# Arbitrary-steps Image Super-resolution via Diffusion Inversion

## Supplementary Material

This supplemental material mainly contains:

- Sec. A discusses the selection of the number of sampling steps.
- Performance comparison of *InvSR* with various base diffusion models in Sec. B.1.
- Ablation studies on the intermediate noise prediction model in Sec. B.2.
- Ablation studies on the loss function in Sec. B.3.
- Discussions on the efficiency and limitation in Sec. B.4.
- Visual comparisons on *ImageNet-Test* dataset in Fig. IV.
- More visual comparisons on real-world examples in Fig. V.

## A. Discussion on Sampling Steps

The proposed method, named *InvSR*, enables a flexible sampling mechanism that allows an arbitrary number of sampling steps. This naturally raises an interesting question: how do we determine an appropriate number of sampling steps for general image super-resolution (SR) tasks? We answer this question from two aspects.

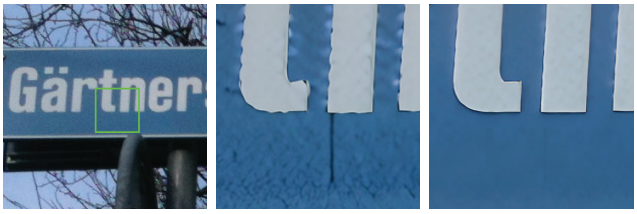
First, as shown in Tables 2 and 3, and Fig. 3 of the main text, *InvSR* achieves promising results with only a single sampling step, evidently outperforming recent state-of-the-art (SotA) one-step methods. Therefore, we recommend setting the sampling steps to one for most real-world applications, effectively balancing efficiency and performance.

Second, we can also adjust sampling steps according to the type of image degradation. Generally, image degradations can be categorized into two main classes: blurriness and noise. As illustrated in Fig. 1 and Fig. II, multi-step sampling would incorrectly amplify noise, leading to undesirable artifacts for images with heavy noise. In contrast, for images primarily suffering from blurriness, multi-step sampling proves beneficial, as it generates more detailed and realistic image structures. In practice, we could first estimate the noise level using some off-the-shelf degradation estimation models, such as Mou *et al*. [2]. Based on the estimated noise level, one can determine whether a one-step or multi-step sampling is more appropriate. In cases where multi-step sampling is favored, the number of sampling steps can be freely adjusted to achieve a satisfactory result.

## B. Experiments

### B.1. Base Diffusion Model

For the pre-trained diffusion models used in *InvSR*, we considered two prevailing variants of Stable Diffusion [3], namely SD-2.0[1] and SD-Turbo[2]. Table I provides a quantitative comparison of *InvSR* equipped with these two base models. When reducing the sampling steps to one, *InvSR* demonstrated similar performance with both SD-2.0 and SD-Turbo. However, in the multi-step sampling scenarios, the model based on SD-Turbo exhibited more stable performance, particularly in terms of reference metrics. Furthermore, a visual comparison under five sampling steps, as illustrated in Fig. I, reveals that the SD-2.0-based model produced noticeable artifacts, aligning with the quantitative results. We thus employed SD-Turbo as our base model throughout this study.




Zoomed LR | *InvSR* with SD-2.0 | *InvSR* with SD-Turbo

Figure I. A typical visual comparison of the proposed *InvSR* based on different diffusion models: SD-2.0 and SD-Turbo. Note that these results are achieved with five sampling steps.

### B.2. Intermediate Noise Prediction

In our proposed diffusion inversion framework, we opt to sample the noise maps randomly rather than employing a noise prediction model for intermediate timesteps. This choice is motivated by the high SNR (signal-to-noise ratio) constraint imposed on the inversion timesteps, as elaborated in Sec. 3.2.3 of the main text. To further validate this choice, we introduced an additional baseline, denoted as "*InvSR-Int*", which integrates an extra noise predictor specifically trained for intermediate timesteps. Table II reports a detailed comparison between *InvSR* and *InvSR-Int*. It can be observed that the performance differences between these two models are negligible. Therefore, we omit the intermediate noise prediction in *InvSR*, further simplifying the overall framework.

