SPDMark: Selective Parameter Displacement for Robust Video Watermarking

Supplementary Material

A. Datasets and training

We train on 10,000 videos from OpenVid-1M [24] using AdamW (β_1=0.9, β_2=0.999, weight decay 10^{-2}), learning rate 10^{-4}, batch size 1 on 1 NVIDIA A6000 GPU, for 6000 steps. For the first 2k steps, we optimize only the message recovery loss $\mathcal{L}_{rec}$; from 2k steps onward, we optimize $L_{\text{rec}} + L_{\text{imp}}$. Training uses 8-frame clips at 256 resolution. At test time, we evaluate full-length generations (25 frames for SVD-XT and 16 frames for MS).

B. Computational cost

Training takes $\approx$ 8 GPU hours. Only one basis per block is active at inference, so the decoding cost matches a single rank-r low-rank update per targeted block. The added parameters are $\approx$2M. Empirically, this adds $<5\%$ to the decoding time versus the frozen decoder.

C. Generation Quality Evaluation Metrics

Subject Consistency (SC). For a video with frames $1, \ldots, T$, let d_t be the L2-normalized DINO [5] image feature of frame t. SC averages the cosine similarity of each frame to (i) the first frame and (ii) its previous frame:

$$S_{\text{SC}} = \frac{1}{T-1} \sum_{t=2}^{T} \frac{1}{2} \big(\langle d_1, d_t \rangle + \langle d_{t-1}, d_t \rangle \big).$$

Background Consistency (BC). Let c_t be the L2-normalized CLIP image feature of frame t. BC mirrors SC but uses CLIP features [27]:

$$S_{\text{BC}} = \frac{1}{T-1} \sum_{t=2}^{T} \frac{1}{2} \big(\langle c_1, c_t \rangle + \langle c_{t-1}, c_t \rangle \big).$$

Motion Smoothness (MS). Drop the odd frames to form a lower-FPS sequence and synthesize them with a video frame-interpolation model (AMT) [23]. For each removed frame f_{2t-1} with interpolation $\hat{f}_{2t-1}$, compute the mean absolute error (MAE). The raw error is then normalized to $[0, 1]$ (same normalization as the flicker metric):

$$E = \frac{1}{T/2} \sum_{t=1}^{T/2} \text{MAE}\Big(\hat{f}_{2t-1}, f_{2t-1} \Big), \qquad S_{\text{MS}} = \frac{255 - E}{255}.$$

Imaging Quality (IQ). Per frame, run the MUSIQ [20] image-quality predictor (0–100), then average over frames and linearly rescale:

$$S_{\text{IQ}} = \frac{1}{T} \sum_{t=1}^{T} \frac{\text{MUSIQ}(t)}{100}.$$

D. Robustness

We evaluate SPDMark across photometric, temporal, and post-processing attacks.

D.1. Attack Protocol

Photometric and Spatial Attacks. We simulate common visual distortions encountered during content sharing: **Gaussian Noise**: additive noise $\mathcal{N}(0, \sigma^2)$ with $\sigma = 0.05$. **Gaussian Blur**: Gaussian blur with an 11×11 kernel and $\sigma = 2.0$. **Rotation**: rotation by $15°$, followed by resizing/cropping back to the original dimensions. **Center Crop**: retain the central 90% of the frame in both height and width, then resize it back to the original resolution. **Rescale**: downsample by $0.5\times$ using bicubic interpolation, then upsample back. **Color Jitter**: random brightness and contrast adjustments within $\pm 10\%$. **Subtitle**: add a semi-transparent caption box with overlaid text near the bottom of each frame, simulating subtitle or caption overlays.

Post-Processing and Screen Recording Simulation. We include transformations approximating recompression pipelines and phone screen capture: **Multi-Stage Recompression**: two-pass encoding: (1) H.264 at CRF=28, then (2) decode and re-encode with H.265 at a target bitrate of 600 kbps. **Screen Recording**: approximate screen capture by downscaling to 70% resolution and upscaling back, adding Gaussian noise ($\sigma = 0.03$), applying mild vignetting, and finally recompressing at 600 kbps. **Denoising**: apply mild Gaussian smoothing followed by small affine jitter, approximating denoising and stabilization-style post-processing. **STTN Inpainting**: We apply pretrained STTN-based video inpainting to a masked rectangular region in each video. Frames are resized to 432×240, the masked region is regenerated from neighboring and reference frames, and the inpainted result is composited back and resized to the original resolution.

Temporal Attacks. To evaluate temporal integrity and forensic capabilities: **Frame Drop**: randomly delete 50% of frames uniformly. **Frame Swap (Random)**: apply random permutation through pairwise swaps. **Frame Swap (Adjacent)**: swap selected adjacent frame pairs. **Frame Insert**: insert a single frame at a random position, either by duplicating a neighboring frame or by inserting a random noise frame. **Video Trim**: remove two frames from the beginning and two frames from the end of the video.

