1 Additional Implementation Details

1.1 NC-SDEdit Applied to VE-SDE

Song et al. [8] demonstrated that the noise perturbations used in DDPM [3] and SMLD [7] correspond to discretizations of variance preserving (VP) and variance exploding (VE) SDEs respectively.

Specifically, consider the following stochastic differential equation:

$$d\boldsymbol{x} = \overline{\boldsymbol{f}}(\boldsymbol{x}, t)dt + \bar{g}(t)d\boldsymbol{w}, \tag{1}$$

where $\overline{\boldsymbol{f}} : \mathbb{R}^d \mapsto \mathbb{R}^d$ is the drift coefficient of $\boldsymbol{x}(t)$, $\bar{g} : \mathbb{R} \mapsto \mathbb{R}$ is the diffusion coefficient coupled with the standard d-dimensional Wiener process $\boldsymbol{w} \in \mathbb{R}^d$. By carefully choosing $\bar{f}, \bar{g}$, one can achieve spherical Gaussian distribution as $t \to T$.

For the given forward SDE in Eq. (1), there exists a reversetime SDE running backwards:

$$d\boldsymbol{x} = [\overline{\boldsymbol{f}}(\boldsymbol{x}, t) - \bar{g}(t)^2 \underbrace{\nabla_{\boldsymbol{x}} \log p_t(\boldsymbol{x})}_{\text{score function}}]dt + \bar{g}(t)d\overline{\boldsymbol{w}} \tag{2}$$

where dt is the infinitesimal negative time step, and $\overline{\boldsymbol{w}}$ is the Brownian motion running backwards.

First, by choosing

$$\overline{\boldsymbol{f}}(\boldsymbol{x}, t) = -\frac{1}{2}\beta(t)\boldsymbol{x}, \quad \bar{g}(t) = \sqrt{\beta(t)}, \tag{3}$$

where $0 < \beta(t) < 1$ is a monotonically increasing function of noise scale, one achieves the VP-SDE [3]. On the other hand, VE-SDE choose

$$\overline{\boldsymbol{f}} = \boldsymbol{0}, \quad \bar{g} = \sqrt{\frac{d\left[\sigma^2(t)\right]}{dt}}, \tag{4}$$

where $\sigma(t) > 0$ is again a monotonically increasing function, typically chosen to be a geometric series [7].

VP-SDE can be seen as the continuous version of DDPM [3]. On the other hand, SMLD [7] can be seen as the discrete version of VE-SDE. Specifically, the forward SMLD diffusion step is given by:

$$\boldsymbol{x}_t = \boldsymbol{x}_0 + \sigma_t \boldsymbol{z} \tag{5}$$

where $\sigma_t = \sigma_{\min}\left(\frac{\sigma_{\max}}{\sigma_{\min}}\right)^{\frac{t-1}{T-1}}$, as defined in [8] and $\boldsymbol{z} \sim \mathcal{N}(0, 1)$.

The diffusion process of SDXL-1.0-refiner [5] is constructed through the VE-SDE. therefore, we incorporate the corresponding Noise Calibration algorithm as shown in Algorithm 1.

Algorithm 1 Noise Calibration (VE-SDE)

Input: reference x^r, initial denoising step t_0, diffusion model $\epsilon_\theta(x_t, t)$, iteration steps N, stop frequency ν
$\epsilon_{t_0} \sim \mathcal{N}(0, 1)$
for $n = 1$ **to** N **do**
  $x_{t_0} = x^r + \sigma_{t_0}\epsilon_{t_0}$
  $\hat{x}_0^{t_0} = x_{t_0} - \sigma_{t_0}\epsilon_\theta\left(x_{t_0}, t_0\right)$
  $\epsilon_{t_0} = \epsilon_\theta(x_{t_0}, t_0) + (f_h^\nu(\hat{x}_0^{t_0}) - f_h^\nu(x^r))/\sigma_{t_0}$
end for

