InstantViR: Real-Time Video Inverse Problem Solver with Distilled Diffusion Prior

Supplementary Material

1. Implementation Details

We implement InstantViR based on the pre-trained Wan2.1-1.3B text-to-video model [24]. The student solver is initialized with the teacher's weights and fine-tuned using our proposed amortized distillation objective. Our framework supports both the original WanVAE [24] and the accelerated LeanVAE [4]. The training process utilizes the measurement consistency loss (likelihood) to enforce data fidelity and the distribution matching distillation (prior) loss [30] to inherit the generative prior. We utilize the Open-Sora dataset [12] solely as a source of raw videos to synthesize degraded measurements $\boldsymbol{y}$. The ground-truth clean videos $\boldsymbol{x}$ are never used in the loss computation; the student learns to reconstruct $\boldsymbol{x}$ solely through the guidance of the likelihood term and the teacher prior. Below, we detail the specific configurations for operators, baselines, and hyperparameters.

• Auxiliary Score Network s_φ: s_φ is a trainable copy of the *Wan2.1-T2V-1.3B* backbone (32 layers, 2048 dimensions, 16 heads). It is trained via a denoising score matching loss $\mathcal{L}_{\text{DSM}}(\varphi) = \mathbb{E}_{t,\boldsymbol{\epsilon}}\left[\|s_\varphi(\hat{\boldsymbol{z}}_t, t, \boldsymbol{c}) - \boldsymbol{\epsilon}\|_2^2\right]$ on the noised $\hat{\boldsymbol{z}}_t \sim p_t(\hat{\boldsymbol{z}}_t|\hat{\boldsymbol{z}}_0)$, where $\hat{\boldsymbol{z}}_0 = G_\theta(\boldsymbol{y}, \boldsymbol{c})$ is generated from *student* model G_θ with measurement $\boldsymbol{y}$ and condition $\boldsymbol{c}$.

• Bidirectional-to-Causal Initialization: The student is initialized with exact same DiT architecture and weights as the teacher. The causal behavior is implemented by using a block-wise causal attention mask during the forward pass, preserving pre-trained prior without structural modification.

• Latent Space Alignment: *Architecture match* in Algo.1 refers to the latent compatibility between WanVAE and LeanVAE (both 16-channel, $8\times$ downsampling). This alignment allows the Wan2.1-initialized student to process LeanVAE latents directly, paired with the official checkpoint-initialized LeanVAE decoder.

Forward Operators. We evaluate our method on three standard video inverse problems, where the degradation operators $\mathcal{A}(\cdot)$ are implemented as follows:

- **Random Inpainting:** We apply random binary masks with a masking ratio of 50%. The masks are applied in the latent space for training efficiency.
- **Gaussian Deblurring:** We apply a spatial Gaussian blur kernel with a size of 61×61 and a standard deviation $\sigma = 3.0$ to each video frame.
- **4× Super-Resolution:** We perform anti-aliased downsampling with a factor of 4. Specifically, the high-resolution video is downsampled in pixel space using a Resizer, and the resulting low-resolution video is then encoded back into the latent space to serve as the model input.

Baselines. For image-based diffusion baselines such as DPS [5] and SVI [10], due to memory constraints and their inherent image-processing nature, the visual results are generated by processing patches (256×256) and stitching them together. All other video-based methods [9] process frames or blocks directly.

Hyperparameters. To ensure stable training and efficient convergence, our configuration is based on the established settings of CausVid [31]. We utilize the AdamW optimizer with a fixed learning rate of 2×10^{-6} and apply gradient clipping with a threshold of 1.0 to mitigate potential instability during the distillation process. The distribution matching distillation [30] (prior) loss incorporates a warmup phase of 1,000 steps. Consistent with the teacher model's distillation requirements, we set the timestep shift parameter to 8.0, the real guidance scale to 3.5, and the fake generation update ratio to 5, while processing temporal data with a block size of 3 frames.

