Improving Motion in Image-to-Video Models via Adaptive Low-Pass Guidance

Supplementary Material

A. Details about ALG

A.1. Implementation details of ALG

General algorithm for ALG. Algorithm 1 shows the general algorithm for ALG applying to all models. Note that while $\kappa(t)$ is written in the most general form possible (*i.e.*, any $\kappa : [0, 1] \to \mathbb{R}$), we use step function in our experiments (Sec. 3.2). When $\kappa(t) = 0$, no filter is applied (*i.e.*, $\mathbf{x}_{\text{init}}^{(t)} = \mathbf{x}_{\text{init}}$) and the sampling becomes equivalent to classifier-free guidance (CFG).

Algorithm 1 Image-to-video sampling with Adaptive Low-Pass Guidance (ALG)

Require: Denoiser $\mathbf{v}_\theta$, encoder E, decoder G, input conditioning image $\mathbf{w}_{\text{init}}$, prompt $\mathbf{c}$, guidance w, low-pass filter $\mathcal{F}_{\text{LP}}$, strength schedule $\kappa : [0, 1] \to \mathbb{R}$, total inference steps N

$\mathbf{x}_{\text{init}} \leftarrow E(\mathbf{w}_{\text{init}})$ ▷ Encode the input conditioning image
$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$
for $i = 1$ to N **do**
  $t \leftarrow \frac{i}{N}$
  $\mathbf{x}_{\text{init}}^{(t)} \leftarrow \mathcal{F}_{\text{LP}}\big(\mathbf{x}_{\text{init}},\, \kappa(t)\big)$
  $\mathbf{v}_{\text{ALG}} \leftarrow \mathbf{v}_\theta(\mathbf{x}, \mathbf{x}_{\text{init}}, t, \varnothing) + w \left[\mathbf{v}_\theta(\mathbf{x}, \mathbf{x}_{\text{init}}^{(t)}, t, \mathbf{c}) - \mathbf{v}_\theta(\mathbf{x}, \mathbf{x}_{\text{init}}^{(t)}, t, \varnothing)\right]$
  $\mathbf{x} \leftarrow \text{SolverStep}(\mathbf{x}, \mathbf{v}_{\text{ALG}}, t)$
end for
return $G(\mathbf{x})$ ▷ Decode the final latent into video

While the implementation of ALG is straightforward, subtle differences per model (Wan 2.1/2.2 [55], LTX-Video [16]) exist, depending on specific model architecture and implementation. In most cases, $\mathbf{x}$ is a 5-dimensional tensor (batch size, frame count, channel count, width, height). $\mathbf{x}_{\text{init}}$ is provided as input by either concatenating to $\mathbf{x}$ along channel dimension (Wan 2.1/2.2) or replacing the first input token with $\mathbf{x}_{\text{init}}$ (LTX-Video). Note that as $\mathbf{x}_{\text{init}}$ has a different shape (*i.e.*, frame count is 1) than $\mathbf{x}$, we expand its shape via zero padding. Implementation details of ALG for each model is explained below.

Implementation of ALG for Wan 2.1. Wan 2.1 [55] is a flow-matching model based on a diffusion transformer (DiT; [41]) backbone, fine-tuned from its pre-trained base text-to-video model checkpoint to perform an image-to-video generation task. The conditioning image is first encoded into the VAE latent space using the VAE encoder, and is incorporated into the input via zero-padding followed by channel-wise concatenation. We implement ALG by simply applying low-pass filter to the input conditioning image latent before performing zero-padding. Wan 2.1 has an additional input to the DiT—the CLIP embedding of the conditioning image. As the purpose of the CLIP embedding is to provide a high-level semantic information (and not fine-grained details of the image such as small edges), we do not apply low-pass filter to the input for the CLIP encoder and just use the original image. We implement ALG under the official diffusers codebase of Wan 2.1.