1.2 Details on Low-Frequency and High-Frequency Decomposition

To further mitigate the issue of oversmoothed texture, FreeU [6] employ spectral modulation in the Fourier domain to selectively diminish low-frequency components for the skip features. We employ the same method to extract the high-frequency and low-frequency components of the reference reference x^r and the initial estimate $\hat{x}_0^{t_0}$. Taking the extraction of the low-frequency component as an example, mathematically, this operation is performed as follows:

$$\begin{aligned} \mathcal{F}\left(x\right) &= \mathrm{FFT}\left(x\right), \\ \mathcal{F}'\left(x\right) &= \mathcal{F}\left(x\right) \odot \boldsymbol{\beta}_l^\nu, \\ f_l^\nu(x) &= \mathrm{IFFT}\left(\mathcal{F}'\left(x\right)\right), \end{aligned} \tag{6}$$

where FFT($\cdot$) and IFFT($\cdot$) are Fourier transform and inverse Fourier transform. $\odot$ denotes element-wise multiplication, and $\boldsymbol{\beta}_l^\nu$ is a Fourier mask:

$$\boldsymbol{\beta}_l^\nu = \begin{cases} 1 & \text{if } r < \nu, \\ 0 & \text{otherwise,} \end{cases} \tag{7}$$

where r is the radius. ν is the threshold frequency. If you want to extract the high-frequency component, replace $\boldsymbol{\beta}_l^\nu$ in Eq. (6) with:

$$\boldsymbol{\beta}_h^\nu = \begin{cases} 0 & \text{if } r < \nu, \\ 1 & \text{otherwise.} \end{cases} \tag{8}$$

2 Additional Experimental Results

2.1 Performance Demonstration of NC-SDEdit with Different t_0

Fig. 1 shows the enhancement effect of Noise Calibration on the original enhanced results under different initial denoising step t_0 conditions. Specifically, when only using SDEdit for video enhancement, at a small initial denoising step t_0, such as 200 or 400, the enhanced video will have many temporal noise points. When t_0=600, although the noise points basically disappear, content changes begin to appear, such as additional sail. When t_0 continues to increase to 800, content inconsistency continues to increase. However, our method only needs to iterate the initial random noise three times to achieve a significant improvement in content consistency, regardless of the value of t_0.

Fig. 1: Performance Demonstration of NC-SDEdit with Different t_0

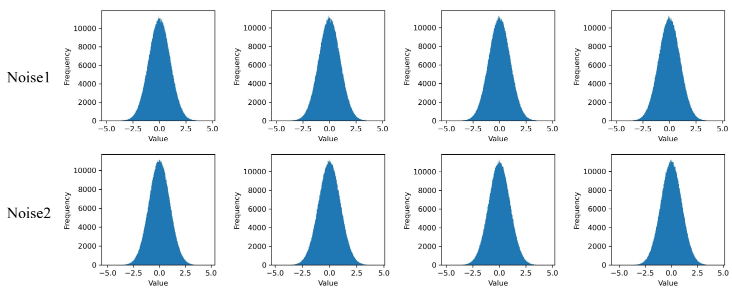

Fig. 2: Display of Distribution Before and After Noise Calibration

2.2 Display of Distribution Before and After Noise Calibration

We randomly selected 4 videos and correspondingly generated 4 noises "Noise1" from a standard normal distribution. As can be seen from Fig. 2, the noise "Noise2" obtained by Noise Calibration in the initial random noise "Noise1" still satisfies the standard normal distribution. We believe that when the initial denoising step t_0 increases, although $||f_l^\nu(x^r) - f_l^\nu(\hat{x}_0^{t_0}))||$ generally becomes larger, $\frac{\sqrt{\bar{\alpha}_{t_0}}}{\sqrt{1-\bar{\alpha}_{t_0}}}$ becomes smaller. Moreover, during each iteration, the overall value of $\frac{\sqrt{\bar{\alpha}_{t_0}}}{\sqrt{1-\bar{\alpha}_{t_0}}}(f_l^\nu(x^r) - f_l^\nu(\hat{x}_0^{t_0}))$ will not be very large. Therefore, the noise "Noise2" after iterations is still not much different from "Noise1".

