VideoGigaGAN: Towards Detail-rich Video Super-Resolution
Supplementary Material

This supplementary document includes additional quantitative results, our network architecture, and details on implementations and training.

We encourage readers to refer to our project website (https://videogigagan.github.io/) for more visual results.

A. Network architecture

A.1. GigaGAN upsampler

We show the configurationss of our GigaGAN upsampler in Table 1. For the low-pass filters, we use a kernel of $\frac{1}{16}[1, 4, 6, 4, 1]$ before the downsampling operations.

A.2. Flow-guided feature propagation module

We follow the architecture in BasicVSR++ [3]. We use SPyNet [22] as our flow estimator to reduce memory cost. For the feature extraction, we use 5 residual blocks. The number of residual blocks for propagation is set to 7. The kernel size of the deformable convolutional netowrk (DCN) is 3. We encourage readers to refer to BasicVSR++ [3] for more details.

B. Training and evaluation details

Datasets. Following previous works [2, 4], we use REDS [21] and Vimeo-90K [30] for training purpose. For REDS, we use clips 000, 011, 015, 020 of the training set for testing, and clips 000, 001, 006, 017 are used for validation, the rest of the clips are used for training. The ground truth has a resolution of 1280×720. For Vimeo-90K, in addition to its official test set Vimeo-90K-T, we use UDM10 [32] and Vid4 [17] for testing purpose. The ground truth has a resolution of 448×256.

Degradation. We use MMagic's [19] script for degradations - Bicubic (BI) and Blur Downsampling (BD). For BD, the ground truth is blurred by a Gaussian filter with $\sigma = 1.6$, followed by a $4\times$ subsampling.

Training settings. We use Adam optimizer [12] for training with a fixed learning rate of 5×10^{-5}. During training, we randomly crop a 64×64 patch from each LR input frames at the same location. We use 10 frames of each video and a batch size of 32 for training. The batch is distributed into 32 NVIDIA A100 GPUs. The total number of training iterations for each model is $100,000$.

Test settings. During the testing, we use the full-frame of the videos. Particularly, for Vimeo-90K-T, we follow its tradition and only evaluate PSNR, SSIM and LPIPS [33] on the center frame.

Metrics. We consider two aspects in our evaluation: per-frame quality and temporal consistency.

For **per-frame quality**, we use **PSNR, SSIM, and LPIPS** [33]. Except for REDS4, we evaluate PSNR and SSIM on y-channel following previous works [2, 3, 18].

For **temporal consistency**, we use warping error E_{warp} [13] and proposed referenced warping error $E^{\text{ref}}_{\text{warp}}$. Please refer to our main paper for the definition of $E^{\text{ref}}_{\text{warp}}$. We use RAFT [26] as our flow estimator when computing temporal consistency.

C. Additional quantitative results and discussion

Reliance on LPIPS metric. Recent works [5, 10, 23–25] highlight that PSNR/SSIM metrics often favor blurry results. Following generative VSR methods like StableVSR [24], we use LPIPS but acknowledge its limitations in capturing higher-level structures [7]. To address this, we also evaluate FID [8] and DISTS [6]. Our model still shows superior perceptual scores.

Table 1. GigaGAN model configurations

$\mathbf{z}$ dimension	512
$\mathbf{w}$ dimension	512
Mapping network layers	4
Activation	LeakyReLU
$\mathcal{G}$ channel base	32768
$\mathcal{G}$ channel max	512
$\mathcal{G}$ # of filters N for adaptive kernel selection	[1, 1, 1, 1, 1, 2, 4, 8, 16, 16, 16, 16]
$\mathcal{G}$ spatial self-attention resolutions	[8, 16]
$\mathcal{G}$ temporal attention resolutions	[8, 16, 32, 64]
$\mathcal{G}$ attention depth	[2, 2, 2, 1]
$\mathcal{G}$ temporal attention window size	1
$\mathcal{G}$ temporal convolution kernel size	3
$\mathcal{G}$ # synthesis block per resolution	[4, 4, 4, 4, 4, 4, 3]
$\mathcal{G}$ # downsampling blocks	3
$\mathcal{D}$ channel base	32768
$\mathcal{D}$ channel max	512
$\mathcal{D}$ attention depth	[2, 2, 1]
$\mathcal{D}$ attention resolutions	[8, 16]
$\mathcal{G}$ model size	369M
$\mathcal{D}$ model size	179M

Table 2. Additional results on REDS4 [21] dataset.

