0.8028	–
Bicubic & RCAN [59]	4×	15.6M	32.63 / 0.9002	28.87 / 0.7889	27.77 / 0.7436	26.82 / 0.8087	30.77 / 0.8460
Bicubic & ESRGAN [50]	4×	16.3M	32.74 / 0.9012	29.00 / 0.7915	27.84 / 0.7455	27.03 / 0.8152	30.92 / 0.8486
Bicubic & SAN [14]	4×	15.7M	32.64 / 0.9003	28.92 / 0.7888	27.78 / 0.7436	26.79 / 0.8068	–
TAD & TAU [26]	4×	–	31.81 / –	28.63 / –	28.51 / –	26.63 / –	31.16 / –
CAR & EDSR [49]	4×	52.8M	33.88 / 0.9174	30.31 / 0.8382	29.15 / 0.8001	29.28 / 0.8711	32.82 / 0.8837
IRN (ours)	4×	4.35M	36.19 / 0.9451	32.67 / 0.9015	31.64 / 0.8826	31.41 / 0.9157	35.07 / 0.9318

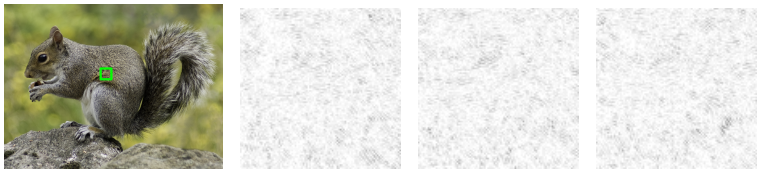
Quantitative Results Table 1 summarizes the quantitative comparison results of different reconstruction methods where IRN significantly outperforms previous state-of-the-art methods regarding PSNR and SSIM in all datasets. We leave the results of IRN+ in the appendix because it is a visual-perception-oriented model. As shown in Table 1, upscaling-optimal downscaling models largely enhance the reconstruction of HR images by state-of-the-art SR models compared with downscaling with Bicubic interpolation. However, they still hardly achieve satisfying results due to the ill-posed nature of upscaling. In contract, with the invertibility, IRN significantly boosts the PSNR metric about 4-5 dB and 2-3 dB on each benchmark dataset in $2\times$ and $4\times$ scale downsampling and reconstruction, and the improvement goes as large as 5.94 dB compared with the state-of-the-art downscaling and upscaling model. These results indicate an exponential improvement of IRN in the reduction of information loss, which also accords with the significant improvement in SSIM.

Moreover, the number of parameters of IRN is relatively small. When Bicubic downscaling and super-resolution methods require large model size (>15M) for better results, our IRN only has 1.66M and 4.35M parameters in scale $2\times$ and $4\times$ respectively. It indicates that our model is light-weight and efficient.

Fig. 3. Qualitative results of upscaling the $4\times$ downscaled images. IRN recovers rich details, leading to both visually pleasing performance and high similarity to the original images. IRN+ produces even sharper and more realistic details. See the appendix for more results.

Qualitative Results We then qualitatively evaluate IRN and IRN+ by demonstrating details of the upscaled images. As shown in Fig. 8, HR images reconstructed by IRN and IRN+ achieve better visual quality and fidelity than those of previous state-of-the-art methods. IRN recovers richer details, which contributes to the pleasing visual quality. IRN+ further produces sharper and more realistic images as the effect of the distribution matching objective. For the 'Comic' example, we observe that the IRN and IRN+ are the only models that can recover the complicated textures on the headwear and necklace, as well as the sharp and realistic fingers. Previous perceptual-driven methods such as ESRGAN [50] also claim that the sharpness and reality of their generated HR images are satisfied. However, the visually unreasonable and unpleasing details produced by their model often lead to dissimilarity to the original images. We leave the high-resolution version and more results in the appendix for spacing reason.

Visualisation on the Influence of z As described in previous sections, we aim to let $z \sim p(z)$ focus on the randomness of high-frequency contents only. In Table 1, the PSNR difference is less than 0.02 dB for each image with different samples of z. In order to verify whether z has learned only to influence high-frequency information, we calculate and present the difference between different draws of z in Fig. 7. We can see in the figure that there is only a tiny noisy distinction in high-frequency regions without typical textures, which can hardly be perceived when combined with low-frequency contents. This indicates that our IRN has learned to reconstruct most meaningful high-frequency contents, while embedding senseless noise into randomness.



(a) (b) (c) (d)

Fig. 4. Visualisation of the difference of upscaled HR images from multiple draws of z. (a): original image; (b-d): HR image differences of three z drawn from a common z sample. Darker color means larger difference. It shows that the differences are random noise in high-frequency regions without a typical texture.

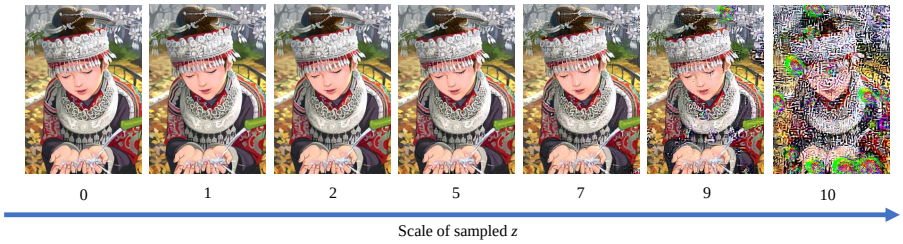


Fig. 5. Results of HR images by IRN+ with out-of-distribution samples of z. We train z with an isotropic Gaussian distribution, and illustrate upscaling results when scaling z sampled from the isotropic Gaussian distribution.