2.3 Demonstration of Enhancement Effects on Real Videos

Figs. 3 and 4 demonstrate that Noise Calibration can greatly maintain content consistency before and after enhancement for real videos. However, to ensure the effectiveness of the enhancement, it is necessary to employ alternative generative models that are better at understanding and simulating the physical world in motion, such as Sora [1].

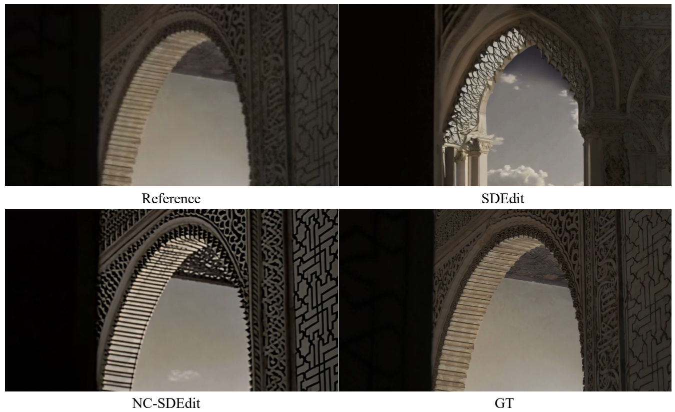

Fig. 3: Display of Real Video Enhancement on UDM10 [9] with VideoCrafter [2]

Fig. 4: Display of Real Video Enhancement on REDS4 [4] with MS-Vid2Vid-XL [10]

3 More Qualitative Results

We show more video enhancement results produced by our method based on VideoCrafter in Figs. 5 and 6. Furthermore, the enhanced effects of Noise Calibration on existing state-of-the-art (SOTA) refinements can be seen in Figs. 7 and 8.