Regarding the objective function, the weight of the measurement consistency loss (likelihood) is tuned specifically for the difficulty of each degradation: it is set to 0.1 for random inpainting, 1.0 for Gaussian deblurring, and 0.3 for $4\times$ super-resolution. Uniquely for the inpainting task, we compute the prior loss exclusively on the masked regions to force the student to focus its generative capacity on hallucinating the missing content rather than reconstructing visible pixels. Finally, computational settings are adjusted based on the VAE employed; we use a batch size of 2 for the standard WanVAE [24], which we increase to 4 when utilizing the more memory-efficient LeanVAE [4]. For the latter, the latent shape is explicitly configured as $1 \times 21 \times 16 \times 60 \times 104$ to correspond with the high-resolution target outputs.

2. Algorithm

We provide detailed pseudocodes for the training pipeline (Algorithm 1) and the efficient block-wise streaming inference (Algorithm 2).

Training Protocol and VAE Alignment. Algorithm 1 outlines the core training loop. A crucial implementation detail involves the handling of the VAE when training the accelerated version, InstantViR†. For the standard version, the

Algorithm 1 InstantViR: Training Pipeline

Require: Dataset of raw videos (unpaired), frozen teacher diffusion model s_θ (Wan2.1), teacher VAE $\mathcal{E}/\mathcal{D}$, student VAE $\mathcal{E}'/\mathcal{D}'$ (Wan or Lean), degradation operator $\mathcal{A}$, student solver q_ϕ, auxiliary score network s_φ.

Initialize student parameters $\phi \leftarrow \theta$ (if architectures match) or custom init. Initialize φ.
Freeze teacher s_θ and VAEs.
while not converged **do**
  Sample batch $\boldsymbol{x}_{raw}$; Generate measurement $\boldsymbol{y} = \mathcal{A}(\boldsymbol{x}_{raw}) + \boldsymbol{n}$; Encode $\boldsymbol{z}_{\text{in}} = \mathcal{E}'(\boldsymbol{y})$.
  // 1. Student Prediction
  Predict clean latent: $\hat{\boldsymbol{z}}_0 = q_\phi(\boldsymbol{z}_{\text{in}})$. ▷ Latent in Student space
  // 2. Likelihood Term (Measurement Consistency)
  Decode to pixel: $\hat{\boldsymbol{x}}_0 = \mathcal{D}'(\hat{\boldsymbol{z}}_0)$.
  $\mathcal{L}_{\text{data}} = \|\boldsymbol{y} - \mathcal{A}(\hat{\boldsymbol{x}}_0)\|^2$.
  // 3. Prior Term (DMD with Latent Alignment)
  Sample $t, \boldsymbol{\epsilon}$.
  if Using LeanVAE ($\mathcal{E}' \neq \mathcal{E}$) **then**
    Bridge to Teacher Space: $\boldsymbol{z}_{target} = \mathcal{E}(\hat{\boldsymbol{x}}_0)$. ▷ Diff. through LeanDec $\rightarrow$ WanEnc
  else
    $\boldsymbol{z}_{target} = \hat{\boldsymbol{z}}_0$.
  end if
  Diffuse target: $\boldsymbol{z}_t = \alpha_t \boldsymbol{z}_{target} + \sigma_t \boldsymbol{\epsilon}$. ▷ Noise in Teacher space
  Teacher score: $\boldsymbol{g}_\theta = s_\theta(\boldsymbol{z}_t, t, \text{text})$.
  Student score (aux): $\boldsymbol{g}_\phi = s_\varphi(\boldsymbol{z}_t, t)$.
  $\mathcal{L}_{\text{prior}} = w(t)\|\boldsymbol{g}_\theta - \boldsymbol{g}_\phi\|^2$.
  // 4. Updates
  Update $\phi \leftarrow \phi - \eta_1 \nabla_\phi(\mathcal{L}_{\text{data}} + \lambda \mathcal{L}_{\text{prior}})$.
  Update φ to match score of noised $\boldsymbol{z}_{target}$ (via Denoising Score Matching).
end while