As mentioned above, we train the model to encourage $p(z)$ to obey a simple and easy-to-sample distribution, i.e., isotropic Gaussian distribution. In order to further verify the effectiveness of the learned model, we feed $(y, \alpha z)$ into our IRN+ to obtain x_α by controlling the scale of sampled z with different values of α. As shown in Fig. 5, a larger deviation to the original distribution results in more noisy textures and distortion. It demonstrates that our model transforms z faithfully to follow the specified distribution, and is also robust to slight distribution deviation.

Table 2. Analysis results (PSNR/SSIM) of training IRN with L_1 or L_2 LR guide and HR reconstruction loss, with/without partial distribution matching loss, on Set5, Set14, BSD100, Urban100 and DIV2K validation sets with scale 4×.

L_{guide}	L_{recon}	$L_{distr'}$	Set5	Set14	BSD100	Urban100	DIV2K
L_1	L_1	Yes	34.75 / 0.9296	31.42 / 0.8716	30.42 / 0.8451	30.11 / 0.8903	33.64 / 0.9079
L_1	L_2	Yes	34.93 / 0.9296	31.76 / 0.8776	31.01 / 0.8562	30.79 / 0.8986	34.11 / 0.9116
L_2	L_1	Yes	36.19 / 0.9451	32.67 / 0.9015	31.64 / 0.8826	31.41 / 0.9157	35.07 / 0.9318
L_2	L_2	Yes	35.93 / 0.9402	32.51 / 0.8937	31.64 / 0.8742	31.40 / 0.9105	34.90 / 0.9308
L_2	L_1	No	36.12 / 0.9455	32.18 / 0.8995	31.49 / 0.8808	30.91 / 0.9102	34.90 / 0.9308

Analysis on the Losses We conduct experiments to analyze the components in the loss of Eqs. (4, 5, 9). As shown in Table 2, IRN performs the best when the LR guidance loss is the L_2 loss and the HR reconstruction loss is the L_1 loss. The reason is that the L_1 loss encourages more pixel-wise similarity, while the L_2 loss is less sen-

sitive to minor changes. In the forward procedure, we utilize the Bicubic-downscaled images as guidance, but we do not aim to exactly learn the Bicubic downscaling, which may harm the inverse procedure. The forward reconstruction loss only acts as a constraint to maintain visually pleasing downscaling, so the L_2 loss is more suitable. In the backward procedure, on the other hand, our goal is to reconstruct the ground truth image accurately. Therefore, the L_1 loss is more appropriate, as also identified by other super-resolution works. Table 2 also demonstrates the necessity of the partial distribution matching loss of Eq. (9), which restricts the marginal distributions on $\mathcal{Z}$, and benefits the forward distribution learning.

4.3 Evaluation on Downscaled LR Images

We also evaluate the quality of LR images downscaled by our IRN. We demonstrate the similarity index between our LR images and Bicubic-based LR images, and present similar visual perception of them, to show that IRN is able to perform as well as Bicubic.

Table 3. SSIM results between the images downscaled by IRN and by Bicubic on the Set5, Set14, BSD100, Urban100 and DIV2K validation sets.

Scale	Set5	Set14	BSD100	Urban100	DIV2K
2×	0.9957	0.9936	0.9936	0.9941	0.9945
4×	0.9964	0.9927	0.9923	0.9916	0.9933

As shown in Table 3, images downscaled by IRN are extremely similar to those by Bicubic. Fig. 12 and more figures in the appendix illustrate the visual similarity between them, which demonstrates the proper perception of our downscaled images.

(a) (b) (c) (d)

Fig. 6. Demonstration of the downscaled images from Set14 and DIV2K validation sets. Left column (a,c): Image downscaled by Bicubic. Right column (b,d): Image downscaled by IRN. They share a similar visual perception.

5 Conclusion

In this paper, we propose a novel invertible network for the image rescaling task, with which the ill-posed nature of the task is largely mitigated. We explicitly model the statistics of the case-specific high-frequency information that is lost in downscaling as a latent variable following a specified case-agnostic distribution which is easy to sample from. The network models the rescaling processes by invertibly transforming between

an HR image and an LR image with the latent variable. With the statistical knowledge of the latent variable, we draw a sample of it for upscaling from a downscaled LR image (whose specific high-frequency information was lost during downscaling, of course). We design a specific invertible architecture tailored for image rescaling, and an effective training objective to enforce the model to have desired downscaling and upscaling behavior, as well as to output the latent variable with the specified properties. Extensive experiments demonstrate that our model significantly improves both quantitative and qualitative performance of upscaling reconstruction from downscaled LR images, while being light-weighted.

