Normalizing Flow as a Flexible Fidelity Objective for Photo-Realistic Super-resolution

Andreas Lugmayr Martin Danelljan Fisher Yu Luc Van Gool Radu Timofte
CVL, ETH Zürich, Switzerland

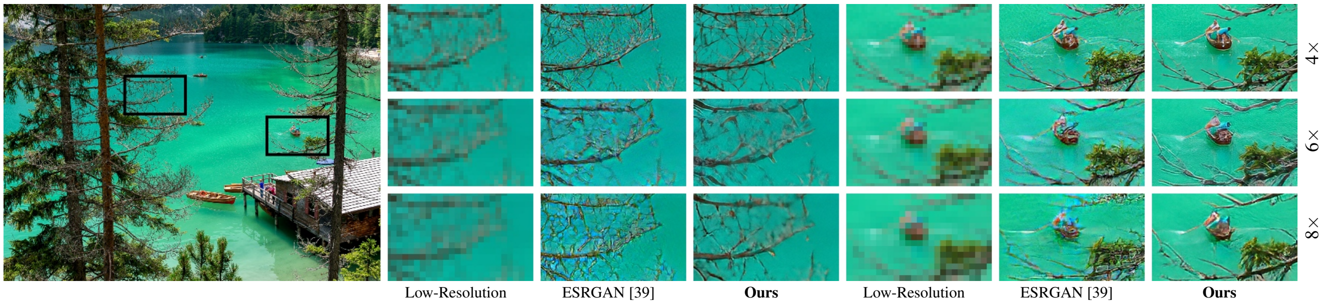

Figure 1. We replace the L_1 loss for super-resolution with a flow-based generalization. The flexibility of our flow-based fidelity loss alleviates the inherent conflict with adversarial losses, leading to a more photo-realistic result and better consistency with the input.

Abstract

Super-resolution is an ill-posed problem, where a ground-truth high-resolution image represents only one possibility in the space of plausible solutions. Yet, the dominant paradigm is to employ pixel-wise losses, such as L_1, which drive the prediction towards a blurry average. This leads to fundamentally conflicting objectives when combined with adversarial losses, which degrades the final quality. We address this issue by revisiting the L_1 loss and show that it corresponds to a one-layer conditional flow. Inspired by this relation, we explore general flows as a fidelity-based alternative to the L_1 objective. We demonstrate that the flexibility of deeper flows leads to better visual quality and consistency when combined with adversarial losses. We conduct extensive user studies for three datasets and scale factors, where our approach is shown to outperform state-of-the-art methods for photo-realistic super-resolution. Code and trained models: git.io/AdFlow

1. Introduction

Photo-realistic image super-resolution (SR) is the task of upscaling a low-resolution (LR) image by adding natural-looking high-frequency content. Since this information is not contained in the LR image, SR assumes that a prior can be learned to add plausible high-frequency components. In general, however, there are infinitely many possible high-resolution (HR) images mapped to the same LR image. Therefore, this task is highly ill-posed, rendering the learning of powerful deep SR models highly challenging.

To cope with the ill-posed nature of the SR problem, existing state-of-the-art methods employ an ensemble of multiple losses designed for different purposes [23, 39, 47]. In particular, these works largely rely on the L_1 loss for fidelity and the adversarial loss for perceptual quality. Theoretically, the L_1 objective aims to predict the average overall plausible HR image manifestations under a Laplace model. That leads to blurry SR predictions, which are generally not perceptually pleasing. In contrast, the adversarial objective prefers images with natural characteristics and high-frequency details. These two losses are thus fundamentally conflicting in nature [5, 6].

The conflict between the L_1 and the adversarial loss has important negative consequences as seen in Figure 1. In order to find a decent trade-off, a precarious balancing between the two terms is needed. The found compromise is not optimal in terms of fidelity nor perceptual quality. Moreover, the conflict between the two losses results in a remarkably inferior low-resolution consistency. That is, the down-sampled version of the predicted SR image is substantially different from the original LR image. The conflict between the losses drives the prediction towards a point *outside* the space of plausible HR images (Illustrated in Fig. 2).

We attribute those shortcomings to the L_1 loss. Since SR is a highly ill-posed problem, the L_1 loss imposes a rigid and exceptionally inaccurate model of the complicated image manifold of solutions. Ideally, we want a loss that ensures fidelity while not penalizing realistic image patches preferred by the adversarial loss. In this work, we therefore first revisit the L_1 loss and view it from a probabilistic perspective. We observe that the L_1 objective corresponds to a one-layer conditional normalizing flow. That inspires us to explore flow-based generalizations capable of better capturing the manifold of plausible HR images to mitigate the conflict between adversarial and fidelity-based objectives.

A few very recent works [30, 40] have investigated flows for SR. However, these approaches use heavy-weight flow networks as an *alternative* to the adversarial loss for perceptual quality. In this work, we pursue a very different view, namely the flow as a fidelity-based generalization of the L_1 objective. Our goal is not to replace the adversarial loss but to find a fidelity-based companion that can enhance the effectiveness of adversarial learning for SR. In contrast, to [30], this allows us to employ much shallower and more practical flow networks, ensuring substantially faster training and inference times. Furthermore, we demonstrate that the adversarial loss effectively removes artifacts generated by purely flow-based methods.

Contributions: Our main contributions of this work are as follows: **(i)** We revisit the L_1 loss from a probabilistic perspective, expressing it as a one-layer conditional flow. **(ii)** We generalize the L_1 fidelity loss by employing a deep flow and demonstrate that it can be more effectively combined with an adversarial loss. **(iii)** We design a more practical, efficient, and stable flow architecture, better suited to the combined objective, leading to 2.5× faster training and inference compared to [30]. **(iv)** We perform comprehensive experiments analyzing the flow loss combined with adversarial losses, giving valuable insights on the effects of increasing the flexibility of the fidelity-based objective. In comprehensive user studies, totaling over 50 000 votes, our approach outperforms state-of-the-art on three different datasets and scale factors.

2. Related Work

Single Image Super-Resolution: is the task of estimating a high-resolution image from a low-resolution counterpart. It is fundamentally an ill-posed inverse problem. While originally addressed by employing interpolation techniques, learned methods are better suited for this complex task. Early learned approaches used sparse-coding [7, 36, 44, 45] and local linear regression [37, 38, 43]. In recent years, deep learning based methods have largely replaced previous techniques for Super-Resolution owing to their highly impressive performance.