student q_ϕ and teacher s_θ share the same WanVAE [24] latent space. However, when training InstantViR† to utilize the ultra-efficient LeanVAE [4], a latent space mismatch arises: the student q_ϕ predicts Lean-latents to maximize inference speed, while the frozen teacher s_θ (Wan2.1) requires Wan-latents to compute the score distillation loss ($\mathcal{L}_{\text{prior}}$). Therefore, in the actual implementation of Algorithm 1, if the VAE is switched to LeanVAE, we perform an additional differentiable bridging step before teacher score computation. Specifically, the student's predicted latent $\hat{\boldsymbol{z}}_0$ is passed through the LeanDecoder to pixel space and immediately re-encoded via the frozen WanEncoder. This ensures the teacher provides valid guidance despite the student operating in a more efficient latent manifold.

Streaming Inference with KV Cache. Algorithm 2 details the deployment phase. Unlike standard diffusion sampling, which requires iterating over time t, our solver is a one-step feed-forward network ($t = 0$). To achieve high throughput for long videos, we process the video in non-overlapping temporal blocks ($n = 1 \dots N$). Crucially, to satisfy the causal dependency without redundant computation, we maintain a running Key-Value (KV) Cache ($\mathcal{K}, \mathcal{V}$). As shown in lines 10-16, for each new block $\boldsymbol{y}^{(n)}$, we only compute the Query $\mathbf{Q}^{(n)}$ for the current frames, while retrieving the Keys and Values from the history ($\mathbf{K}_{\text{past}}, \mathbf{V}_{\text{past}}$) to perform attention. This reduces the complexity of the attention mechanism from quadratic with respect to total video length to linear, enabling theoretically infinite streaming generation.

3. Additional Results

We present extensive qualitative comparisons on three challenging video inverse problems: video inpainting, Gaussian deblurring, and $4\times$ super-resolution. We compare InstantViR against strong diffusion-based baselines, including DPS [5], SVI [10], and Vision-XL [9]. Additionally, we showcase the text-guided controllability of our model in restoration tasks.

Video Inpainting Comparison. Figure 1 compares our method against DPS, SVI, and Vision-XL on a random inpainting task (50% mask). Sampling-based methods like DPS and SVI often struggle with consistency or produce blurry artifacts in large masked regions. Vision-XL produces higher quality but remains computationally expensive. InstantViR (Ours) and its accelerated variant (Ours†) achieve

Algorithm 2 InstantViR: Inference Pipeline

Require: Degraded video stream $\boldsymbol{y}$, trained solver q_ϕ, VAE decoder $\mathcal{D}$, block size L.