References

1. Agustsson, E., Timofte, R.: Ntire 2017 challenge on single image super-resolution: Dataset and study. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops. pp. 126–135 (2017) 10
2. Agustsson, E., Tschannen, M., Mentzer, F., Timofte, R., Gool, L.V.: Generative adversarial networks for extreme learned image compression. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 221–231 (2019) 5
3. Ardizzone, L., Kruse, J., Wirkert, S., Rahner, D., Pellegrini, E.W., Klessen, R.S., Maier-Hein, L., Rother, C., Köthe, U.: Analyzing inverse problems with invertible neural networks. In: Proceedings of the International Conference on Learning and Representations (2019) 3, 5, 10
4. Ardizzone, L., Lüth, C., Kruse, J., Rother, C., Köthe, U.: Guided image generation with conditional invertible neural networks. arXiv preprint arXiv:1907.02392 (2019) 3, 5, 6, 9
5. Arjovsky, M., Bottou, L.: Towards principled methods for training generative adversarial networks. In: Proceedings of the International Conference on Learning and Representations (2017) 9
6. Ballé, J., Laparra, V., Simoncelli, E.P.: End-to-end optimized image compression. arXiv preprint arXiv:1611.01704 (2016) 5
7. Ballé, J., Minnen, D., Singh, S., Hwang, S.J., Johnston, N.: Variational image compression with a scale hyperprior. arXiv preprint arXiv:1802.01436 (2018) 5
8. Behrmann, J., Grathwohl, W., Chen, R.T., Duvenaud, D., Jacobsen, J.H.: Invertible residual networks. In: International Conference on Machine Learning. pp. 573–582 (2019) 4
9. Bengio, Y., Léonard, N., Courville, A.: Estimating or propagating gradients through stochastic neurons for conditional computation. arXiv preprint arXiv:1308.3432 (2013) 7
10. Berg, R.v.d., Hasenclever, L., Tomczak, J.M., Welling, M.: Sylvester normalizing flows for variational inference. In: Proceedings of the Conference on Uncertainty in Artificial Intelligence (2018) 4
11. Bevilacqua, M., Roumy, A., Guillemot, C., Alberi-Morel, M.L.: Low-complexity single-image super-resolution based on nonnegative neighbor embedding (2012) 10
12. Bruckstein, A.M., Elad, M., Kimmel, R.: Down-scaling for better transform compression. IEEE Transactions on Image Processing **12**(9), 1132–1144 (2003) 1
13. Chen, R.T., Behrmann, J., Duvenaud, D., Jacobsen, J.H.: Residual flows for invertible generative modeling. arXiv preprint arXiv:1906.02735 (2019) 4
14. Dai, T., Cai, J., Zhang, Y., Xia, S.T., Zhang, L.: Second-order attention network for single image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 11065–11074 (2019) 2, 3, 11, 21
15. Dinh, L., Krueger, D., Bengio, Y.: NICE: Non-linear independent components estimation. In: Workshop of the International Conference on Learning Representations (2015) 3, 4, 7, 9
16. Dinh, L., Sohl-Dickstein, J., Bengio, S.: Density estimation using real NVP. In: Proceedings of the International Conference on Learning Representations (2017) 3, 4, 5, 7, 9
17. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE transactions on pattern analysis and machine intelligence **38**(2), 295–307 (2015) 2, 3, 11, 21
18. Freedman, G., Fattal, R.: Image and video upscaling from local self-examples. ACM Transactions on Graphics (TOG) **30**(2), 12 (2011) 3
19. Giachetti, A., Asuni, N.: Real-time artifact-free image upscaling. IEEE Transactions on Image Processing **20**(10), 2760–2768 (2011) 1
20. Glasner, D., Bagon, S., Irani, M.: Super-resolution from a single image. In: 2009 IEEE 12th international conference on computer vision. pp. 349–356. IEEE (2009) 3

21. Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., Bengio, Y.: Generative adversarial nets. In: Advances in Neural Information Processing Systems. pp. 2672–2680. NIPS Foundation, Montréal, Canada (2014) 3, 20
22. Grathwohl, W., Chen, R.T., Betterncourt, J., Sutskever, I., Duvenaud, D.: FFJORD: Free-form continuous dynamics for scalable reversible generative models. In: Proceedings of the International Conference on Learning and Representations (2019) 4
23. Huang, J.B., Singh, A., Ahuja, N.: Single image super-resolution from transformed self-exemplars. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 5197–5206 (2015) 10
24. Irani, D.G.S.B.M.: Super-resolution from a single image. In: Proceedings of the IEEE International Conference on Computer Vision, Kyoto, Japan. pp. 349–356 (2009) 2
25. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution. In: European conference on computer vision. pp. 694–711. Springer (2016) 10
26. Kim, H., Choi, M., Lim, B., Mu Lee, K.: Task-aware image downscaling. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 399–414 (2018) 1, 2, 4, 8, 11, 21
27. Kim, K.I., Kwon, Y.: Single-image super-resolution using sparse regression and natural image prior. IEEE transactions on pattern analysis and machine intelligence **32**(6), 1127–1133 (2010) 3
28. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 20
29. Kingma, D.P., Dhariwal, P.: Glow: Generative flow with invertible 1x1 convolutions. In: Advances in Neural Information Processing Systems. pp. 10215–10224 (2018) 3, 4, 9
30. Kingma, D.P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., Welling, M.: Improved variational inference with inverse autoregressive flow. In: Advances in Neural Information Processing Systems. pp. 4743–4751 (2016) 4
31. Kopf, J., Shamir, A., Peers, P.: Content-adaptive image downscaling. ACM Transactions on Graphics (TOG) **32**(6), 173 (2013) 4
32. Kumar, M., Babaeizadeh, M., Erhan, D., Finn, C., Levine, S., Dinh, L., Kingma, D.: Videoflow: A flow-based generative model for video. arXiv preprint arXiv:1903.01434 (2019) 4
33. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 4681–4690 (2017) 10, 20
34. Li, Y., Liu, D., Li, H., Li, L., Li, Z., Wu, F.: Learning a convolutional neural network for image compact-resolution. IEEE Transactions on Image Processing **28**(3), 1092–1107 (2018) 1, 2, 4, 11
35. Lienhart, R., Maydt, J.: An extended set of haar-like features for rapid object detection. In: Proceedings. international conference on image processing. vol. 1, pp. I–I. IEEE (2002) 6
36. Lim, B., Son, S., Kim, H., Nah, S., Mu Lee, K.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of the IEEE conference on computer vision and pattern recognition workshops. pp. 136–144 (2017) 2, 3, 7, 10, 11, 21
37. Lin, W., Dong, L.: Adaptive downsampling to improve image compression at low bit rates. IEEE Transactions on Image Processing **15**(9), 2513–2521 (2006) 1
38. Liu, J., He, S., Lau, R.W.: $l\_\{0\}$-regularized image downscaling. IEEE Transactions on Image Processing **27**(3), 1076–1085 (2017) 4
39. Martin, D., Fowlkes, C., Tal, D., Malik, J., et al.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. Iccv Vancouver: (2001) 10