Table 5. Robustness under frame regeneration attacks. Values report bit accuracy when 30%, 50%, 70%, or 100% of frames are regenerated. Watermark detection remained successful in all settings.

Attack setting	30%	50%	70%	100%
Diffusion regeneration (step=60)	0.961	0.935	0.866	0.802
Diffusion regeneration (step=30)	0.961	0.925	0.858	0.807
Compression [1] ($q = 4$)	0.976	0.953	0.891	0.797
Compression [1] ($q = 8$)	0.937	0.892	0.851	0.874
Compression [6] ($q = 3$)	0.977	0.950	0.895	0.788
Compression [6] ($q = 6$)	0.961	0.915	0.855	0.829

D.2. Frame Regeneration Attacks

To further evaluate robustness against regeneration attacks, we consider frame-level regeneration scenarios in which a fraction of frames in the video is regenerated using either diffusion-based editing [37] or VAE-based compression pipelines [1, 6]. Table 5 reports bit accuracy when 30%, 50%, 70%, or 100% of the frames are regenerated. Across all settings, watermark detection remains successful, while bit accuracy degrades gradually as a larger fraction of frames is regenerated. This behavior is expected because regeneration changes the visual evidence available to the frame-wise extractor, but the video-level verification procedure can still accumulate enough valid frame matches to detect the watermark.

D.3. Model-Level Attacks

We additionally evaluate `SPDMark` under model-level changes. First, we plug the watermark encoder and extractor trained on the base SVD model directly into the SVD-XT pipeline, which corresponds to a denoiser-level change since SVD-XT is a fine-tuned variant of SVD. In this setting, the watermark remains recoverable with Bit Acc. $= 0.987$ and Robust Acc. $= 0.909$, while preserving generation quality (SC/BC/MS/IQ $= 0.964/0.958/0.963/0.680$). To further test robustness across denoiser architectures, we apply the same trained watermark encoder and extractor to Latte, which uses a Transformer-based denoiser. Even without retraining, the watermark remains recoverable with Bit Acc. $= 0.9790$ and Robust Acc. $= 0.9079$, with SC/BC/MS/IQ $= 0.984/0.979/0.973/0.619$. We also evaluate post-training quantization of the VAE decoder by simulating INT8 per-channel weight quantization using round-and-dequantize. Under this quantization, `SPDMark` remains recoverable with Bit Acc. $= 0.9797$ and Robust Acc. $= 0.9035$, indicating resilience to moderate deployment-time quantization.

In contrast, when the adversary fine-tunes the VAE decoder while keeping the learned basis shifts frozen, watermark extraction fails (Bit Acc. ≈ 0.65). This behavior is expected because `SPDMark` relies on the decoder weights remaining consistent with the learned basis-shift dictionary. Overall, these results suggest that `SPDMark` is robust to denoiser-side changes and post-training quantization in provider-controlled deployments, but not to adversarial retraining of the decoder itself.

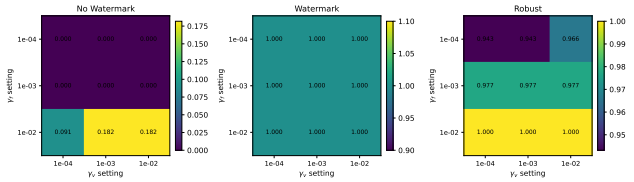

Figure 4. Detection behavior of SPDMark under different threshold settings (γ_f, γ_v). No Watermark reports the false positive rate on non-watermarked videos, Watermark reports the true positive rate on clean watermarked videos, and Robust reports the average true positive rate on attacked watermarked videos.

D.4. Sensitivity to Detection Thresholds

SPDMark uses frame-level and video-level verification thresholds, denoted by (γ_f, γ_v), during alignment-based watermark detection. To assess sensitivity to these parameters, we vary both thresholds over a small grid of operating points and report three quantities: *No Watermark*, which measures the false positive rate on non-watermarked videos; *Watermark*, which measures the true positive rate on clean watermarked videos; and *Robust*, which measures the average true positive rate on attacked watermarked videos.

Figure 4 shows that SPDMark is stable across a broad range of threshold settings. In particular, the *Watermark* true positive rate remains at 100% for all tested values of (γ_f, γ_v). The *Robust* true positive rate also remains high, ranging from 94.3% to 100.0%. As expected, looser thresholds slightly increase the *No Watermark* false positive rate; however, it remains at 0% for the stricter and default settings.

E. Additional Qualitative Examples

Figures 5 (SVD) and 6 (ModelScope) provide extended visualisations. For each base model, we show three videos and, for each video, display seven consecutive frames from the non-watermarked video followed by the corresponding `SPDMark` sample of the same video. `SPDMark` closely tracks the clean videos: textures, edges, and colors remain visually consistent, and we do not observe watermark-induced artifacts. The sequences further indicate that temporal coherence is preserved.