Initialize KV Cache: $\mathcal{K} \leftarrow \emptyset, \mathcal{V} \leftarrow \emptyset$.
Split $\boldsymbol{y}$ into temporal blocks $\{\boldsymbol{y}^{(1)}, \boldsymbol{y}^{(2)}, \ldots, \boldsymbol{y}^{(N)}\}$.
for $n = 1$ to N **do**
    Encode measurement: $\boldsymbol{z}_{\text{in}}^{(n)} = \mathcal{E}(\boldsymbol{y}^{(n)})$ (or resize/embed).
    Set time step $t = 0$ (deterministic inference).
    // Causal Attention Block
    $\mathbf{Q}^{(n)} = \text{Proj}_Q(\boldsymbol{z}_{\text{in}}^{(n)})$
    $\mathbf{K}^{(n)} = \text{Proj}_K(\boldsymbol{z}_{\text{in}}^{(n)}), \mathbf{V}^{(n)} = \text{Proj}_V(\boldsymbol{z}_{\text{in}}^{(n)})$
    Inter-block Attention (Read Cache):
    Retrieve $\mathbf{K}_{\text{past}}, \mathbf{V}_{\text{past}}$ from $\mathcal{K}, \mathcal{V}$.
    $\mathbf{K}_{\text{full}} = [\mathbf{K}_{\text{past}}, \mathbf{K}^{(n)}], \mathbf{V}_{\text{full}} = [\mathbf{V}_{\text{past}}, \mathbf{V}^{(n)}]$.
    Compute Reconstruction:
    $\hat{\boldsymbol{z}}_0^{(n)} = \text{Attention}(\mathbf{Q}^{(n)}, \mathbf{K}_{\text{full}}, \mathbf{V}_{\text{full}})$.
    Pass through MLP and remaining layers of q_ϕ.
    Update Cache:
    Update $\mathcal{K} \leftarrow [\mathcal{K}, \mathbf{K}^{(n)}], \mathcal{V} \leftarrow [\mathcal{V}, \mathbf{V}^{(n)}]$.
    // Stream Output
    Decode block: $\hat{\boldsymbol{x}}_0^{(n)} = \mathcal{D}(\hat{\boldsymbol{z}}_0^{(n)})$.
    Yield $\hat{\boldsymbol{x}}_0^{(n)}$.
end for

superior visual fidelity and temporal coherence in a single step, effectively hallucinating missing content that is consistent with the ground truth.

Video Deblurring Comparison. Figure 2 presents comparisons on Gaussian deblurring. Our method effectively restores sharp details and outperforms optimization-based baselines, which often suffer from over-smoothing or severe temporal flickering. Notably, both our standard WanVAE-based model and the accelerated LeanVAE variant (Ours†) maintain high reconstruction quality, generating sharp and stable results.

Video Super-Resolution Comparison. Figure 3 showcases results for $4\times$ video super-resolution. While baselines like SVI and Vision-XL provide reasonable stability, they often lack high-frequency texture details. InstantViR successfully recovers fine structures and textures, producing a high-resolution output that closely matches the ground truth, demonstrating the power of our distilled video prior.

Tolerance to Mild Parameter Mismatches. To evaluate robustness against real-world degradation deviations, we simulate inference with mismatched Gaussian blur kernels. Applying a ground-truth kernel (e.g., size 61×61, $\sigma = 3.0$) to inputs, we provide slightly perturbed kernels to our solver during reconstruction. When the assumed size varies by ± 2 pixels (e.g., 63×63 or 59×59), performance degradation from PSNR 31.16 to 31.12 or 31.08 is marginal, which demonstrates stable reconstruction quality and tolerance to mild parameter mismatches in practical applications.

Text-Guided Deblurring. Figure 4 demonstrates the controllability of InstantViR. Given the same blurred measurement, our model can restore details according to different text prompts (e.g., adding "glasses" or changing eye color to "green eyes"), generating semantically consistent and realistic content that aligns with the user's intent.