40. Minnen, D., Ballé, J., Toderici, G.D.: Joint autoregressive and hierarchical priors for learned image compression. In: Advances in Neural Information Processing Systems. pp. 10771–10780 (2018) 5
41. Mitchell, D.P., Netravali, A.N.: Reconstruction filters in computer-graphics. In: ACM Siggraph Computer Graphics. vol. 22-4, pp. 221–228. ACM (1988) 3, 7
42. Oeztireli, A.C., Gross, M.: Perceptually based downscaling of images. ACM Transactions on Graphics (TOG) **34**(4), 77 (2015) 4
43. van der Ouderaa, T.F., Worrall, D.E.: Reversible gans for memory-efficient image-to-image translation. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 4720–4728 (2019) 5
44. Rezende, D., Mohamed, S.: Variational inference with normalizing flows. In: Proceedings of the International Conference on Machine Learning. pp. 1530–1538 (2015) 4
45. Rippel, O., Bourdev, L.: Real-time adaptive image compression. In: Proceedings of the 34th International Conference on Machine Learning-Volume 70. pp. 2922–2930. JMLR. org (2017) 5
46. Schulter, S., Leistner, C., Bischof, H.: Fast and accurate image upscaling with super-resolution forests. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 3791–3799 (2015) 1, 3
47. Shannon, C.E.: Communication in the presence of noise. Proceedings of the IRE **37**(1), 10–21 (1949) 5
48. Shen, M., Xue, P., Wang, C.: Down-sampling based video coding using super-resolution technique. IEEE Transactions on Circuits and Systems for Video Technology **21**(6), 755–765 (2011) 1
49. Sun, W., Chen, Z.: Learned image downscaling for upscaling using content adaptive resampler. IEEE Transactions on Image Processing **29**, 4027–4040 (2020) 1, 2, 4, 8, 11, 21
50. Wang, X., Yu, K., Wu, S., Gu, J., Liu, Y., Dong, C., Qiao, Y., Change Loy, C.: Esrgan: Enhanced super-resolution generative adversarial networks. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 0–0 (2018) 2, 3, 7, 10, 11, 12, 21
51. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P., et al.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing **13**(4), 600–612 (2004) 4, 10
52. Weber, N., Waechter, M., Amend, S.C., Guthe, S., Goesele, M.: Rapid, detail-preserving image downscaling. ACM Transactions on Graphics (TOG) **35**(6), 205 (2016) 4
53. Wilson, P.I., Fernandez, J.: Facial feature detection using haar classifiers. Journal of Computing Sciences in Colleges **21**(4), 127–133 (2006) 6
54. Wu, X., Zhang, X., Wang, X.: Low bit-rate image compression via adaptive down-sampling and constrained least squares upconversion. IEEE Transactions on Image Processing **18**(3), 552–561 (2009) 1
55. Yang, J., Wright, J., Huang, T.S., Ma, Y.: Image super-resolution via sparse representation. IEEE transactions on image processing **19**(11), 2861–2873 (2010) 2
56. Yeo, H., Do, S., Han, D.: How will deep learning change internet video delivery? In: Proceedings of the 16th ACM Workshop on Hot Topics in Networks. pp. 57–64. ACM (2017) 1
57. Yeo, H., Jung, Y., Kim, J., Shin, J., Han, D.: Neural adaptive content-aware internet video delivery. In: 13th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 18). pp. 645–661 (2018) 1
58. Zeyde, R., Elad, M., Protter, M.: On single image scale-up using sparse-representations. In: International conference on curves and surfaces. pp. 711–730. Springer (2010) 10
59. Zhang, Y., Li, K., Li, K., Wang, L., Zhong, B., Fu, Y.: Image super-resolution using very deep residual channel attention networks. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 286–301 (2018) 2, 3, 11, 21

60. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 2472–2481 (2018) 2, 3, 11, 21

Appendix: Invertible Image Rescaling

A Details of the distribution loss

According to the main text, we choose the Jensen-Shannon (JS) divergence as the distribution metric and minimize the difference between ${f_\theta^{-1}}_{\#}\left[{f_\theta^y}_{\#}[q(x)]\ p(z)\right]$ and $q(x)$:

$$
\begin{aligned}
& L_{\text{distr}}(\theta) = \text{JS}({f_\theta^{-1}}_{\#}\left[{f_\theta^y}_{\#}[q(x)]\ p(z)\right], q(x)) \\
= & \frac{1}{2}\max_T \Big\{\mathbb{E}_{q(x)}\left[\log\sigma(T(x))\right] \\
& + \mathbb{E}_{x'\sim {f_\theta^{-1}}_{\#}\left[{f_\theta^y}_{\#}[q(x)]\ p(z)\right]}\left[\log\left(1-\sigma(T(x'))\right)\right]\Big\} + \log 2 \\
= & \frac{1}{2}\max_T \left\{\mathbb{E}_{q(x)}\left[\log\sigma(T(x))\right]\right. \\
& \left. + \mathbb{E}_{(y,z)\sim {f_\theta^y}_{\#}[q(x)]\ p(z)}\left[\log\left(1-\sigma(T(f_\theta^{-1}(y,z)))\right)\right]\right\} + \log 2 \\
\approx & \frac{1}{2N}\max_T \sum_n \Big\{\log\sigma(T(x^{(n)})) \\
& + \log\left(1-\sigma(T(f_\theta^{-1}(f_\theta^y(x^{(n)}), z^{(n)})))\right)\Big\} + \log 2. \qquad (11)
\end{aligned}
$$

The first equality stems from the variational form of the JS divergence which is composed for training generative adversarial nets [21]. The second equality is a reformulation using the definition of pushed-forward distribution. The third approximate equality leads to a Monte Carlo estimation to the objective function using the corresponding samples: $\{z^{(n)}\}_{n=1}^N$ i.i.d. drawn from $p(z)$, and $\{x^{(n)}\}_{n=1}^N \sim q(x)$.

B Detailed Training Strategies on DIV2K dataset

We train and compare our model in $2\times$ and $4\times$ downscaling scale with one and two downscaling modules respectively. Each downscaling module has 8 InvBlocks and downscale the original image by $2\times$. We use Adam optimizer [28] with $\beta_1 = 0.9, \beta_2 = 0.999$ to train our model. The mini-batch size is set to 16. The input HR image is randomly cropped into 144×144 and augmented by applying random horizontal and vertical flips. In the pre-training stage, the total number of iteration is $50K$, and the learning rate is initialized as 2×10^{-4} where halved at $[10k, 20k, 30k, 40k]$ mini-batch updates. The hyper-parameters in Eqn.10 are set as $\lambda_1 = 1, \lambda_2 = 16, \lambda_3 = 1$. After pre-training, we finetune our model for another $20K$ iterations as described in Sec.3.3. The learning rate is initialized as 1×10^{-4} and halved at $[5k, 10k]$ iterations. We set $\lambda_1 = 0.01, \lambda_2 = 16, \lambda_3 = 1, \lambda_4 = 0.01$ in Eqn.11 and pre-train the discriminator for 5000 iterations. The discriminator is similar to [33], which contains eight convolutional layers with 3×3 kernels, whose numbers increase from 64 to 512 by a factor 2 each two layers, and two dense layers with 100 hidden units.

Table 4. Quantitative evaluation results (PSNR / SSIM) of different 4× image downscaling and upscaling methods on benchmark datasets: Set5, Set14, BSD100, Urban100, and DIV2K validation set. For our model, differences on average PSNR / SSIM of different samples for z are less than 0.02. We report the mean result. The best result is in red, while the second is in blue.

Downscaling & Upscaling	Scale	Param	Set5	Set14	BSD100	Urban100	DIV2K
Bicubic & Bicubic	4×	/	28.42 / 0.8104	26.00 / 0.7027	25.96 / 0.6675	23.14 / 0.6577	26.66 / 0.8521
Bicubic & SRCNN [17]	4×	57.3K	30.48 / 0.8628	27.50 / 0.7513	26.90 / 0.7101	24.52 / 0.7221	–
Bicubic & EDSR [36]	4×	43.1M	32.62 / 0.8984	28.94 / 0.7901	27.79 / 0.7437	26.86 / 0.8080	29.38 / 0.9032
Bicubic & RDN [60]	4×	22.3M	32.47 / 0.8990	28.81 / 0.7871	27.72 / 0.7419	26.61 / 0.8028	–
Bicubic & RCAN [59]	4×	15.6M	32.63 / 0.9002	28.87 / 0.7889	27.77 / 0.7436	26.82 / 0.8087	30.77 / 0.8460
Bicubic & ESRGAN [50]	4×	16.3M	32.74 / 0.9012	29.00 / 0.7915	27.84 / 0.7455	27.03 / 0.8152	30.92 / 0.8486
Bicubic & SAN [14]	4×	15.7M	32.64 / 0.9003	28.92 / 0.7888	27.78 / 0.7436	26.79 / 0.8068	–
TAD & TAU [26]	4×	–	31.81 / –	28.63 / –	28.51 / –	26.63 / –	31.16 / –
CAR & EDSR [49]	4×	52.8M	33.88 / 0.9174	30.31 / 0.8382	29.15 / 0.8001	29.28 / 0.8711	32.82 / 0.8837
IRN (ours)	4×	4.35M	36.19 / 0.9451	32.67 / 0.9015	31.64 / 0.8826	31.41 / 0.9157	35.07 / 0.9318
IRN+ (ours)	4×	4.35M	33.59 / 0.9147	29.97 / 0.8444	28.94 / 0.8189	28.24 / 0.8684	32.24 / 0.8921