Implementation of ALG for Wan 2.2. Wan 2.2 [55] is likewise a DiT [41]-based flow-matching image-to-video model, but with a two-stage denoiser (one for high-noise steps and the other for low-noise steps) and an explicit first-frame masking mechanism. Similarly to Wan 2.1, the input image is first encoded into the VAE latent space, temporally zero-padded to match the target number of frames, and concatenated channel-wise. ALG is implemented similarly to Wan 2.1 as well, and is applied to the VAE-encoded input image before concatenation. One difference is that the CLIP embedding is disabled for the 14B sized Wan 2.2 model. We implement ALG on top of the official diffusers implementation of Wan 2.2.

Implementation of ALG for LTX-Video. LTX-Video [16] is a DiT [41]-based flow-matching model. Distinctively from Wan 2.1 and Wan 2.2, LTX-Video incorporates the input conditioning image by substituting the first frame of the noisy video latent with the clean conditioning image latent at each denoising step. At each denoising step, a scheduled noise is added to the conditioning image latent (which is used as the first frame). To integrate ALG into LTX-Video, we apply ALG by using low-pass filtered conditioning image latent as the first frame during the early steps (*i.e.*, before $t \in [0, t_{\text{trans}})$) and switching to the original conditioning image with a scheduled noise added to it thereafter (*i.e.*, $t \in [t_{\text{trans}}, 1)$).

Model	t_{trans}	Runtime (sec.)		Dynamic Degree	
		Default	ALG	Default	ALG
Wan 2.2	0.10	475	494	31.7	39.0
Wan 2.1	0.20	476	527	28.9	39.4
LTX-Video	0.10	58	59	15.5	21.5

Table 5. Comparison of CFG and ALG on inference time and dynamism, measured using a single NVIDIA H200 GPU for video generation.

Model	Type	Source
Wan 2.2	T2V	https://huggingface.co/Wan-AI/Wan2.2-T2V-A14B-Diffusers
	I2V	https://huggingface.co/Wan-AI/Wan2.2-I2V-A14B-Diffusers
Wan 2.1	T2V	https://huggingface.co/Wan-AI/Wan2.1-T2V-14B
	I2V	https://huggingface.co/Wan-AI/Wan2.1-I2V-14B-480P-Diffusers
LTX-Video	I2V	https://huggingface.co/Lightricks/LTX-Video

Table 6. Models used in our experiments.

Low-pass filter implementation. In our ALG experiments, we use downsampling followed by upsampling as our choice of low-pass filter. Specifically, we first bilinearly downsample the original latent into a smaller latent size (so that the latent width becomes latent_width$/\kappa(t)$), then upsample it back to the original latent size. While there are various possible choices of interpolation functions other than bilinear interpolation, we use it in our main experiments due to its simplicity.

Additional techniques. As discussed in Sec. 3.2, we apply two additional techniques that we find to improve video quality. First, after generation ends, the first frame of the final latent is overriden with the clean input latent. This is similar to `expand_timesteps` feature enabled by default for Wan 2.2 5B model (not in 14B, which we use), which overrides the first frame of the noisy latent with the clean latent during denoising. Our technique is distinct in that we override after denoising is entirely over, instead of during denoising. We empirically find this technique to slightly improve video quality, while using `expand_timesteps` feature introduces noticeable artifacts. Additionally, we find that denoising using the clean input latent at the beginning of the denoising process for 1 or 2 steps to help improve quality. Note that this does not affect the duration of denoising steps using the low-pass filtered latent. Instead, it merely *delays* the exposure of the model to the low-pass filtered latent slightly. These two techniques do not incur any additional computational overhead. The configuration for these techniques are described in Tab. 7 (*i.e.*, first-frame override, low-pass filter delaying).

Computational cost. Note that ALG introduces additional inference cost compared to original CFG. Specifically, CFG (Eq. (3)) requires a forward pass of two conditions $(\mathbf{x}_{\text{init}}, \mathbf{c})$ and $(\mathbf{x}_{\text{init}}, \varnothing)$, while ALG requires additional computation of $(\mathbf{x}_{\text{init}}, \varnothing)$ for the first term of Eq. (3). Thus, ALG introduces a tradeoff between inference cost and dynamic degree, which can be controlled by setting hyperparameter t_{trans}. However, this overhead is marginal, as $\kappa(t) \neq 0$ for only few t values (*i.e.*, we only apply low-pass filter in the early steps; see Tab. 7). We present the running time in seconds to generate one video per model, for the default method (CFG) and our method (ALG) in Tab. 5. As shown, the additional cost introduced by ALG is at most around 11% (for Wan 2.1), while Dynamic Degree increases by on average 33% (36.3% for Wan 2.1).