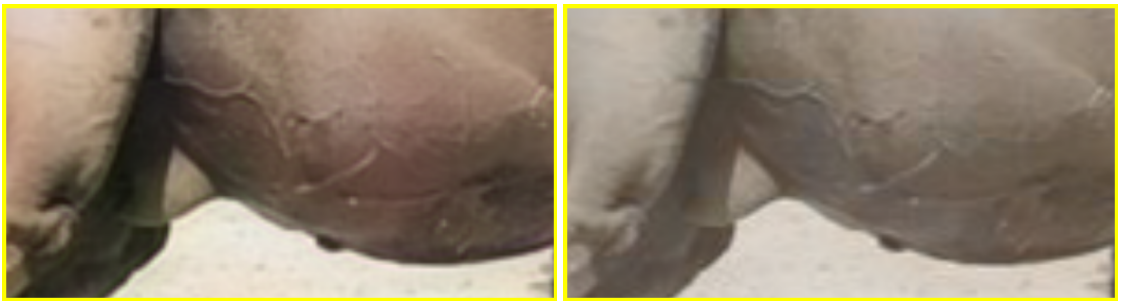
Model	LPIPS↓	FID↓	DISTS↓
EvTexture [9]	0.1684	101.9	0.065
StableVSR [24]	0.1934	96.2	0.045
RealESRGAN [28]	0.4509	98.2	1.750
OVSR [31]	0.1746	123.8	0.063
Ours	**0.1582**	**95.0**	**0.041**

Additional comparison. We additionally compare with a GAN-based model RealESRGAN [29] and OVSR [31] in Table 2. Our model performs better.

RWE metric. We propose the Referenced Warping Error (RWE) in Eqn. (4) in the main paper. We apply it to the *blind video colorization task* on the DAVIS dataset, comparing it with the Warping Error (WE) to demonstrate its robustness. We evaluate two methods, DVP [15] and All-in-One deflicker [14], and report results in Table 3 and Figure 1. Similar trends emerge: "all-black frames" score 0 WE but high RWE, while All-in-one deflicker achieves a WE lower than GT but favors blurrier results.

Table 3. Additional results for the RWE metric.

Method	RWE($\times 10^{-3}$) ↓	WE($\times 10^{-3}$) ↓
GT	0	3.416
All-black frames	5.416	**0**
DVP [15]	**3.377**	2.649
All-in-one deflicker [14]	4.589	1.595



DVP, sharper veins | All-in-one deflicker

Figure 1. Visual comparison between DVP [15] and All-in-One deflicker [14]. Although the All-in-One deflicker shows lower WE, its output is blurrier than DVP.

D. More visual results

Artifacts of adding LPIPS to training loss. We retrain BasicVSR++ [3] and RVRT [16] with additional perceptual loss and report in Table 4. Training RVRT with LPIPS is unstable and diverges. We observe that training BasicVSR++ with LPIPS produces severe checkerboard artifacts in all results (zoom in for details), as also observed in previous papers [20, 27]

Table 4. **LPIPS loss.** Adding LPIPS to the training loss improves performance on LPIPS, but it introduces lower PSNR/SSIM and makes the training unstable.