Initial deep learning approaches [10, 11, 19, 22, 25] for SR aimed at minimize the L_2 or L_1 distance between the SR and Ground-Truth image. With this objective, the model is effectively trained to predict a mean of plausible super-resolutions corresponding to the given input LR image. To alleviate this problem [23] introduced an adversarial and perceptual loss. Since then, this strategy has remained the predominant approach to super-resolution [2, 12, 14, 17, 18, 26, 35, 39]. Only very few works have investigated other learning formulations. Notably, Zhang *et al*. [47] introduces a selection mechanism based on perceptual quality metrics. In order to achieve an explorable SR formulation, Bahat *et al*. [4] recently trained a stochastic SR network based on mainly adversarial objectives. The output acts as a prior for a low-resolution consistency enforcing module, optimizing the image in a post-processing step.

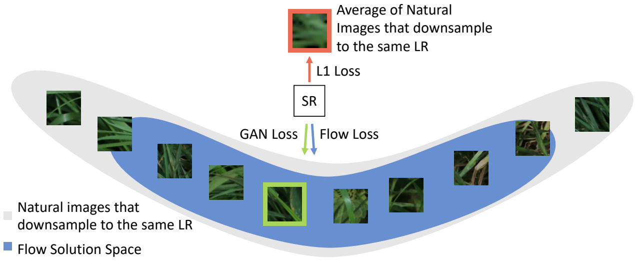
Average of Natural Images that downsample to the same LR
L1 Loss
SR
GAN Loss
Flow Loss
Natural images that downsample to the same LR
Flow Solution Space

Figure 2. While L_1 loss drags the SR prediction towards a blurry mean, both Flow and GAN loss push the prediction towards the real image manifold. Replacing L_1 with flow as fidelity term therefore reduces conflict with the GAN loss.

In recent works, invertible networks have gained popularity for image-to-image translation [8, 9, 24, 30, 34, 40, 41, 42]. Xiao *et al*. [42] uses invertible networks to learn down and upscaling of images. This is similar to compression, but where the compressed representation is constrained to be an LR image. For super-resolution, [30] recently introduced a new strategy based on Normalizing Flows. It aims at replacing adversarial losses with normalizing flows [8, 9, 34]. In contrast, we investigate conditional flows as a replacement for the L_1 loss. In fact, we demonstrate that it forms a direct generalization of the L_1 objective. The aim of this work is to investigate flows as an alternative fidelity-based companion to the adversarial loss.

3. Method

3.1. Revisiting the L_1 Loss

The standard paradigm for learning a SR network g is to directly penalize the *reconstruction error* between a predicted image $g(x)$ and the ground truth HR image $y \in \mathbb{R}^{H \times W \times C}$ corresponding to the LR x. The reconstruction error is usually measured by applying simple norms in a color space (*e.g*., RGB or YCbCr). While initial methods [11, 19, 22] employed the L_2 norm, *i.e*. the mean squared error, later works [25, 49] studied the benefit of the L_1 error,

$$L_1(y, g(x)) = \|y - g(x)\|_1. \tag{1}$$

To understand the implications of this objective function, we use its probabilistic interpretation. Namely, that the L_1

loss (1) corresponds to the Negative Log-Likelihood (NLL) of the Laplace distribution. This derivation will be particularly illustrative for the generalizations considered later.

We first consider a latent variable $z \in \mathbb{R}^{H \times W \times C}$ with the standard Laplace distribution $z \sim \mathcal{L}(0, 1)$. Let f be a be a function that encodes the LR-HR pair into the latent space as $z = f(y; x) = y - g(x)$. Through the inverse relation $y = f^{-1}(z; x) = z + g(x)$ it is easy to see that y follows a Laplace distribution with mean $g(x)$,

$$p(y|x; \theta) = \mathcal{L}(y; g(x), 1) = \frac{1}{2^D} e^{-\|y - g(x)\|_1} . \quad (2)$$

Here, $D = HWC$ is the total dimensionality of y. From a probabilistic perspective, we are thus predicting the conditional distribution $p(y|x; \theta)$ of the HR output image y given the LR x. In particular, our SR network g estimates the *mean* of this distribution under a Laplacian model. In order to learn the parameters θ of the network, we simply minimize the NLL $-\log p(y|x; \theta)$ of (2), which is equal to the L_1 loss (1) up to an additive constant.

In the aforementioned Laplacian model (2), derived from the L_1 loss (1), only the mean $g(x)$ is estimated from the LR image. Thus, the model assumes that the variance, which reflects the possible variability of each pixel, remains constant. This assumption is however, not accurate. Indeed, super-resolving a constant blue sky is substantially *easier* than estimating the pixel values of a highly textured region, such as the foliage of a tree. In the former case, the predicted pixels should have low variance, while the latter has high variability, corresponding to different possible textures of foliage. For a Laplace distribution, we can encode the variability in the scale parameter b, which is proportional to the standard deviation. By predicting the scale parameter $b(x) \in \mathbb{R}^{H \times W \times C}$ for each pixel, we can learn a more accurate distribution that also quantifies some aspect of the ill-posed nature of the SR problem.

We easily extend our model with the scale parameter prediction by modifying our function f as,

$$z = f(y; x) = \frac{y - g(x)}{b(x)} . \quad (3)$$

Since this yields a Laplace distribution $p(y|x; \theta) = \mathcal{L}(y; g(x), b(x))$, we achieve the NLL,

$$-\log p(y|x; \theta) \propto \left\| \frac{y - g(x)}{b(x)} \right\|_1 + \sum_{ijc} \log b(x)_{ijc} . \quad (4)$$

In practice, we can easily modify an SR network to jointly estimating the mean $g(x)$ and scale $b(x)$ by doubling the number of output dimensions. The loss (4) stimulates the network to predict larger scale values $b(x)$ for 'uncertain' pixels, that are likely to have large error $\|y - g(x)\|$. In principle, (4) thus extends the L_1 objective to better cope with the ill-posed nature of the SR problem by predicting a more flexible distribution of the HR image. In the next section, we will further generalize the objectives (1), (4) through normalizing flows, to achieve an even more flexible fidelity loss.

3.2. Generalizing the L_1 Loss With Flows

To capture the probability distribution of the error between the prediction $g(x)$ and ground-truth y, we also need to consider spatial dependencies. Neighboring pixels are generally highly correlated in natural images. Indeed, to create coherent textures, even long-range correlations need to be considered. However, the L_1 loss (1) and its extension (4) assume each pixel in y to be conditionally independent given x. In fact, sampling from the predicted conditional distribution $p(y|x; \theta)$ is equivalent to simply adding Laplacian white noise to the predicted mean $g(x)$. In super-resolution, we strive to create fine textures and details. To achieve this, the predictive distribution $p(y|x; \theta)$ must capture complex correlations in the image space.