E.1. Sampling Configuration on ModelScope

Setup. We ablate three factors for ModelScope(Table 6): number of generated frames, diffusion steps, and CFG. For each configuration, we use the same prompt set and seed

Figure 5. SVD videos: Seven frames per row. For each video, the No watermark row is followed by the corresponding SPDMark row (for the same video).

protocol as in the main results, and report Bit accuracy, Robust accuracy, and Video quality metrics. The results mirror those of SVD-XT: longer videos improve robust accuracy, while diffusion steps and guidance have a limited effect on extraction or video quality.

Table 6. Sampling and ablation studies on **ModelScope**.

Factor	Setting	Bit Acc ↑	Robust Acc ↑	SC ↑	BC ↑	MS ↑	IQ ↑
Frames	8	0.880	0.863	0.964	0.974	0.974	0.480
	25	0.942	0.916	0.883	0.936	0.948	0.604
Steps	10	0.969	0.918	0.894	0.941	0.969	0.583
	25	0.977	0.925	0.934	0.961	0.972	0.624
Guidance	6	0.977	0.925	0.933	0.963	0.972	0.625
	12	0.977	0.929	0.953	0.967	0.971	0.629

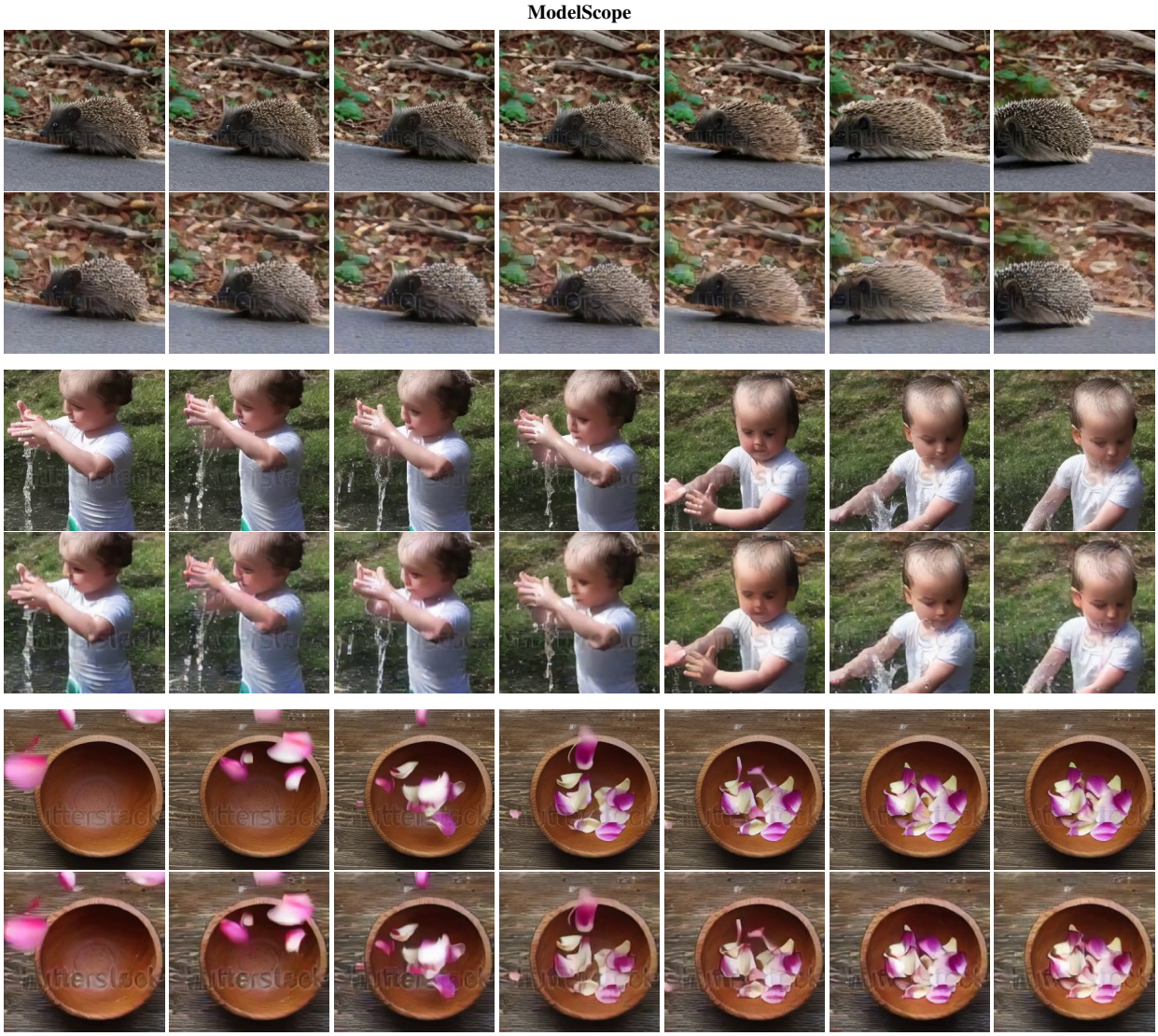

Figure 6. ModelScope videos: Seven frames per row. For each video, the No watermark row is followed by the corresponding SPDMark row (for the same video).