C Quantitive results of IRN+

IRN+ aims at producing more realistic images by minimizing the distribution difference, not exactly matching details of original images as IRN does. The difference will lead to lower PSNR and SSIM, which is the same as GAN-based super-resolution methods. Despite the difference, IRN+ still outperforms most methods in PSNR and SSIM as shown in Table.4, demonstrating the good similarity between the reconstructed images and original HR images.

D Different samples of z

As shown in Fig. 7, there are only tiny noisy distinction in high-frequency areas without typical textures, which can hardly perceived when combined with low-frequency contents. Different samples lead to different but perceptually meaningless noisy distinctions.

E More qualitative results

As shown in Fig.8,9,10,11, images reconstructed by IRN and IRN+ significantly outperforms previous both PSNR-oriented and perceptual-driven methods in visual quality and similarity to original images. IRN is able to reconstruct rich details including detailed lines and textures, which contributes to the pleasing perception. IRN+ further produce sharper and more realistic images as a result of the distribution matching objective.

F Evaluation on downscaled images

As shown in Fig. 12, images downscaled by IRN share a similar visual perception with images downscaled by bicubic.

Fig. 7. Difference between upscaled images by different samples of z. (a): Original image. (b-d): Residual of three randomly upscaled images with another sample (averaged over the three channels). (e-g): Detailed difference of (b-d). The darker the larger difference. To ensure the visual perception, we set rebalance factor by 20.

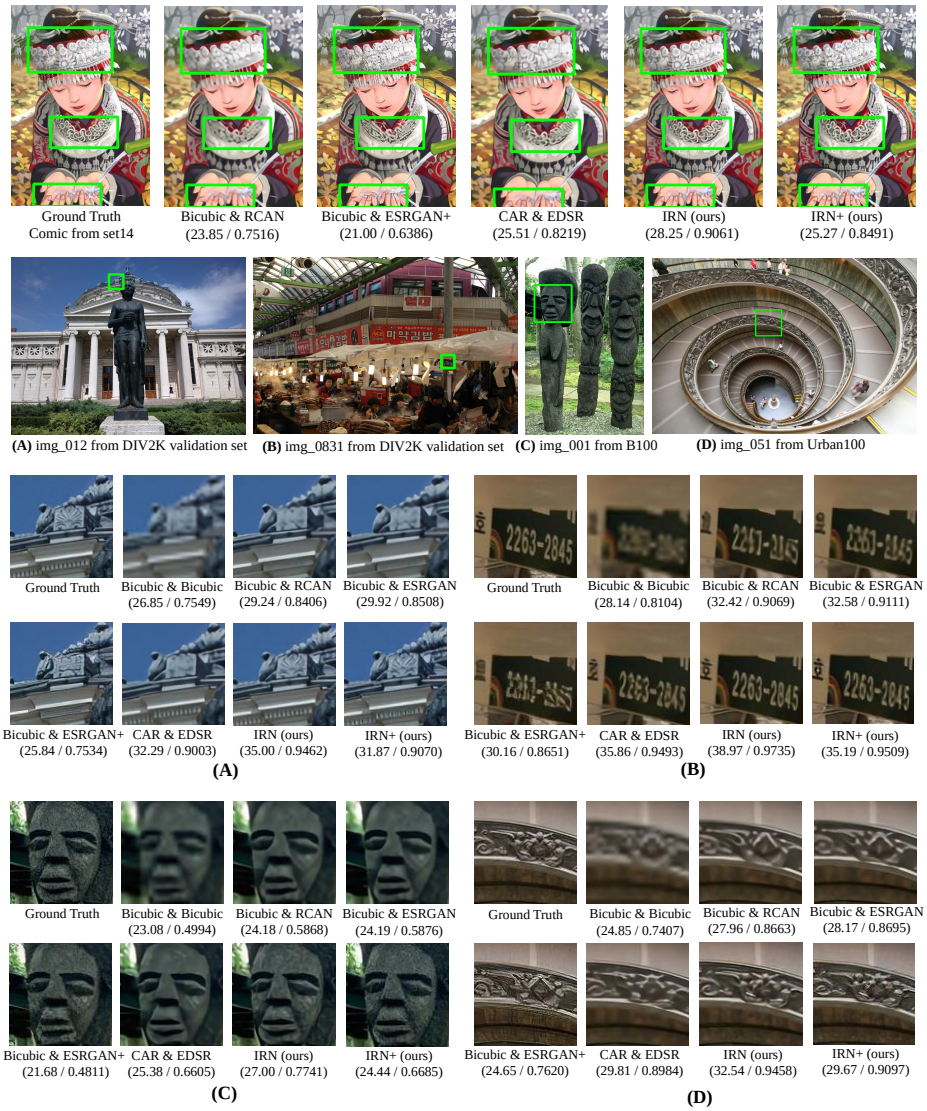


Fig. 8. More qualitative results of upscaling the $4\times$ downscaled images on Set14, BSD100, Urban100 and DIV2K validation datasets.