In this paper, we generalize the L_1 loss (4) with the aim of achieving a more flexible objective, better capturing the ill-posed setting. That is done through the probabilistic interpretation discussed in Sec. 3.1. We observe that the function f introduced in Sec. 3.1 corresponds to a one-layer conditional normalizing flow with a Laplacian latent space. We can thus generalize this setting by constructing deeper flow networks f. While prior works [3, 30, 33, 40] investigate conditional flows for SR as a replacement for adversarial losses, we see it as a generalization of the fidelity-based L_1 loss. With this view, we aim to find a fidelity-based objective better suited for ill-posed problems and, therefore, more effectively combined with adversarial losses.

The purpose of the function f is to map the HR-LR pair (y, x) to a latent space $z \sim p_z$, which follows a simple distribution. By increasing the depth and complexity of the flow f, more flexible conditional densities, and therefore also NLL-based losses, are achieved. In the general case, we let flow f to be conditioned on the embedding $E(x)$ of the LR image x as $f(y; x) = f(y; E(x))$. In fact, the network E can be seen as predicting the *parameters* of the conditional distribution $p(y|x; \theta) = p(y|E(x); \theta)$. In this view, the embedding $E(x)$ generalizes the purpose of the SR network $g(x)$, which predicts the mean of the Laplace distribution in the L_1 case (1). In the general Laplace case (3), the LR embedding network needs to generate both the mean and the scale $E(x) = (g(x), b(x))$. Thanks to the flexibility of conditional flow layers, we can however, still use the underlying image representation of any standard SR architecture as E. For example, we generate the embedding E by concatenating a series of intermediate feature maps from, *e.g.*, the RRDB [39], or the RCAN [48] architecture. For simplicity, we often drop the explicit dependence on E

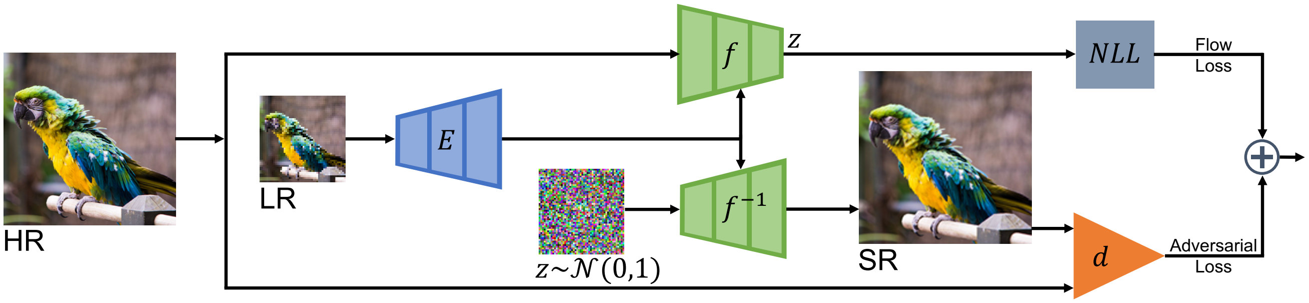

Figure 3. Overview of our super-resolution approach. Our flow-based NLL loss replaces the often used L_1 loss. We accomplish this by encoding the LR image with the network $E(x)$. This conditions the flow f, which encodes the GT image y. From that, we obtain the NLL loss that drives the SR fidelity. We combine this with a standard adversarial loss, calculated using a discriminator d.

in the flow f and simply write $f(y;x)$.

In order for $f(y;x)$ to be a valid conditional flow network, we need to preserve invertibility in the first coordinate. Under this condition, the conditional density is derived using the change of variable formula [9, 21, 30] as,

$$p_{y|x}(y|x,\theta) = p_z\big(f_\theta(y;x)\big)\left|\det \frac{\partial f_\theta}{\partial y}(y;x)\right| . \tag{5}$$

The latent space prior $z \sim p_z$ is set to a simple distribution, *e.g*. standard Gaussian or Laplacian. The second factor in (5) is the resulting volume scaling, given by the determinant of the Jacobian $\frac{\partial f_\theta}{\partial y}$. We can easily draw samples from the model by inverting the flow as $y = f^{-1}(z;x)$, $z \sim p_z$. The network f^{-1} thus transforms a simple distribution p_z to capture the complex correlations in the output image space.

The NLL training objective is obtained by applying the negative logarithm to (5),

$$-\log p_{y|x}(y|x,\theta) = -\log p_z(z) - \log\left|\det \frac{\partial f_\theta}{\partial y}(y;x)\right| \tag{6a}$$

$$= -\log p_z(z) - \sum_{n=0}^{N-1} \log\left|\det \frac{\partial f_\theta^n}{\partial h^n}(h^n; E(x))\right| , \tag{6b}$$

where $z = f_\theta(y;x)$. In the second equality, we have decomposed f_θ into the sequence of N flow layers $h^{n+1} = f_\theta^n(h^n; E(x))$, with $h^0 = y$ and $h^N = z$. This allows for efficient computation of the log-determinant term.

We can now derive that the flow objective (6) generalizes the scaled L_1 loss (4) and thereby also the standard L_1 loss (1). By using the function f defined in (3) and the standard Laplacian latent variable $p_z(z) = \mathcal{L}(z;0,1)$, we derive the first term in (4) is by inserting (3) into the first term $-\log p_z(z)$ in (6a). For the second term, we first immediately obtain the Jacobian of (3) as a diagonal matrix $\frac{\partial f}{\partial y} = \operatorname{diag}(\frac{1}{b(x)})$ with elements $\frac{1}{b(x)_{ijk}}$. Inserting this result into the log-determinant term in (6a) yields the second term in (4). A more detailed derivation is provided in the supplementary material. Next, we employ the flow-based fidelity objective in a full super-resolution framework by combining it with adversarial losses.

3.3. Flow-Fidelity with Adversarial Losses

The introduction of adversarial losses [23] pioneered a new direction in super-resolution, aiming to generate perceptually pleasing HR outputs from the natural image manifold. In order to achieve this, the adversarial loss needs to be combined with fidelity-based objectives, ensuring that the generated SR image is close to the HR ground-truth. Therefore, SRGAN [23] and later works [2, 12, 14, 17, 26, 35] most typically combine the adversarial loss with the L_1 objective. However, these two objectives are fundamentally conflicting. Unlike the L_1 loss that pulls the super-resolution towards the mean of all plausible manifestations, the adversarial loss forces the generator to choose exactly one image of the natural image manifold. Hence, the adversarial objective ideally assigns a low loss on all natural image patches. In contrast, such predictions generate a high L_1 loss since it prefers the blurry average of plausible predictions. We aim to resolve this issue by replacing the L_1 loss with the aforementioned flow-based generalizations.

The flow can learn a more flexible conditional distribution $p(y|x;\theta)$ of the HR y. It therefore better spans the natural image manifold while simultaneously encouraging consistency with the input LR image x. The NLL loss (6) of the flow distribution does therefore not penalize patches from the natural image manifold to the same extent. That allows the adversarial objective to drive the generated SR images towards perceptually pleasing results without being penalized by the fidelity-based loss. Conversely, the flow-based fidelity loss allows the network to learn from the single provided ground-truth HR image y, without reducing perceptual quality or incurring a higher adversarial loss.