### B.3. Loss Functions

In addition to the commonly used $L_2$ loss, we incorporate LPIPS [5] loss and GAN [1] loss to train our noise predictor, as formulated in Eq.(11) of the main text. The hyper-parameters of $\lambda_l$ and $\lambda_g$ are introduced to control the importance of the LPIPS and GAN losses, respectively. Table III

[1] https://huggingface.co/stabilityai/stable-diffusion-2-base
[2] https://huggingface.co/stabilityai/sd-turbo

Table I. Quantitative comparisons of the proposed *InvSR* equipped with two different based models, namely SD-2.0 and SD-Turbo, on the dataset of *ImageNet-Test*.

| Base models | #Steps | Index of the sampled timesteps | Metrics | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | PSNR↑ | SSIM↑ | LPIPS↓ | NIQE↓ | PI↓ | CLIPIQA↑ | MUSIQ↑ |
| SD-Turbo | 5 | {250, 200, 150, 100, 50} | 22.70 | 0.6412 | 0.2844 | 4.8757 | 3.4744 | 0.6733 | 69.8427 |
| SD-2.0 | | | 21.40 | 0.6063 | 0.3274 | 5.1508 | 3.8709 | 0.6467 | 67.6056 |
| SD-Turbo | 3 | {150, 100, 50} | 23.84 | 0.6713 | 0.2575 | 4.2719 | 3.0527 | 0.6823 | 70.4569 |
| SD-2.0 | | | 23.13 | 0.6566 | 0.2776 | 4.2449 | 3.1467 | 0.6722 | 69.5178 |
| SD-Turbo | 1 | {200} | 24.14 | 0.6789 | 0.2517 | 4.3815 | 3.0866 | 0.7093 | 72.2909 |
| SD-2.0 | | | 23.36 | 0.6637 | 0.2647 | 4.3304 | 3.1545 | 0.6969 | 71.4974 |

Table II. Quantitative comparisons of *InvSR* to the baseline method *InvSR-Int* that combines an additional noise predictor for the intermediate timesteps on the dataset of *ImageNet-Test*.

| Methods | #Steps | Index of the sampled timesteps | Metrics | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | | | PSNR↑ | SSIM↑ | LPIPS↓ | NIQE↓ | PI↓ | CLIPIQA↑ | MUSIQ↑ |
| *InvSR* | 5 | {250, 200, 150, 100, 50} | 22.70 | 0.6412 | 0.2844 | 4.8757 | 3.4744 | 0.6733 | 69.8427 |
| *InvSR-Int* | | | 22.70 | 0.6412 | 0.2844 | 4.8785 | 3.4718 | 0.6734 | 69.8466 |

Table III. Quantitative ablation studies on the loss function of Eq.(11) in the main text, wherein the hyper-parameters $\lambda_l$ and $\lambda_g$ control the weight importance of the LPIPS loss and the GAN loss, respectively.

| Methods | Hyper-parameters | | Metrics | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | $\lambda_l$ (LPIPS loss) | $\lambda_g$ (GAN loss) | PSNR↑ | SSIM↑ | LPIPS↓ | NIQE↓ | PI↓ | CLIPIQA↑ | MUSIQ↑ |
| Baseline1 | 0.0 | 0.0 | 26.71 | 0.7365 | 0.2850 | 9.2792 | 6.4147 | 0.6168 | 64.6069 |
| Baseline2 | 2.0 | 0.0 | 26.24 | 0.7274 | 0.2841 | 8.4367 | 5.7973 | 0.6501 | 66.1726 |
| Baseline3 | 0.0 | 0.1 | 24.11 | 0.6809 | 0.2599 | 4.4518 | 3.1229 | 0.7078 | 72.5045 |
| *InvSR-1* | 2.0 | 0.1 | 24.14 | 0.6789 | 0.2517 | 4.3815 | 3.0866 | 0.7093 | 72.2909 |