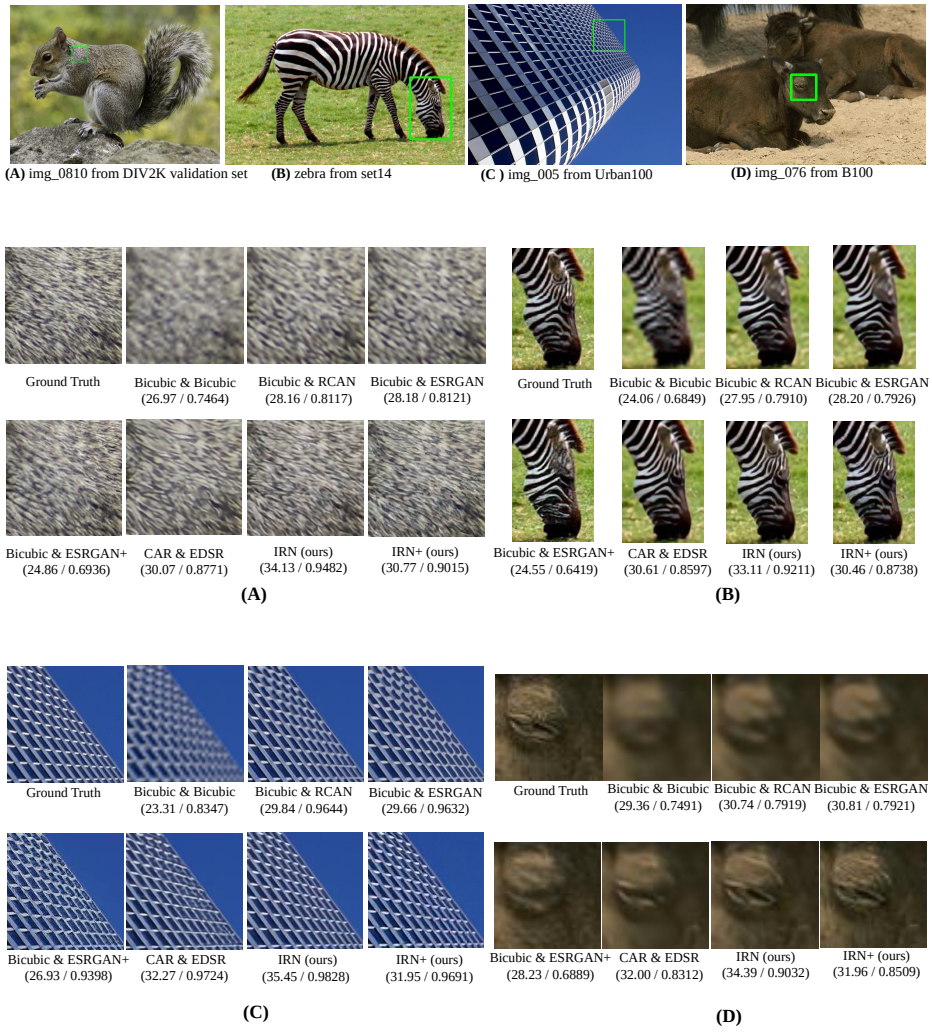


Fig. 9. More qualitative results of upscaling the $4\times$ downscaled images on Set14, BSD100, Urban100 and DIV2K validation datasets.

Fig. 10. More qualitative results of upscaling the $4\times$ downscaled images on DIV2K validation dataset.