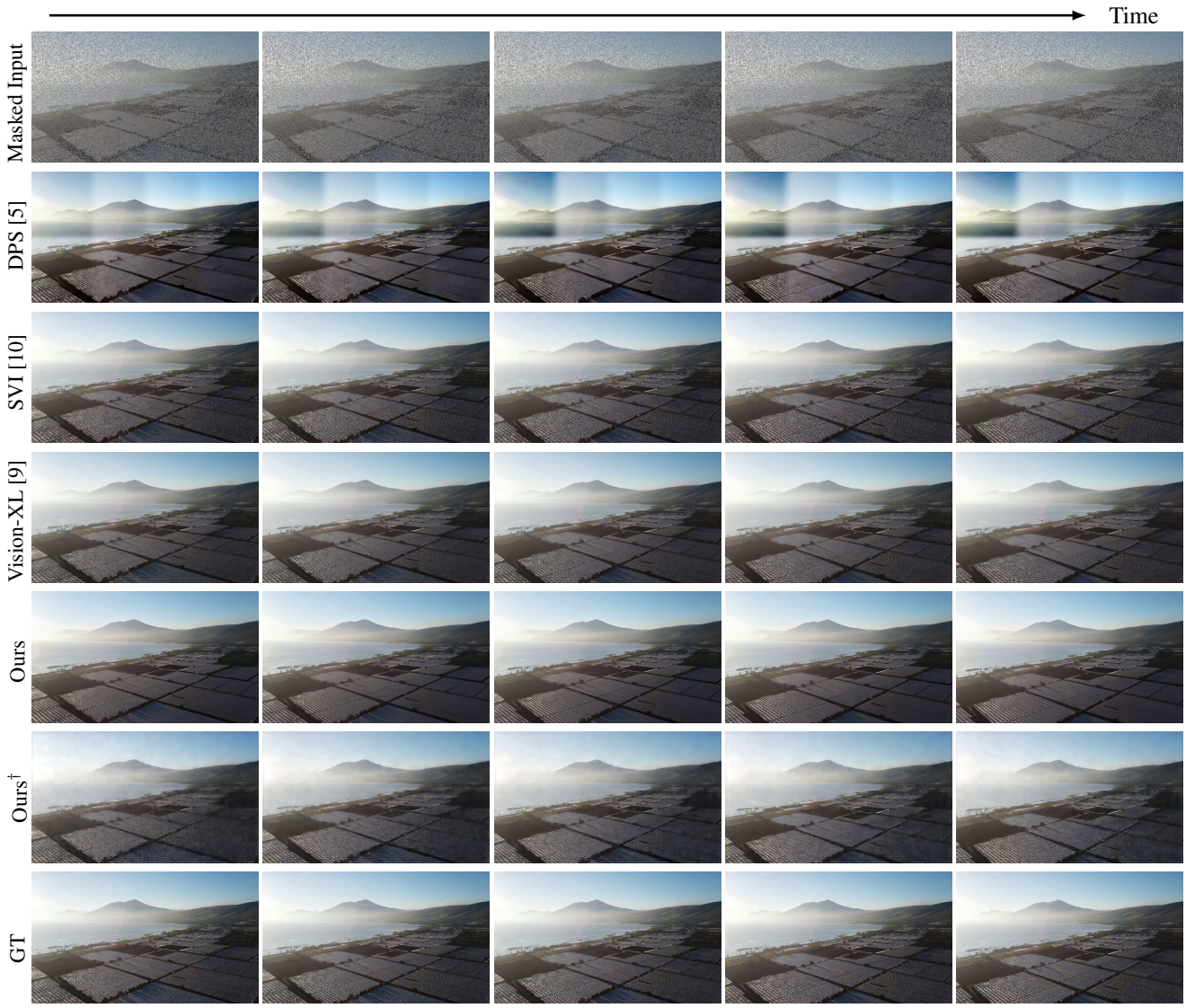

Figure 1. **Video Inpainting qualitative comparison.** Each row shows a complete sequence reconstructed by a specific method. InstantViR (both WanVAE **Ours** and LeanVAE **Ours**† variants) produces coherent content for every frame while requiring only a single feed-forward pass.