B. Experimental setup details

In this section, we provide additional details about our experiments, including additional experimental results (both qualitative and quantitative), inference setup (model checkpoints, inference parameters), and computational resources (GPU, memory).

B.1. Inference setup

Model checkpoints and configuration. The overview of the model checkpoints and configurations for the four models used in our experiments are presented in Tab. 6 and Tab. 7. For all experiments, we use the default settings from the original model provider. While Wan 2.2 supports multiple resolutions (480p and 720p), we found that setting the resolution to a larger size than 480p leads to a very slow inference speed, making the evaluation using three datasets very difficult. Thus, we resorted to 480p resolution for Wan 2.2 generation. We note that we perform all our experiments by generating 1 video either using a

	Model	Wan 2.2	Wan 2.1	LTX-Video
Base config.	Video length	5s	5s	5s
	Num. of frames	81	81	121
	Denoising steps	50	50	30 + 10
	Resolution	832×480	832×480	1216×704
	CFG scale	5.0	5.0	3.0
	Miscellaneous	-	CLIP conditioning	Two-stage inference
ALG config.	t_{trans}	0.1	0.2	0.1
	κ_*	2.5	2.5	4.0
	First-frame override	True	True	False
	Low-pass filter delaying	0.04	-	-

Table 7. Details for experiment with each image-to-video model.

single NVIDIA H200 GPU or using a single NVIDIA H100 GPU (*i.e.*, no multi-GPU inference was used).

ALG configuration. The bottom row of Tab. 7 summarizes the hyperparameters (t_{trans}, κ_*) for our main experiments (Tab. 2). To determine the hyperparameters, we take 20 prompts (out of 246, randomly chosen) from the VBench evaluation set and apply a grid search to determine the best hyperparameter set. Specifically, we search within $t_{\text{trans}} \in \{0.04, 0.1, 0.2\}$ and $\kappa_t \in \{1.6, 2.5, 4\}$. Note that as shown in Fig. 5a and Fig. 5b, most small t_{trans} values and moderately large κ_* values show reasonable enhancement in dynamic degree, while maintaining video quality. Based on our exploration, the parameters detailed in Tab. 7 are those we found to yield the most advantageous increase in dynamic degree while maintaining or even often improving the overall generation quality. For the Gaussian blur experiments of our component analysis in Sec. 5, we use kernel size of $0.05 \times$ height pixels and σ_{blur} of 80. Finally, note that we low-pass filter the *latents* instead of the raw input conditioning image for all our ALG experiments, as the latents are the actual inputs to the denoiser model.

B.2. Evaluation set

As explained in Sec. 4, we utilize and curate from three benchmark datasets, namely, VBench-I2V [24] test set, PE Video Dataset [4], and VidProM [57]. **VBench-I2V** is an image-to-video benchmark dataset, consisting of image-prompt pair data, as well as a set of evaluation metrics to assess the I2V generation performance. From the image-prompt pairs, we select all prompts except for those used for measuring background quality or camera motion instruction following. We exclude them due to the high inference cost while focusing on the motion and video quality related metrics which are the focus of our evaluation. This gives us 246 prompts and images from the VBench-I2V dataset.

Additionally, to evaluate the effectiveness of ALG for improving motion in real video frames, we curate an image-to-video benchmark dataset from **PE Video Dataset (PVD)**. PVD is a video-text pair dataset including pairs of real videos and expert-annotated captions, and we take 100 random video-caption pair from the entire set. From each video, we took the first frame of the video and used it to construct the input image-caption pair dataset for the image-to-video generation.

Finally, we curate 750 prompts from **VidProM** to evaluate the capability of ALG to enhance image-to-video motion dynamics for synthetic image inputs generated using a state-of-the-art text-to-image (T2I) model. VidProM is a text prompt dataset for text-to-video generation, and we randomly select 750 prompts (after filtering our erroneous prompts) from the set, and use a T2I generation model (*i.e.*, FLUX.1-dev [30]) to construct image-prompt pairs for image-to-video generation.