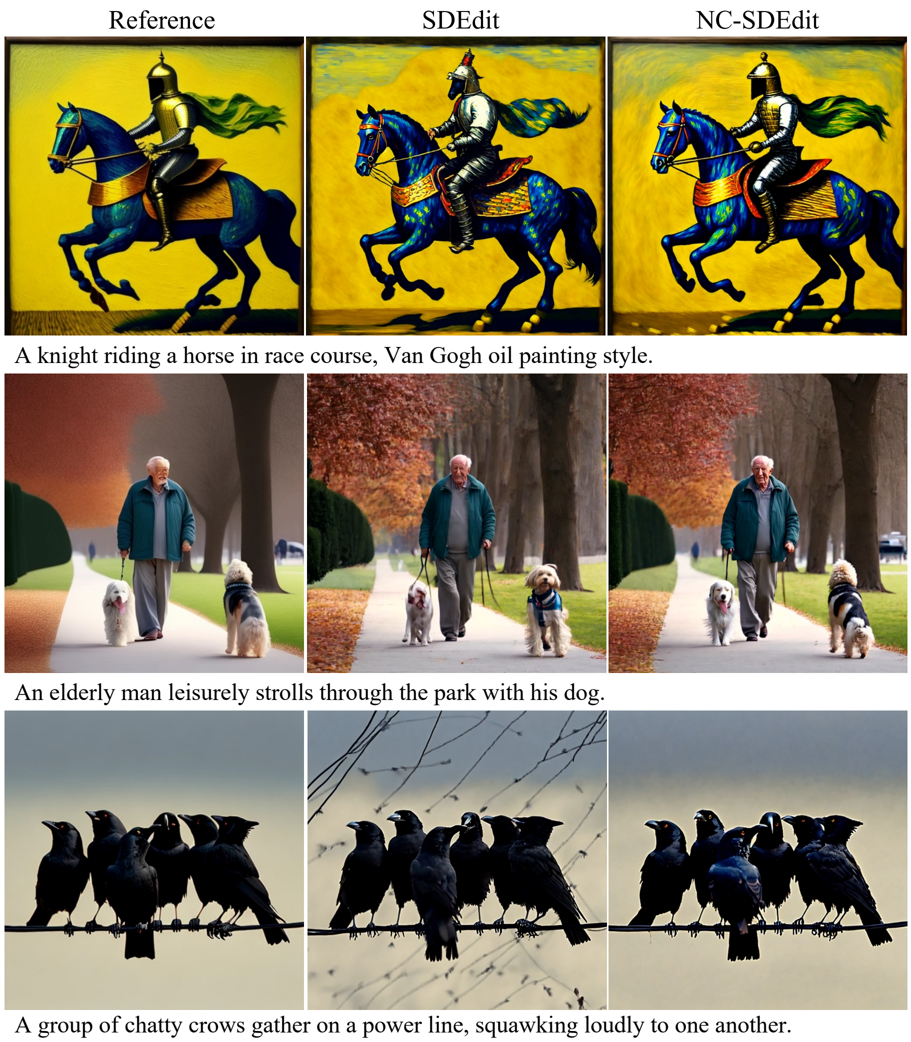

Fig. 5: Visual Comparisons of Video Enhancement based on VideoCrafter

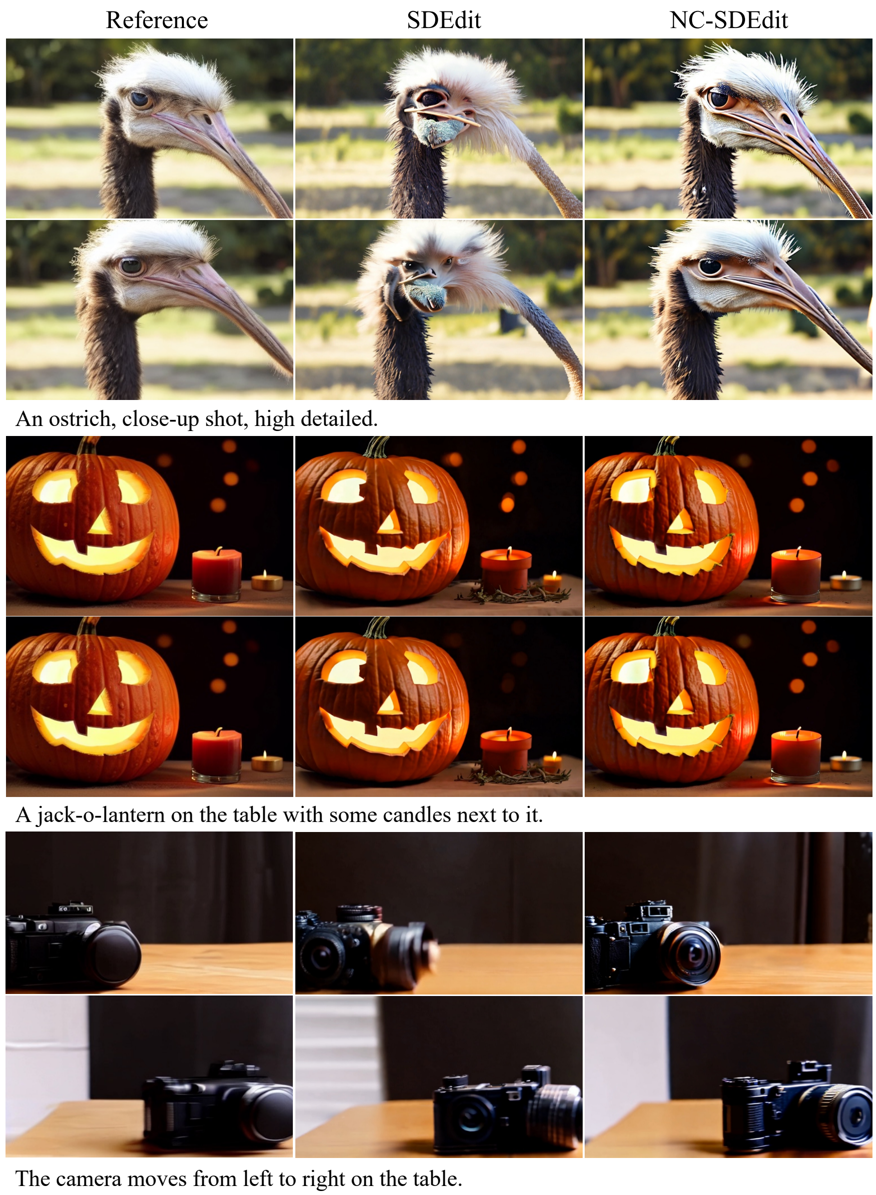

Fig. 6: Visual Comparisons of Video Enhancement based on VideoCrafter

Fig. 7: Visual Demonstration of MS-Vid2Vid-XL [10] with Noise Calibration

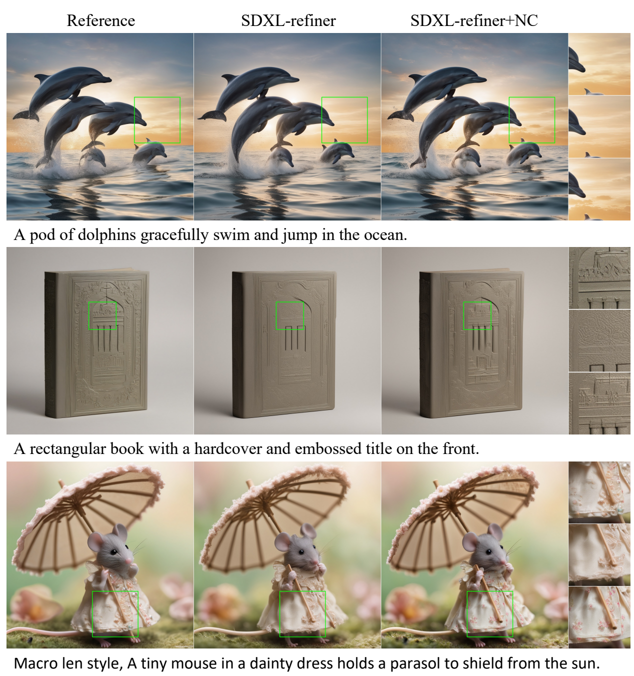

Fig. 8: Visual Demonstration of SDXL-1.0-refiner [5] with Noise Calibration

References

1. Brooks, T., Peebles, B., Holmes, C., DePue, W., Guo, Y., Jing, L., Schnurr, D., Taylor, J., Luhman, T., Luhman, E., Ng, C., Wang, R., Ramesh, A.: Video generation models as world simulators (2024), https://openai.com/research/video-generation-models-as-world-simulators 3
2. Chen, H., Xia, M., He, Y., Zhang, Y., Cun, X., Yang, S., Xing, J., Liu, Y., Chen, Q., Wang, X., et al.: Videocrafter1: Open diffusion models for high-quality video generation. arXiv preprint arXiv:2310.19512 (2023) 4
3. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in neural information processing systems **33**, 6840–6851 (2020) 1