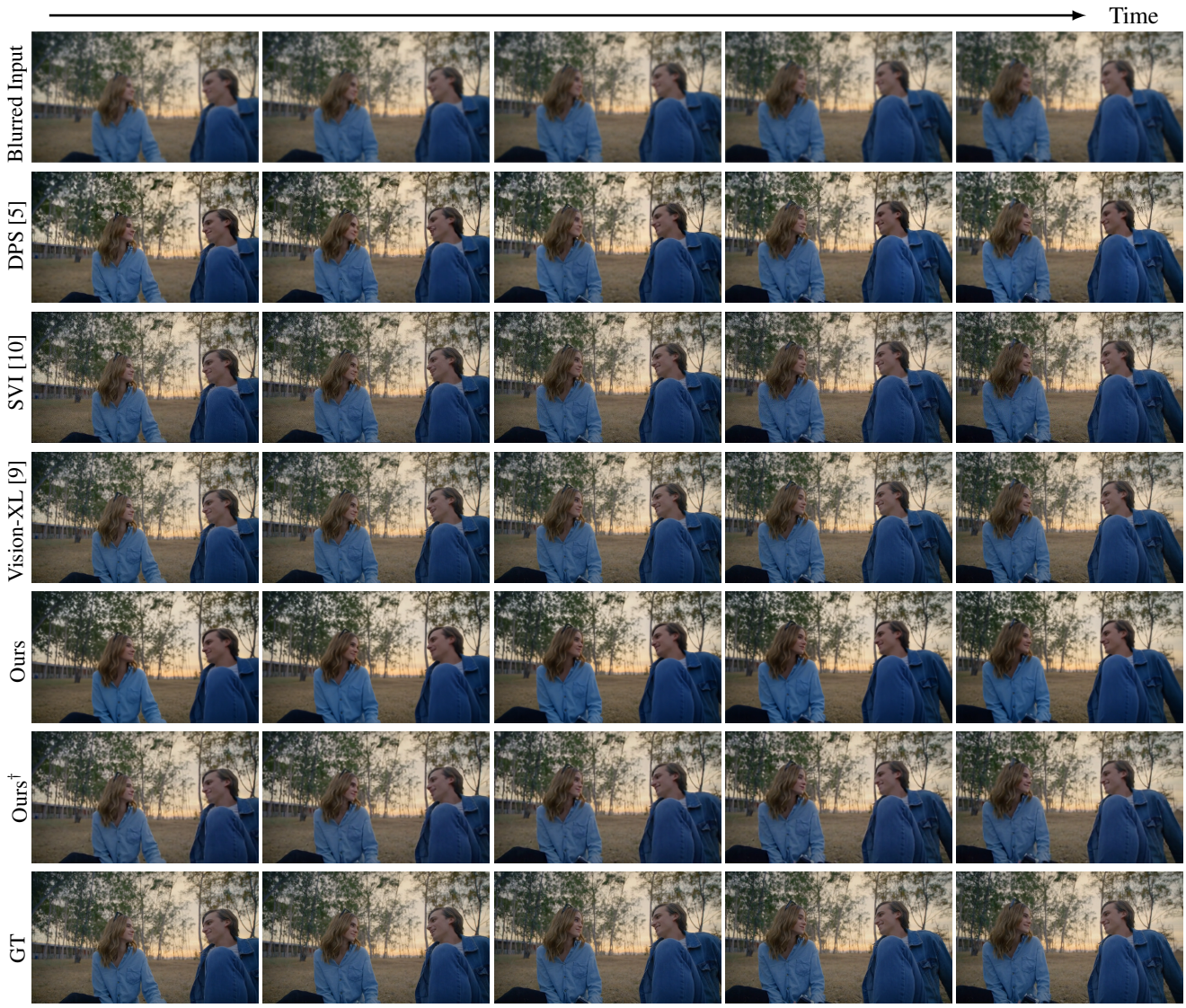

Figure 2. **Video Deblurring qualitative comparison.** Rows correspond to different methods; columns show consecutive frames covering the entire clip. InstantViR (both WanVAE **Ours** and LeanVAE **Ours**† variants) restores fine structures consistently across time, outperforming slower diffusion-based baselines.