Interestingly, the flow network $y = f_\theta^{-1}(z;x)$, $z \sim p_z$ can also be seen as stochastic generator for the adversarial learning. While stochastic generators are fundamental to unconditional Generative Adversarial Networks (GANs) [13], deterministic networks are most common in the conditional setting, including super-resolution. In fact, GANs are well known to be highly susceptible to mode collapse in the conditional setting [16, 32]. In contrast, flows are highly resistant to mode collapse due to the bijective constraint on

f_θ. This is highly important for ill-posed problems such as SR, where we ideally want to span the space of possible predictions. However, it is important to note that the flow is not merely a generator, as in the standard GAN setting. The flow itself also serves as a flexible loss function (6).

Formally, we add the adversarial loss on samples generated by the flow network f. Let d_ϕ be the discriminator with parameters ϕ. For one LR-HR pair (x, y) and random latent sample $z \sim p_z$, we consider the adversarial loss

$$L_{\text{adv}} = \log\big(1 - d_\phi\big(f_\theta^{-1}(z; x)\big)\big) + \log\big(d_\phi(y)\big)\,. \qquad (7)$$

The loss (7) is minimized w.r.t. the flow and LR encoder parameters θ and maximized w.r.t. the discriminator parameters ϕ. In general, any other variant of adversarial loss (7) can be employed. During training, we employ a linear combination of the NLL loss (6) and (7). Our training procedure is detailed in Sec. 3.5.

3.4. Conditional Flow Architecture

Our full approach, depicted in Fig. 3, consists of the super-resolution network E, the flow network f and the discriminator d. We construct our conditional flow network f based on [30] and use the same settings, where not mentioned otherwise. It is based on Glow [21] and RealNVP [9]. It employs a pyramid structure with L scales, each halving the previous layer's spatial size using a squeeze layer and, depending on the number of channels, also bypassing half of the activations directly to the NLL calculation. We use 3, 4, and 4 scale levels for $4\times$, $6\times$ and $8\times$ respectively, each consisting of a series of K flow steps.

Each Flow-Step consists of a sequence of four layers. In encoding direction, we first employ the ActNorm [21] to normalize the activations using a learned channel-wise scale and bias. To establish information transfer across the channel dimension, we then use an invertible 1×1 convolution [21]. The following layers condition the flow on the LR image similar to [30]. First, the Conditional Affine Coupling [9, 30], partitions the channels into two halves. The first half is used as input, together with the LR encoding $E(x)$, to a 3-layer convolutional network module, which predicts the element-wise scale and bias for the second half. This module adds non-linearities and spatial dependencies to the flow network while ensuring easy invertibility and tractable log-determinants. Secondly, the Affine Image Injector is applied, which transforms all channels conditioned on the low-resolution encoding $E(x)$.

Instead of the learnable 1×1 convolutions used in [21, 30], we use constant orthonormal matrices that are randomly sampled at start of the training. We found this to significantly improve training speed while ensuring better stability due to these layers' perfect conditioning. When combined with an adversarial loss, the flow network operates in both the encode f_θ and decode f_θ^{-1} direction during training. To ensure stability during training in both directions, we reparametrize the prediction of the multiplicative unit in the conditional affine coupling layer. In particular, we predict the multiplicative factor as $s = \text{Sigmoid}(\tilde{s})^{-1}$, where $\tilde{s}$ is the unconstrained prediction stemming from the convolutional module in the coupling.

Super-Resolution embedding network $E(x)$: Our flow-based objective is designed as a replacement of L_1 loss. Our formulation is therefore agnostic to the architecture underlying SR embedding network E. We use the popular RRDB [39] SR network as our encoder E. Instead of outputting the final RGB SR image, these networks predict a rich embedding of the LR image. We obtain this in practice by simply concatenating the underlying feature activations at the intermediate RRDB blocks 1, 4, 6 and 8.

Discriminator: We use the VGG-based network from [39] as a discriminator. Since we generate stochastic SR samples during training, we found it beneficial to reduce the discriminator's capacity to ensure a balanced adversarial objective. We, therefore, reduce the internal channel dimension of the discriminator from 64 to 16.

3.5. Training Details

Our approach is trained by a weighted combination of the NLL loss (6) for fidelity and the adversarial loss (7) to increase perceptual quality. We consider the standard bicubic setting in our experiments, where the LR is generated with the MATLAB bicubic downsampling kernel. In particular, we train for both $4\times$ and the challenging $8\times$ SR scenario. We first train the networks E and f using only the flow NLL loss for 200k iterations using initial learning rates of 10^{-5} for $4\times$ and 10^{-6} for $8\times$, which are then decreased step-wise. We fine-tune the network with the adversarial loss for 200k iterations and select the checkpoint with lowest LPIPS [46] as measured the training set. We employ the Adam [20] optimizer. As in [30] we add uniformly distributed noise with a strength of $\frac{1}{32}$ of the signal range to the ground-truth HR. Our network is trained on HR patches of 160×160 pixels for $4\times$ and $8\times$ and 144×144 pixels for $6\times$. In principle, our framework can employ any adversarial loss formulation. To allow for a direct comparison with the popular state-of-the-art network ESRGAN [39], we employ the same relativistic adversarial formulation. For $4\times$ and $6\times$ SR, we weight the adversarial loss with a factor of 10^{-2} and use a discriminator learning rate of 10^{-3}. For $8\times$, we use 0.1 and 10^{-4} respectively We use the same training data employed by ESRGAN [39], consisting of the DF2K dataset. It comprises 2650 training images from Flickr2K [25] and 800 training images from the DIV2K [1] dataset.

AdFlow compared to	4× DIV2K	 BSD	 Urban	6× DIV2K	 BSD	 Urban	8× DIV2K	 BSD	 Urban
BaseFlow	62.1% ± 2.2	68.3% ± 2.4	74.2% ± 2.1	73.4% ± 2.0	80.7% ± 1.8	82.9% ± 1.7	69.2% ± 2.1	73.1% ± 2.0	78.2% ± 1.9
SRFlow	60.1% ± 2.2	67.2% ± 2.4	66.3% ± 2.2	-	-	-	66.2% ± 2.1	67.2% ± 2.1	71.8% ± 2.0
RankSRGAN	56.9% ± 2.2	54.8% ± 2.6	67.5% ± 2.2	-	-	-	-	-	-
ESRGAN	56.1% ± 2.2	51.2% ± 2.6	64.5% ± 2.3	57.5% ± 2.3	62.8% ± 2.2	63.8% ± 2.2	49.9% ± 2.3	54.5% ± 2.2	57.1% ± 2.2
Ground Truth	49.0% ± 2.2	25.4% ± 2.2	29.1% ± 2.2	27.4% ± 2.1	8.9% ± 1.3	11.6% ± 1.5	18.3% ± 1.7	4.2% ± 0.9	7.7% ± 1.2

Table 1. Quantitative results of the user study. Each entry is aggregated from 1 500 votes. For each dataset and scale factor, we directly compare AdFlow with each competing method in a pairwise fashion, as detailed in Sec. 4.1. For each compared method (left), we report the proportion of votes *in favor* of AdFlow along with the 95% confidence interval. We indicate if AdFlow is significantly better or worse.