4. Nah, S., Baik, S., Hong, S., Moon, G., Son, S., Timofte, R., Mu Lee, K.: Ntire 2019 challenge on video deblurring and super-resolution: Dataset and study. In: Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops. pp. 0–0 (2019) 4
5. Podell, D., English, Z., Lacey, K., Blattmann, A., Dockhorn, T., Müller, J., Penna, J., Rombach, R.: Sdxl: Improving latent diffusion models for high-resolution image synthesis. arXiv preprint arXiv:2307.01952 (2023) 1, 7
6. Si, C., Huang, Z., Jiang, Y., Liu, Z.: Freeu: Free lunch in diffusion u-net. arXiv preprint arXiv:2309.11497 (2023) 2
7. Song, Y., Ermon, S.: Generative modeling by estimating gradients of the data distribution. Advances in neural information processing systems **32** (2019) 1
8. Song, Y., Sohl-Dickstein, J., Kingma, D.P., Kumar, A., Ermon, S., Poole, B.: Score-based generative modeling through stochastic differential equations. arXiv preprint arXiv:2011.13456 (2020) 1
9. Yi, P., Wang, Z., Jiang, K., Jiang, J., Ma, J.: Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 3106–3115 (2019) 4
10. Zhang, S., Wang, J., Zhang, Y., Zhao, K., Yuan, H., Qin, Z., Wang, X., Zhao, D., Zhou, J.: I2vgen-xl: High-quality image-to-video synthesis via cascaded diffusion models. arXiv preprint arXiv:2311.04145 (2023) 4, 7