For all results in our experiments in Sec. 4 and Sec. 5, we evaluate each method with one seed and using all prompts in each curated benchmark dataset. It is also worth noting that our evaluations utilize 5-second videos across all base models. This presents a more demanding scenario compared to the VBench leaderboard [24] for I2V generation, which reports results for 2-second videos, demonstrating the capability of ALG to maintain performance over longer temporal sequences.

B.3. Motion augmentation of prompts

In this section, we explain in detail the prompt motion-augmentation technique used for the results in Tab. 4 of Sec. 5. While ALG enhances motion dynamics of I2V generation by adaptively modulating the frequency component during the video sampling process, it is also possible to alter the input text prompts so that the generated videos become more dynamic, orthogonally to ALG. To test the efficacy of such prompt augmentation, as well as ALG's effectiveness combined with such method, we use an LLM (Gemini 2.5 Flash; [7]) in order to modify the prompts to become more dynamic by emphasizing motion-related parts in the prompts. Specifically, we provided all 246 prompts from the VBench-I2V test set [24] and

instructed Gemini 2.5 Flash to make each prompt more dynamic, by using the following prompt: "*For each of these text descriptions of video, make it more dynamic to greatly enhance motion. Do not add new elements; enhance the existing descriptions.*" We then provided all 246 prompts right after this instruction, with each prompt line-separated. The LLM was instructed not to add any new elements to ensure that the prompts do not introduce objects not present in the input image, potentially introducing misalignments between images and prompts. We include all prompts resulting from this augmentation in our supplementary materials. Then, the baseline method (CFG) and our method (ALG) were tested using this motion-enhanced VBench test set on Wan 2.2, as presented in Tab. 4. It is worth noting that both methods benefit from the prompt augmentation, but ALG without augmentation already surpasses CFG in terms of dynamism.

B.4. Input prompts for generation

Table 8 summarizes the text prompts and specific image-to-video models utilized to generate the videos shown in Fig. 4.

Location	Model	Text Prompt
Top (Larger Videos)		
Video 1	Wan 2.2	A white car is swiftly driving on a dirt road near a bush, kicking up dust.
Video 2	Wan 2.2	A beach ball floats up into the sky, realistic handheld style, style of Ken Loach, camera pans up following.
Video 3	LTX-Video	A dog leaping through the air to catch a frisbee in a sunny park.
Video 4	Wan 2.2	create an image depicting the moment Luffy traverses through the dimensional rift, with vibrant colors and swirling energies 8k [the shot should create a lasting impression of horror and shock].
Bottom (Smaller Videos - Clockwise from top-left)		
Video 1	Wan 2.2	A space station orbited above the Earth.
Video 2	Wan 2.2	A princess dancing in a pink dress under a pink and purple sky, tiny shining particals falling from the sky.
Video 3	Wan 2.1	High Speed Super Cute Nuclear Christmas Explossiono, Sci-fi, Virtual Reality 3D.
Video 4	Wan 2.1	Create an 8k video of a beautiful woman with long blonde hair and blue eyes. She is wearing a red dress and high heels. She is walking on a busy street in New York City, smiling and waving at the camera. The video should have a smooth zoom in and out effect, and a slow motion effect when she turns her head.
Video 5	Wan 2.2	A child kicks a bear in a Finnish forest. the weather is sunny. zoom in.
Video 6	Wan 2.1	A photorealistic blue haired Tooth Fairy called Molly fights the many teeth creatures with her magical toothbrush.
Video 7	Wan 2.1	A beautiful girl with dark brown hair jumps out of the sparks of fire, The extension of gold thread, the splash of gold thread, the diffusion of ink, the smearing of ink.
Video 8	Wan 2.2	A large pot of soup filled with vegetables and meat.

Table 8. Input text prompts and models corresponding to the videos in Fig. 4.

B.5. Evaluation metrics

In this section, we provide a detailed explanation for the definition of each metric used in all our evaluation results (Sec. 4 and Sec. 5), including the VBench metrics [24], DOVER [59], and VisionReward [62].