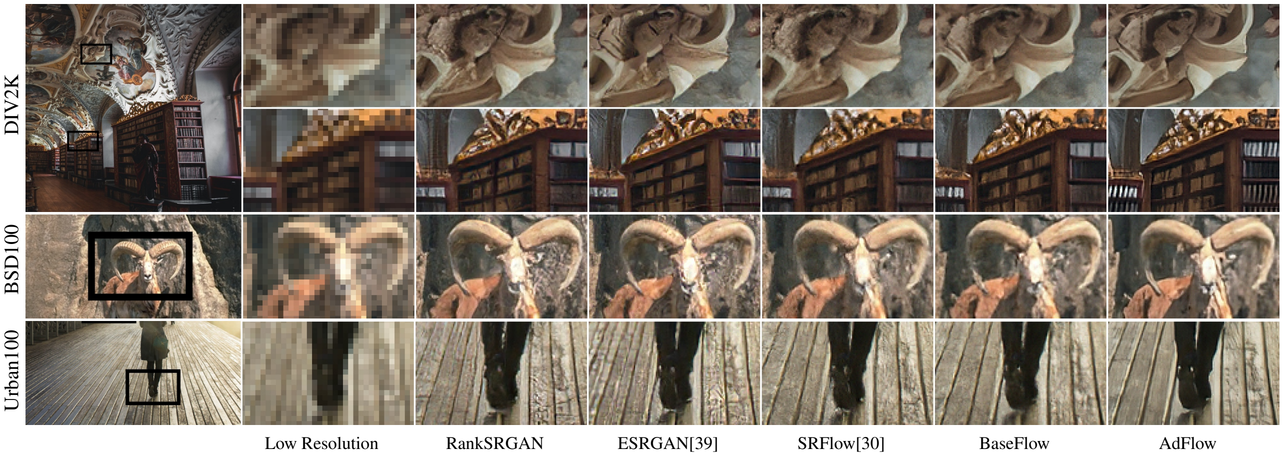

Figure 4. Qualitative comparison with state-of-the-art approaches on the DIV2K (val), BSD100 and Urban100 set for 4× SR.

4. Experiments

We validate our proposed formulation by performing comprehensive experiments on the three standard datasets, namely DIV2K [1], BSD100 [31] and Urban100 [15]. We train our approach for three different scale factors 4×, 6×, and 8×. We term our flow-only baseline as **BaseFlow** and our final method, which also employs adversarial learning, as **AdFlow**. The prediction and evaluation is performed on the full image resolution. For the purely flow-based baseline, we found it best to use a sampling temperature [21, 30] of 0.9. For our final approach with adversarial loss, we found the standard sampling temperature of 1.0 to yield best results. Detailed results and more visual examples are found in the supplementary material.

4.1. State-of-the-Art Comparison

We first compare our approach with state-of-the-art. This work aims to achieve SR predictions that are (i) photo-realistic and (ii) consistent with the input LR image. Since it has become well known [12, 17, 23, 26, 27, 28, 29, 30, 35, 39] that computed metrics, such as PSNR and SSIM, fail to rank methods according to photo-realism (i), we therefore perform extensive user studies as further described below. To assess the consistency of the prediction with the LR input (ii), we first downscale the predicted SR image with the given bicubic kernel and compare the result with the LR input. Their similarity is measured using PSNR, and we therefore refer to this metric as LR-PSNR. The LR consistency penalizes hallucinations and artifacts that cannot be explained from the input image.

User studies: We compare the photo-realism of our AdFlow with other methods in user studies. The user is shown the full low-resolution image where a randomly selected region is marked with a bounding box. Next to this image, two different super-resolutions, or "zooms", of the marked region, are displayed. The user is asked to select "Which image zoom looks more realistic?". In this manner, the user evaluates the photo-realism of our AdFlow versus each compared method. To obtain an unbiased opinion, the methods were anonymized to the user and shown in a different random order for each crop. In each study, we evaluate 3 random crops for each of the 100 images in a dataset (DIV2k, BSD100, or Urban100). We use 5 different users for every study, resulting in 1500 votes per method-to-method comparison in each dataset and scale factor. The full user study is shown in Tab. 1, thus collects over 50 000 votes. Further details are provided in the supplement.

Methods: We compare our approach with state-of-the-art approaches for photo-realistic super-resolution: ESRGAN [39], RankSRGAN [47], and SRFlow [30]. For the two latter approaches, we use the publicly available code and trained models (4× for RankSRGAN, 4× and 8× for SRFlow). In addition to the publicly ESRGAN model for 4× SR, we train models for 6× and 8× SR using the code provided by the authors. All compared methods are trained