Model	LPIPS↓	PSNR↑
BasicVSR++ [3]	0.1786	32.39
RVRT [16]	0.1727	32.74
BasicVSR++ + LPIPS	0.1646	31.42
RVRT + LPIPS	diverged	diverged
VideoGigaGAN (ours)	0.1582	30.46

Figure 2. **Adding LPIPS uncarefully may introduce checkerboard artifacts.** We retrain BasicVSR++ [3] with an additional LPIPS loss. It brings LPIPS metric down, but also introduces visual artifacts such as checkerboard effects. **Zoom in for details**.

StyleGAN3. StyleGAN3 (SG3) [11] is famous for its alias-free attribute. However, directly incorporating SG3 blocks into GigaGAN degrades frame quality and introduces "swirly" artifacts [1] (Figure 3). Also, SG3 may not be tailored for scaling-up purposes, as it removes modules like noise inputs and residual connections, which limits its high-frequency detail delivery. Therefore, we end up with StyleGAN2's blocks as in the original GigaGAN [10]. More studies on anti-aliasing features in generative models are needed in the future.

Input with SG3 blocks GT

Figure 3. **StyleGAN3**. Directly introducing StyleGAN3 (SG3)'s blocks brings a quality drop.

Video results. We encourage readers to refer to our project website (https://videogigagan.github.io/) for more visual results.

E. Limitations

(a) Long video (b) Small objects

Figure 4. **Limitations.** Our approach has some limitations. (a) When the video is long, the feature propagation becomes inaccurate, which may introduce undesired artifacts like incorrect propagated patterns. (b) Our model cannot handle well **small objects**, *e.g.* small characters.

Our model encounters challenges when processing long videos (e.g., 200 frames or more). This difficulty arises from misguided feature propagation caused by inaccurate optical flow in such extended video sequences. Additionally, our model does not perform well in handling small objects, such as text and characters, as the information pertaining to these objects is significantly lost in the LR video input. Examples of these failure cases are illustrated in Figure 4.

References

[1] Brooks, T., Hellsten, J., Aittala, M., Wang, T.C., Aila, T., Lehtinen, J., Liu, M.Y., Efros, A., Karras, T.: Generating long videos of dynamic scenes. In: NeurIPS (2022) 3

[2] Chan, K.C., Wang, X., Yu, K., Dong, C., Loy, C.C.: Basicvsr: The search for essential components in video super-resolution and beyond. In: CVPR (2021) 1

[3] Chan, K.C., Zhou, S., Xu, X., Loy, C.C.: BasicVSR++: Improving video super-resolution with enhanced propagation and alignment. In: CVPR (2022) 1, 3

[4] Chan, K.C., Zhou, S., Xu, X., Loy, C.C.: Investigating tradeoffs in real-world video super-resolution. In: CVPR (2022) 1

[5] Chu, M., Xie, Y., Mayer, J., Leal-Taixé, L., Thuerey, N.: Learning temporal coherence via self-supervision for gan-based video generation. ACM Transactions on Graphics (TOG) **39**(4), 75–1 (2020) 1

[6] Ding, K., Ma, K., Wang, S., Simoncelli, E.P.: Image quality assessment: Unifying structure and texture similarity. IEEE transactions on pattern analysis and machine intelligence **44**(5), 2567–2581 (2020) 1

[7] Fu, S., Tamir, N., Sundaram, S., Chai, L., Zhang, R., Dekel, T., Isola, P.: Dreamsim: Learning new dimensions of human visual similarity using synthetic data. In: NeurIPS (2023) 1

[8] Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. In: NeurIPS (2017) 1

[9] Kai, D., Lu, J., Zhang, Y., Sun, X.: EvTexture: Event-driven Texture Enhancement for Video Super-Resolution. In: Proceedings of the 41st International Conference on Machine Learning. vol. 235, pp. 22817–22839. PMLR (2024) 2

[10] Kang, M., Zhu, J.Y., Zhang, R., Park, J., Shechtman, E., Paris, S., Park, T.: Scaling up gans for text-to-image synthesis. In: CVPR (2023) 1, 3