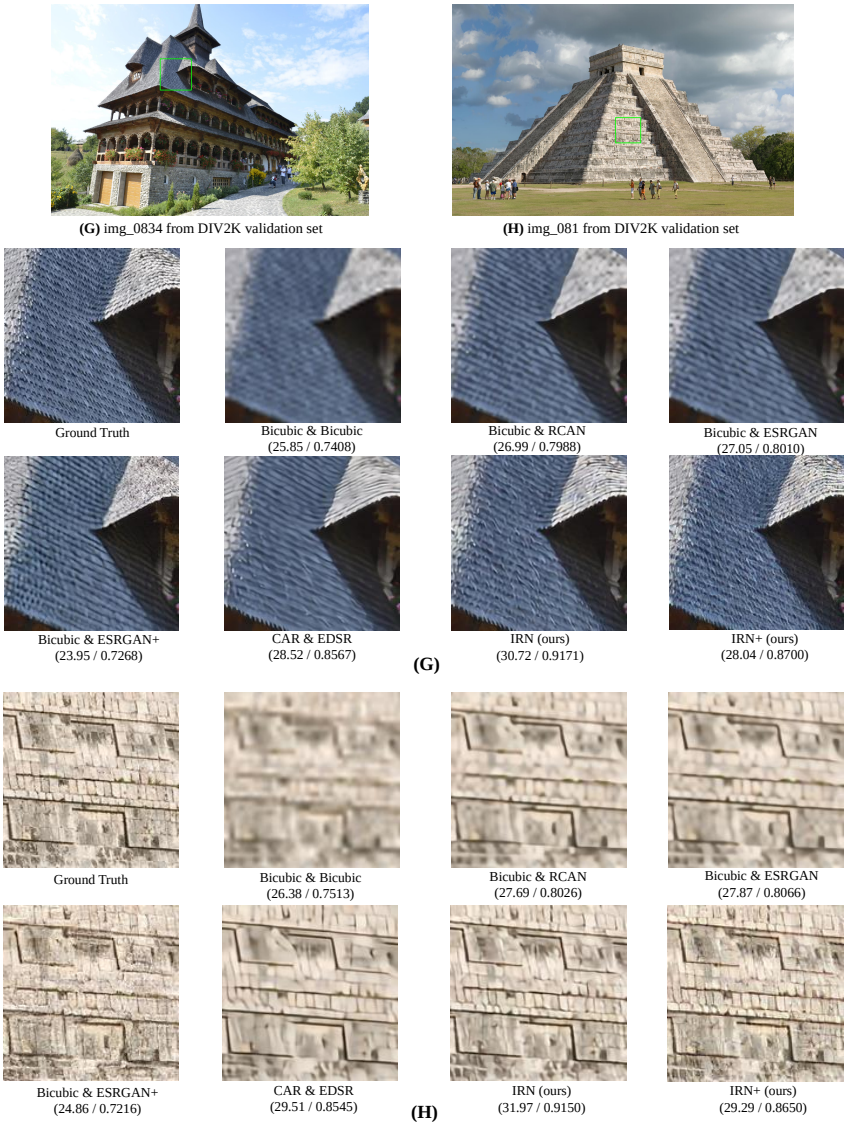


Fig. 11. More qualitative results of upscaling the $4\times$ downscaled images on DIV2K validation dataset.

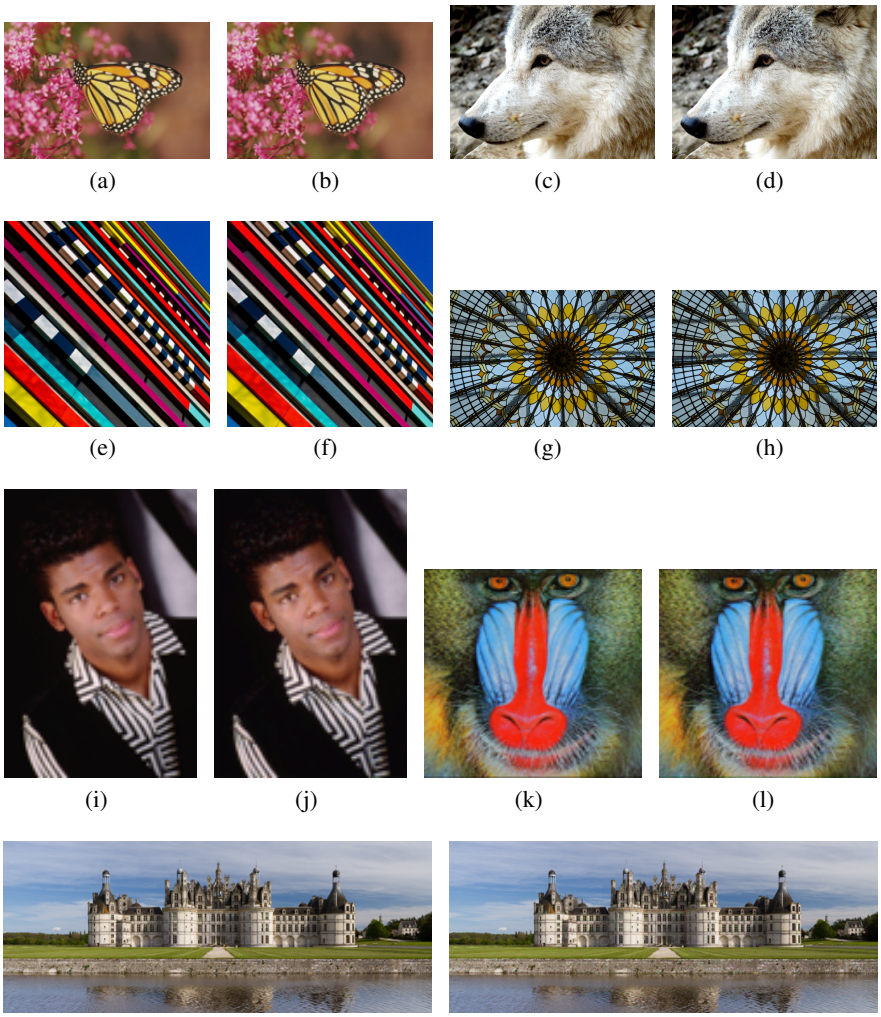


Fig. 12. Demonstration of the downscaled images from Set14, B100, Urban100, and DIV2K validation set. Left column (a,c,e,g,i,k,m): Image downscaled by Bicubic. Right column (b,d,f,h,j,l,n): Image downscaled by IRN. They share a similar visual perception.