Table IV. Efficiency comparisons of different methods on the x4 ($128 \rightarrow 512$) SR task, where the runtime results are tested on an NVIDIA A100 GPU with 40GB memory. For diffusion-based SR approaches, the number of sampling steps is annotated in the format of "Method name-Steps".

| Metrics | Methods | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | BSRGAN | RealESRGAN | StableSR-50 | DiffBIR-50 | SeeSR-50 | ResShift-4 | SinSR-1 | OSEDiff-1 | *InvSR*-1 |
| #Params (M) | 16.70 | 16.70 | 152.70 | 385.43 | 751.50 | 118.59 | 118.59 | 8.50 | 33.84 |
| Runtime (ms) | 65 | 65 | 3459 | 7937 | 6438 | 319 | 138 | 176 | 117 |

provides a quantitative comparison of various baseline models under different loss configurations, and Fig. III demonstrates a typical visual example. We can observe that Baseline1 trained solely with the $L_2$-based diffusion loss produces over-smooth outputs, which is consistent with its superior PSNR scores. Incorporating the GAN loss enhances the generation of finer image details but may introduce undesirable artifacts. The addition of LPIPS loss can mitigate these artifacts to a certain extent, striking a balance between perceptual quality and artifact suppression. Therefore, this study uses both LPIPS and GAN losses to achieve optimal performance.

### B.4. Efficiency and Limitation

Table IV lists an efficiency comparison of various methods on the x4 ($128 \rightarrow 512$) SR task. It can be observed that the proposed *InvSR* demonstrates advantages in runtime among one-step diffusion-based approaches. Despite having a larger number of parameters compared to the recent SotA method OSEDiff [4], *InvSR* achieves a 50% reduction in inference time. This is mainly because OSEDiff relies on an additional image captioning model, whereas *InvSR* does not. However, it is noteworthy that *InvSR* still lags behind GAN-based methods in efficiency due to its reliance on the large-scale Stable Diffusion model. To address the high-efficiency demand in real-world applications, future work will explore model quantization techniques to further accelerate the inference speed.

## References


[1] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. *Adv. Neural Inform. Process. Syst.*, 27, 2014. 1

[2] Chong Mou, Yanze Wu, Xintao Wang, Chao Dong, Jian Zhang, and Ying Shan. Metric learning based interactive mod-


Figure II. Qualitative comparisons of the proposed *InvSR* with different sampling steps, where the number of sampling steps is annotated in the format "*InvSR*-Steps". In the first example, mainly degraded by blurriness, multi-step sampling is preferable to single-step sampling as it progressively recovers finer details. Conversely, in the second example with severe noise, a single sampling step is sufficient to achieve satisfactory results, whereas additional steps may amplify the noise and introduce unwanted artifacts. *(Zoom-in for best view)*

(a) Zoomed LR (b) w/o LPIPS, w/o GAN (c) w/o GAN (d) w/o LPIPS (e) *InvSR*-1

Figure III. Visual comparisons of the proposed method with various loss configurations. (a) Zoomed LR image, (b) Baseline1 with $\lambda_l = 0$ and $\lambda_g = 0$, (c) Baseline2 with $\lambda_l = 2.0$ and $\lambda_g = 0$, (d) Baseline3 with $\lambda_l = 0$ and $\lambda_g = 0.1$, (e) recommended settings of $\lambda_l = 2.0$ and $\lambda_g = 0.1$. *(Zoom-in for best view)*

ulation for real-world super-resolution. In *Eur. Conf. Comput. Vis.*, pages 723–740. Springer, 2022. 1

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 10684–10695, 2022. 1

[4] Rongyuan Wu, Lingchen Sun, Zhiyuan Ma, and Lei Zhang. One-step effective diffusion network for real-world image super-resolution. In *Adv. Neural Inform. Process. Syst.*, 2024. 2