[11] Karras, T., Aittala, M., Laine, S., Härkönen, E., Hellsten, J., Lehtinen, J., Aila, T.: Alias-free generative adversarial networks. In: NeurIPS (2021) 3

[12] Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014) 1

[13] Lai, W.S., Huang, J.B., Wang, O., Shechtman, E., Yumer, E., Yang, M.H.: Learning blind video temporal consistency. In: ECCV (2018) 1

[14] Lei, C., Ren, X., Zhang, Z., Chen, Q.: Blind video deflickering by neural filtering with a flawed atlas. In: CVPR 2, 3

[15] Lei, C., Xing, Y., Chen, Q.: Blind video temporal consistency via deep video prior. In: NeurIPS (2020) 2, 3

[16] Liang, J., Fan, Y., Xiang, X., Ranjan, R., Ilg, E., Green, S., Cao, J., Zhang, K., Timofte, R., Gool, L.V.: Recurrent video restoration transformer with guided deformable attention. In: NeurIPS (2022) 3

[17] Liu, C., Sun, D.: On bayesian adaptive video super resolution. TPAMI **36**(2), 346–360 (2013) 1

[18] Liu, C., Yang, H., Fu, J., Qian, X.: Learning trajectory-aware transformer for video super-resolution. In: CVPR (2022) 1

[19] MMagic Contributors: MMagic: OpenMMLab multimodal advanced, generative, and intelligent creation toolbox. https://github.com/open-mmlab/mmagic (2023) 1

[20] Mustafa, A., Mikhailiuk, A., Iliescu, D.A., Babbar, V., Mantiuk, R.K.: Training a task-specific image reconstruction loss. In: Proceedings of the IEEE/CVF winter conference on applications of computer vision. pp. 2319–2328 (2022) 3

[21] 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: CVPRW (2019) 1, 2

[22] Ranjan, A., Black, M.J.: Optical flow estimation using a spatial pyramid network. In: CVPR (2017) 1

[23] Rombach, R., Blattmann, A., Lorenz, D., Esser, P., Ommer, B.: High-resolution image synthesis with latent diffusion models. In: CVPR (2022) 1

[24] Rota, C., Buzzelli, M., van de Weijer, J.: Enhancing perceptual quality in video super-resolution through temporally-consistent detail synthesis using diffusion models. In: ECCV (2024) 1, 2

[25] Saharia, C., Ho, J., Chan, W., Salimans, T., Fleet, D.J., Norouzi, M.: Image super-resolution via iterative refinement. TPAMI **45**(4), 4713–4726 (2022) 1

[26] Teed, Z., Deng, J.: Raft: Recurrent all-pairs field transforms for optical flow. In: ECCV (2020) 1

[27] Uelwer, T., Michels, F., De Candido, O.: Evaluating robust perceptual losses for image reconstruction. In: I Can't Believe It's Not Better Workshop: Understanding Deep Learning Through Empirical Falsification (2022) 3

[28] Wang, X., Xie, L., Dong, C., Shan, Y.: Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: ICCVW (2021) 2

[29] Wang, X., Xie, L., Dong, C., Shan, Y.: Real-esrgan: Training real-world blind super-resolution with pure synthetic data. In: ICCVW (2021) 2
[30] Xue, T., Chen, B., Wu, J., Wei, D., Freeman, W.T.: Video enhancement with task-oriented flow. IJCV **127**(8), 1106–1125 (2019) 1
[31] Yi, P., Wang, Z., Jiang, K., Jiang, J., Lu, T., Tian, X., Ma, J.: Omniscient video super-resolution. In: ICCV (2021) 2
[32] Yi, P., Wang, Z., Jiang, K., Jiang, J., Ma, J.: Progressive fusion video super-resolution network via exploiting non-local spatio-temporal correlations. In: ICCV (2019) 1
[33] Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: CVPR (2018) 1