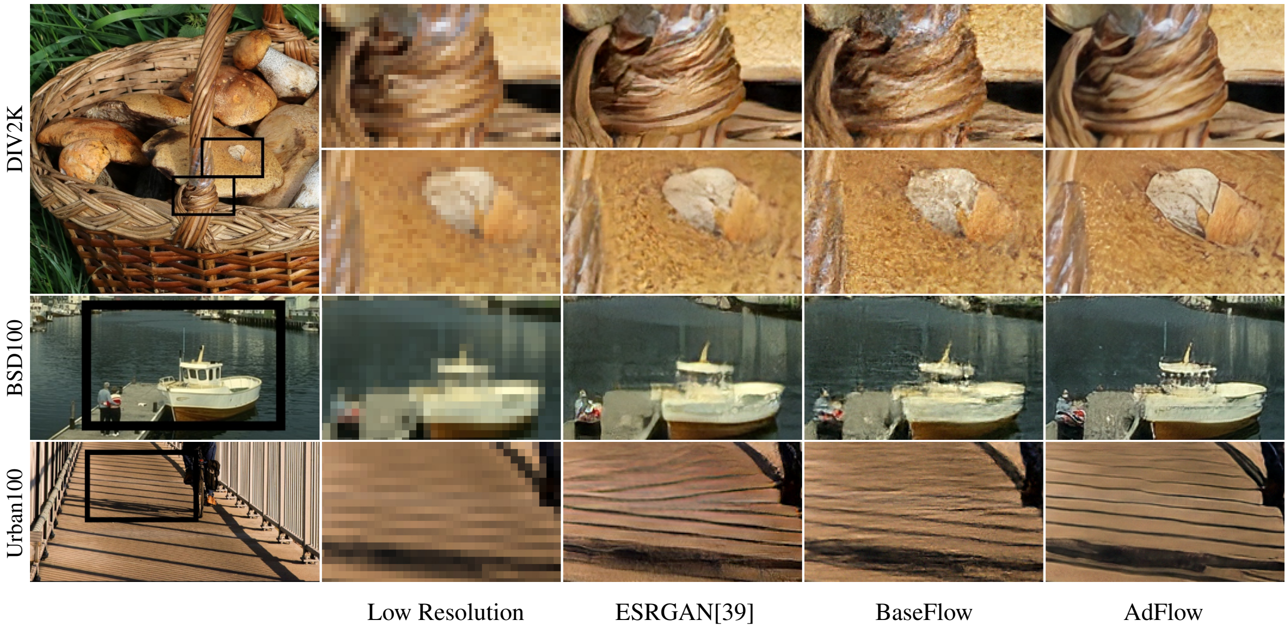

Figure 5. Qualitative comparison with state-of-the-art approaches on the DIV2K (val), BSD100 and Urban100 set for 6× SR.

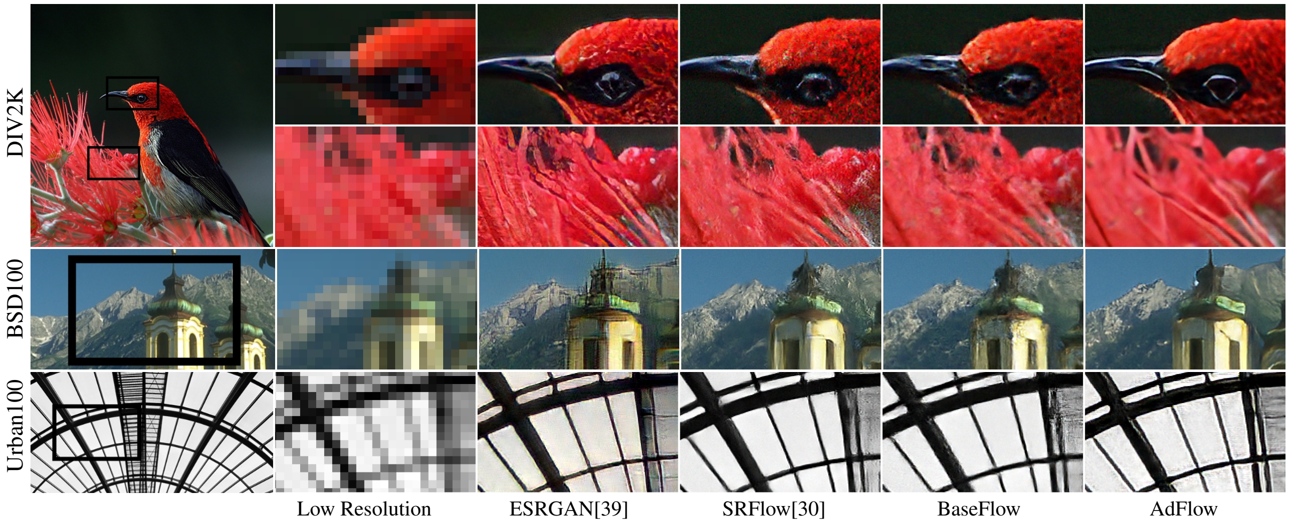

Figure 6. Qualitative comparison with state-of-the-art approaches on the DIV2K (val), BSD100 and Urban100 set for 8× SR.

on the same training set, namely DF2k [25].

Results: The results of our user study are given in Table. 1. The first number represents the ratio of votes in favour of AdFlow and the second the 95% confidence interval. AdFlow outperforms all other methods with significance level 95% in all datasets except for one case. Interestingly, it almost matches the realism of the ground-truth on the DIV2k dataset. The visual results in Fig. 4 show that our AdFlow generates sharp and realistic textures and structures. In contrast, ESRGAN frequently generates visible artifacts while SRFlow and BaseFlow achieve less sharp results. While RankSRGAN experiences fewer artifacts compared to ESRGAN, its predictions are less sharp compared to AdFlow.

The results for higher scale factors 6× and 8× show a similar trend as seen in Fig. 5 and 6. Our approach consistently outperforms the purely flow-based approaches BaseFlow and SRFlow for all scale factors and datasets by over 20% of the votes. As seen in the visual examples, particu-

		4×			6×			8×		
	Method	DIV2K	BSD	Urban	DIV2K	BSD	Urban	DIV2K	BSD	Urban
pure Flow	BaseFlow	49.9	**49.9**	**49.5**	**48.3**	**48.6**	**47.9**	**49.8**	50.2	**48.7**
	SRFlow	**50.0**	**49.9**	**49.5**	-	-	-	49.0	**51.0**	48.1
Adv. combo	RankSRGAN	42.3	41.7	39.9	-	-	-	-	-	-
	ESRGAN	39.0	37.7	36.8	33.2	32.8	30.9	31.3	31.7	28.9
	AdFlow	**45.2**	**45.6**	**43.4**	**37.5**	**38.5**	**36.0**	**46.0**	**46.8**	**42.1**

Table 2. Consistency to the input in terms of LR-PSNR (dB) on the DIV2K (val), BSD100 and Urban100 datasets. We compare for methods that employ adversarial loss (bottom). While purely Flow based methods (top) achieve high LR-PSNR, they have worse perceptual quality (Tab. 1). AdFlow outperforms other methods using adversarial loss for LR consistency significantly.

larly for 6× and 8×, the flow-based approaches often generate strong high-frequency artifacts. In contrast, our AdFlow generates structured and crips textures attributed to the adversarial loss. Compared to ESRGAN, which combines L_1 with adversarial loss, AdFlow produces generally sharper results and has no visible color shift, as seen in Fig. 6. Interestingly, AdFlow demonstrates substantially better generalization to the BSD100 and Urban100 datasets than ESRGAN, as shown in the user study in Table 1. This indicates that ESRGAN tends to overfit to the DIV2k distribution. Qualitative examples for 4×, 6×, and 8× are shown in Fig. 4, 5, and 6, respectively.

We report the LR-PSNR for all datasets and scale factors in Tab. 2. ESRGAN and RankSRGAN obtain poor LR consistency across all datasets, as shown in Tab. 2. AdFlow gains 4.3dB - 15.1dB in LR-PSNR over ESRGAN, indicating less hallucination artifacts and color shift. By employing flow-based fidelity instead of L_1, AdFlow achieves superior photo-realism while ensuring high LR consistency.