[5] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 586–595, 2018. 1

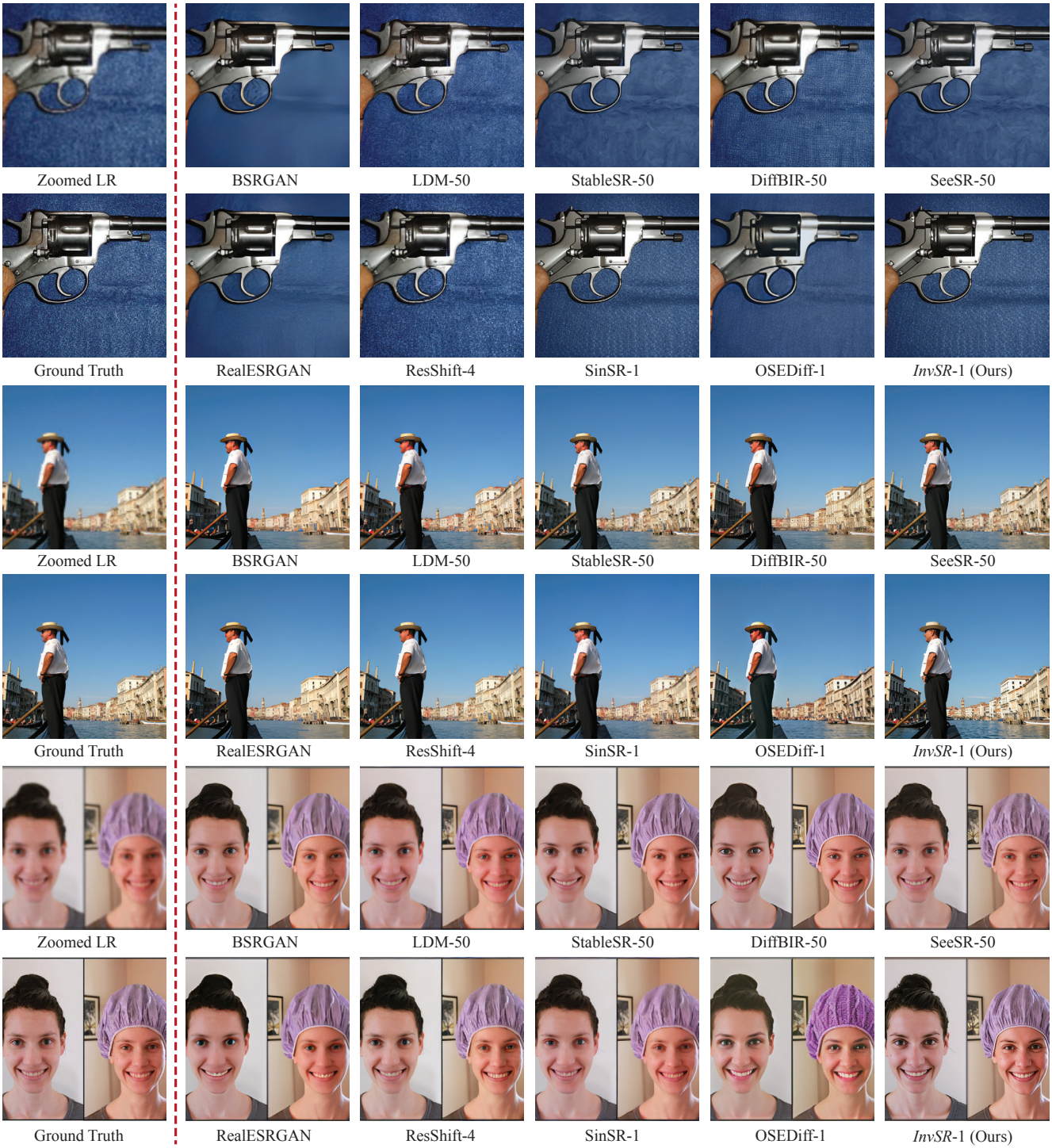



Figure IV. Visual comparisons of various methods on three typical examples from *ImageNet-Test*. For diffusion-based methods, the number of sampling steps is annotated in the format of "Method name-Steps". *(Zoom-in for best view)*


Figure V. Visual comparisons of various methods on four real-world examples from *RealSet80*. For diffusion-based methods, the number of sampling steps is annotated in the format of "Method name-Steps". *(Zoom-in for best view)*