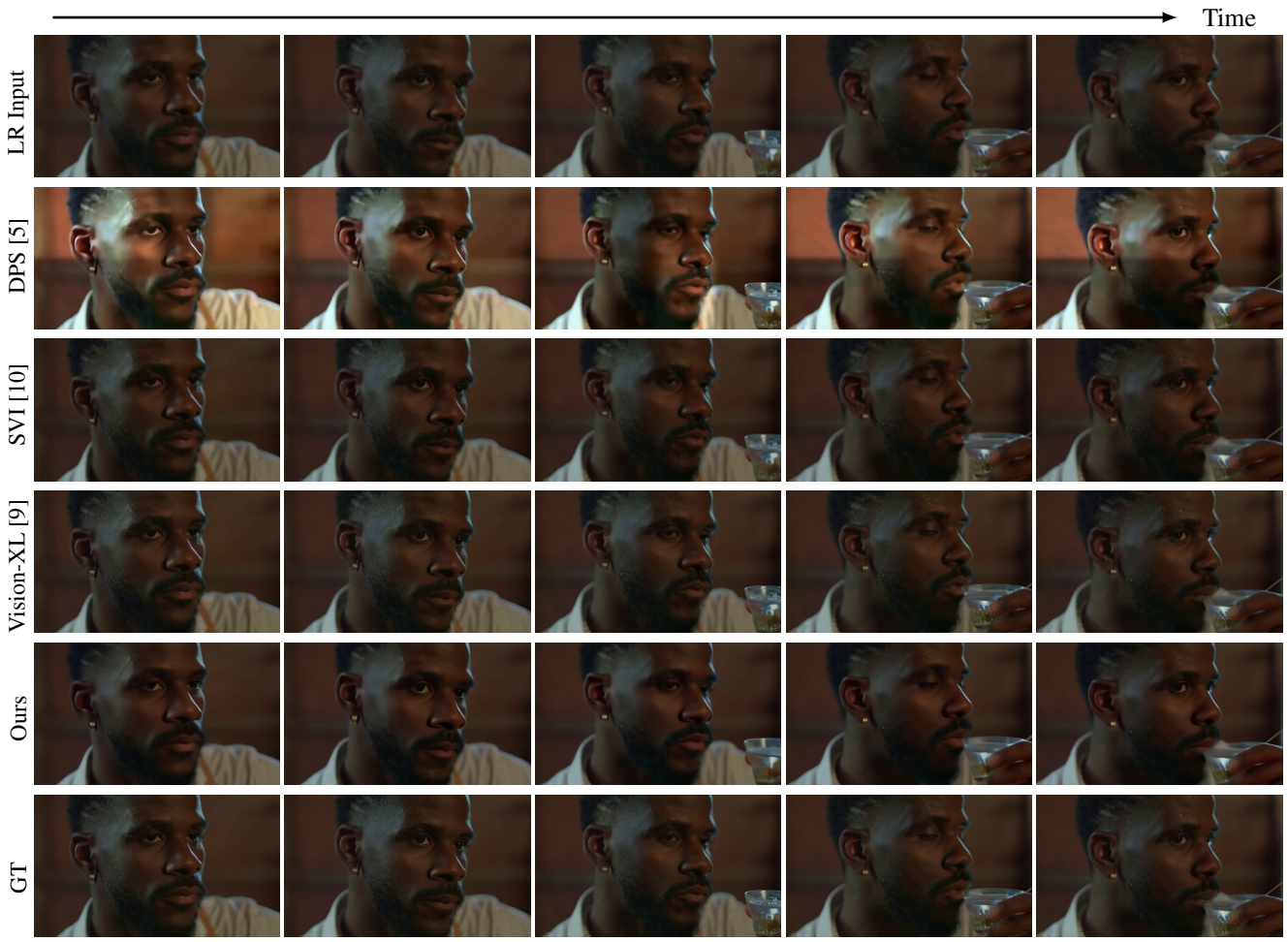

Figure 3. **Video Super-Resolution (4×) qualitative comparison.** InstantViR restores temporally consistent structures, outperforming slower diffusion-based baselines in both sharpness and coherence.

Figure 4. **Text-Guided Video Deblurring.** InstantViR can generate diverse outcomes from the same blurred input based on text prompts, adding specific features like glasses or changing eye color while maintaining temporal coherence.