4.2. Analysis of Flow-based Fidelity Objective

Here, we analyze the impact of generalizing the L_1 loss towards a gradually more flexible flow-based NLL objective. This is done by increasing the number of flow steps K per level inside the flow architecture. We train and evaluate our AdFlow with different depths K for 8× SR. Due to the difficulty and cost of running a large number of user studies, we here use the learned LPIPS [46] distance as a surrogate to assess photo-realism. In Fig. 7 we plot the LPIPS and LR-PSNR on the DIV2K validation set w.r.t. the number of flow-steps K. We also include the results obtained by the L_1 loss, which is an even simpler one-layer flow loss, as discussed in Sec. 3.1 and 3.2. Note that these results correspond to the standard RRDB and ESRGAN, respectively.

As we increase the depth K of the flow network f, the LPIPS decreases while the LR-PSNR increases. This indicates an improvement in perceptual quality and low-resolution consistency. This trend also holds when starting from the L_1 NLL objective. Note that the brief increase in LPIPS is explained by the added stochasticity when transitioning from the L_1 to the $K = 1$ flow. Indeed, a too shallow flow network does not capture rich enough spatial correlations in order to generate more natural samples. However, already at $K = 2$, the flow-based generalization outperforms the L_1 in LPIPS. Increasing the flexibility of the NLL-based fidelity loss, starting from L_1, thus benefits perceptual quality and consistency. This strongly indicates that a flow-based fidelity objective alleviates the conflicts between the adversarial loss and L_1 loss.

Adv. Loss	Affine Coup.	Rand. Rot.	Coupl. Mult.	Percept. Loss	LPIPS ↓	LR-PSNR ↑
					0.349	39.76
✓					0.337	34.85
	✓				0.253	50.16
✓	✓				-	-
	✓	✓			0.254	50.19
✓	✓	✓			-	-
	✓	✓	✓		0.253	49.78
✓	✓	✓	✓		0.253	47.54
✓	✓	✓	✓	✓	0.270	47.35

Table 3. Ablation of architecture choice for adversarial loss (Adv.), the use of Affine Couplings (Aff. Coup.), Random Depth-Wise Rotation (Rand. Rot.), Decode Multiplication for the Affine Couplings (Decode Mult.) and perceptual loss (Percept.) [23].

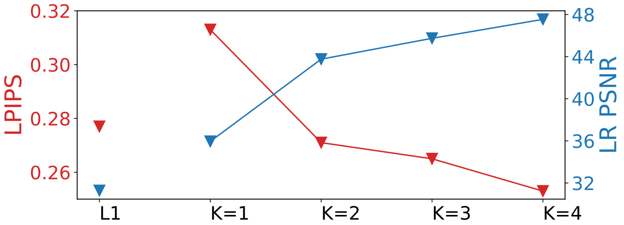

Figure 7. Analysis of the input consistency LR-PSNR and perceptual quality LPIPS for different numbers of Flow Steps K.

4.3. Ablation of Flow Architecture

In Tab. 3 we show results of our ablative experiments for 8× SR on DIV2k. First, we ablate the use of Conditional Affine Couplings. Removing this layer (top row) results in a conditionally linear flow, thereby radically limiting its expressiveness. This leads to a substantially worse LPIPS and LR-PSNR, demonstrating the importance of a flexible flow network. Second, we replace the learnable 1×1 convolutions with fixed random rotation matrices (Rand. Rot.). While widely preserving the quality in all metrics, it reduces the training time by 37.1%. Next, we consider the reparametrization of the coupling layers (Coupl. Mult.). We found this to be critical for training stability when combined with the adversarial loss. Lastly, we investigate the use of the VGG-based perceptual loss [23] that is commonly used in SR methods. It is generally employed as a more perceptually inclined fidelity loss to complement the L_1 objective. However, we found the perceptual loss not to be beneficial. This indicates that the more flexible flow-based fidelity loss can also effectively replace the VGG loss.

5. Conclusion

We explore conditional flows as a generalization of the L_1 loss in the context of photo-realistic super-resolution. In particular, we tackle the conflicting objectives between L_1 and adversarial losses. Our flow-based alternatives offer both improved fidelity to the input low-resolution and a higher degree of flexibility. Extensive user studies clearly demonstrate the advantages of our approach over state-of-the-art on three datasets and scale factors. Lastly, our experimental analysis brings new insights into the learning of super-resolution methods, paving for further explorations in the pursuit of more powerful learning formulations.

Acknowledgements: This work was supported by the ETH Zürich Fund (OK), a Huawei Technologies Oy (Finland) project and an Nvidia GPU grant and AWS.

References

[1] Eirikur Agustsson and Radu Timofte. Ntire 2017 challenge on single image super-resolution: Dataset and study. In *CVPR Workshops*, 2017.

[2] Namhyuk Ahn, Byungkon Kang, and Kyung-Ah Sohn. Image super-resolution via progressive cascading residual network. In *CVPR*, 2018.

[3] Lynton Ardizzone, Carsten Lüth, Jakob Kruse, Carsten Rother, and Ullrich Köthe. Guided image generation with conditional invertible neural networks. *CoRR*, abs/1907.02392, 2019.

[4] Yuval Bahat and Tomer Michaeli. Explorable super resolution. In *CVPR*, 2020.

[5] Yochai Blau, Roey Mechrez, Radu Timofte, Tomer Michaeli, and Lihi Zelnik-Manor. The 2018 PIRM challenge on perceptual image super-resolution. In Laura Leal-Taixé and Stefan Roth, editors, *Computer Vision - ECCV 2018 Workshops - Munich, Germany, September 8-14, 2018, Proceedings, Part V*, volume 11133 of *Lecture Notes in Computer Science*, pages 334–355. Springer, 2018.

[6] Yochai Blau and Tomer Michaeli. The perception-distortion tradeoff. In *CVPR*, pages 6228–6237, 2018.

[7] Dengxin Dai, Radu Timofte, and Luc Van Gool. Jointly optimized regressors for image super-resolution. *Comput. Graph. Forum*, 34(2):95–104, 2015.

[8] Laurent Dinh, David Krueger, and Yoshua Bengio. NICE: non-linear independent components estimation. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Workshop Track Proceedings*, 2015.

[9] Laurent Dinh, Jascha Sohl-Dickstein, and Samy Bengio. Density estimation using real NVP. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[10] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Learning a deep convolutional network for image super-resolution. In *ECCV*, pages 184–199, 2014.

[11] Chao Dong, Chen Change Loy, Kaiming He, and Xiaoou Tang. Image super-resolution using deep convolutional networks. *TPAMI*, 38(2):295–307, 2016.

[12] Manuel Fritsche, Shuhang Gu, and Radu Timofte. Frequency separation for real-world super-resolution. In *2019 IEEE/CVF International Conference on Computer Vision Workshops, ICCV Workshops 2019, Seoul, Korea (South), October 27-28, 2019*, pages 3599–3608. IEEE, 2019.

[13] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[14] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Deep back-projection networks for super-resolution. In *CVPR*, 2018.

[15] Jia-Bin Huang, Abhishek Singh, and Narendra Ahuja. Single image super-resolution from transformed self-exemplars. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5197–5206, 2015.

[16] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A. Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, pages 5967–5976, 2017.

[17] Xiaozhong Ji, Yun Cao, Ying Tai, Chengjie Wang, Jilin Li, and Feiyue Huang. Real-world super-resolution via kernel estimation and noise injection. In *The IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2020.

[18] Deokyun Kim, Minseon Kim, Gihyun Kwon, and Daeshik Kim. Progressive face super-resolution via attention to facial landmark. In *arxiv*, volume abs/1908.08239, 2019.

[19] Jiwon Kim, Jung Kwon Lee, and Kyoung Mu Lee. Accurate image super-resolution using very deep convolutional networks. In *CVPR*, 2016.

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

[21] Diederik P. Kingma and Prafulla Dhariwal. Glow: Generative flow with invertible 1x1 convolutions. In *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, 3-8 December 2018, Montréal, Canada*, pages 10236–10245, 2018.

[22] Wei-Sheng Lai, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep laplacian pyramid networks for fast and accurate super-resolution. In *CVPR*, 2017.

[23] Christian Ledig, Lucas Theis, Ferenc Huszár, Jose Caballero, Andrew Cunningham, Alejandro Acosta, Andrew P Aitken, Alykhan Tejani, Johannes Totz, Zehan Wang, et al. Photo-realistic single image super-resolution using a generative adversarial network. *CVPR*, 2017.

[24] Jingyun Liang, Andreas Lugmayr, Kai Zhang, Martin Danelljan, Luc Van Gool, and Radu Timofte. Hierarchical conditional flow: A unified framework for image super-resolution and image rescaling. In *ICCV*, 2021.

[25] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. *CVPR*, 2017.

[26] Andreas Lugmayr, Martin Danelljan, and Radu Timofte. Unsupervised learning for real-world super-resolution. In *ICCV Workshops*, 2019.

[27] Andreas Lugmayr, Martin Danelljan, and Radu Timofte. Ntire 2021 learning the super-resolution space challenge. In *CVPRW*, 2021.

[28] Andreas Lugmayr, Martin Danelljan, Radu Timofte, et al. Aim 2019 challenge on real-world image super-resolution: Methods and results. In *ICCV Workshops*, 2019.

[29] Andreas Lugmayr, Martin Danelljan, Radu Timofte, et al. Ntire 2020 challenge on real-world image super-resolution: Methods and results. *CVPR Workshops*, 2020.

[30] Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. Srflow: Learning the super-resolution space with normalizing flow. In *ECCV*, 2020.

[31] D. Martin, C. Fowlkes, D. Tal, and J. Malik. A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In *Proc. 8th Int'l Conf. Computer Vision*, volume 2, pages 416–423, July 2001.

[32] Michaël Mathieu, Camille Couprie, and Yann LeCun. Deep

multi-scale video prediction beyond mean square error. In *ICLR*, 2016.

[33] Albert Pumarola, Stefan Popov, Francesc Moreno-Noguer, and Vittorio Ferrari. C-flow: Conditional generative flow models for images and 3d point clouds. In *CVPR*, pages 7949–7958, 2020.

[34] Danilo Jimenez Rezende and Shakir Mohamed. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, pages 1530–1538, 2015.

[35] Mehdi S. M. Sajjadi, Bernhard Schölkopf, and Michael Hirsch. Enhancenet: Single image super-resolution through automated texture synthesis. In *IEEE International Conference on Computer Vision, ICCV 2017, Venice, Italy, October 22-29, 2017*, pages 4501–4510. IEEE Computer Society, 2017.

[36] Libin Sun and James Hays. Super-resolution from internet-scale scene matching. In *ICCP*, 2012.

[37] Radu Timofte, Vincent De Smet, and Luc Van Gool. A+: Adjusted anchored neighborhood regression for fast super-resolution. In *ACCV*, pages 111–126. Springer, 2014.

[38] Radu Timofte, Vincent De Smet, and Luc Van Gool. Anchored neighborhood regression for fast example-based super-resolution. In *ICCV*, pages 1920–1927, 2013.

[39] Xintao Wang, Ke Yu, Shixiang Wu, Jinjin Gu, Yihao Liu, Chao Dong, Chen Change Loy, Yu Qiao, and Xiaoou Tang. Esrgan: Enhanced super-resolution generative adversarial networks. *ECCV*, 2018.

[40] Christina Winkler, Daniel E. Worrall, Emiel Hoogeboom, and Max Welling. Learning likelihoods with conditional normalizing flows. *arxiv*, abs/1912.00042, 2019.

[41] Valentin Wolf, Andreas Lugmayr, Martin Danelljan, Luc Van Gool, and Radu Timofte. Deflow: Learning complex image degradations from unpaired data with conditional flows. In *CVPR*, 2021.

[42] Mingqing Xiao, Shuxin Zheng, Chang Liu, Yaolong Wang, Di He, Guolin Ke, Jiang Bian, Zhouchen Lin, and Tie-Yan Liu. Invertible image rescaling. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision - ECCV 2020 - 16th European Conference, Glasgow, UK, August 23-28, 2020, Proceedings, Part I*, volume 12346 of *Lecture Notes in Computer Science*, pages 126–144. Springer, 2020.

[43] Chih-Yuan Yang and Ming-Hsuan Yang. Fast direct super-resolution by simple functions. In *ICCV*, pages 561–568, 2013.

[44] Jianchao Yang, John Wright, Thomas S. Huang, and Yi Ma. Image super-resolution as sparse representation of raw image patches. In *CVPR*, 2008.

[45] Jianchao Yang, John Wright, Thomas S. Huang, and Yi Ma. Image super-resolution via sparse representation. *IEEE Trans. Image Processing*, 19(11):2861–2873, 2010.

[46] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. *CVPR*, 2018.

[47] Wenlong Zhang, Yihao Liu, Chao Dong, and Yu Qiao. Ranksrgan: Generative adversarial networks with ranker for image super-resolution. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 3096–3105, 2019.

[48] Yulun Zhang, Kunpeng Li, Kai Li, Lichen Wang, Bineng Zhong, and Yun Fu. Image super-resolution using very deep residual channel attention networks. In *ECCV*, volume 11211 of *Lecture Notes in Computer Science*, pages 294–310. Springer, 2018.

[49] Hang Zhao, Orazio Gallo, Iuri Frosio, and Jan Kautz. Loss functions for image restoration with neural networks. *IEEE Trans. Computational Imaging*, 3(1):47–57, 2017.