VBench: Motion-related metrics. VBench includes one metric that assesses the degree of motion presented in the generated videos (*Dynamic Degree*). Note that the *VBench-Avg.* metric reported in Sec. 4 is the average value of all VBench metrics explained in this section, including Dynamic Degree.

- *Dynamic Degree*: This is the metric that is central to our evaluation of the enhanced motion dynamics of videos. For a single video, Dynamic Degree is computed by computing the magnitude of the top-5% optical flow between frames using RAFT [52], and then thresholding this value to determine whether each frame interval is *dynamic* or *static*. Then, the video is labeled as "dynamic" if the percentage of the dynamic interval exceeds a certain threshold. Both thresholds (frame interval flow magnitude, percentage of dynamic intervals) are determined adaptively according to the video resolution in order to ensure a fair cross-resolution comparison. Additionally, the FPS (frame per second) values is normalized to 8 FPS.

VBench: Image-to-video consistency metrics. We employ the I2V Subject Consistency metric from the VBench evaluation suite. This metric assess the fidelity of the video compared to the given input conditioning image. Note that the reported *VBench-I2V* metric in Sec. 4 refers to I2V Subject Consistency.

- *I2V Subject Consistency*: This metric assesses the consistency of the subject in the input image and the subject in the generated video frames. It is computed by measuring the DINO [5] similarity between input image and all video frames. Additionally, DINO similarity between consecutive frames is measured, and take the weighted average of these two similarities is used as the final metric value.

VBench: Video quality metrics. Video quality metrics include 5 sub-metrics: *Subject Consistency*, *Temporal Flickering*, *Motion Smoothness*, *Aesthetic Quality*, and *Imaging Quality*. The first 3 metrics assess the quality of the video in a temporally dependent manner, and the last 2 metrics measure the frame-wise quality. Note that our reported average metric *VBench-QS* in Sec. 4 is the average of these five metrics.

- *Temporal Quality - Subject Consistency*: This measures the consistency of the subject within a video, and is calculated by computing the DINO feature similarity between frames.
- *Temporal Quality - Temporal Flickering*: Unlike the first two consistency metrics (Subject Consistency and Background Consistency), which gauge semantic consistency, this metric focuses on the consistency of high-frequency local details by computing the mean absolute difference of frames.
- *Temporal Quality - Motion Smoothness*: This metric assesses the smoothness of the generated motions using the motion priors in a video frame interpolation model [32].
- *Frame-wise Quality - Aesthetic Quality*: This evaluates how aesthetically beautiful the individual frames are, using the LAION aesthetic predictor [31]. This predictor takes into account various beauty aspects including color combination, lighting, photo-realism, and the layout of the image.
- *Frame-wise Quality - Imaging Quality*: Measures how distortion-free the frames are. Distortion includes various imaging-related factors, such as over-exposure, noise, and blur, and is measured using the MUSIQ [26] image quality predictor.

DOVER: Aesthetic and technical quality metrics. DOVER [59] evaluates video quality by separating two perceptual dimensions: *aesthetic* (semantic appeal, composition, meaningfulness) and *technical* (low-level distortions), where each corresponds to one branch of the model. The overall DOVER score reported in Sec. 4 is a weighted sum of the two, using weights defined in the original DOVER work [59], determined to best match human judgments.

- *DOVER – Aesthetic Quality*: This metric measures the aesthetic perception of a video, focusing on semantic content, composition, object arrangement, and overall visual pleasantness. This is computed from an *Aesthetic View* produced by spatial downsampling and sparse temporal sampling, which preserve semantics while suppressing distortions.
- *DOVER – Technical Quality*: This measures the presence of low-level distortions, such as blur, noise, exposure errors, and jitter. DOVER constructs a *Technical View* by sampling fragmentary spatiotemporal patches so that the metric reflects technical fidelity only, not global semantics.

VisionReward: Human-preference reward model for videos and images. VisionReward [62] is a multi-dimensional human preference reward model for images and videos, built using a VLM model (QWen2-VL [56]). It decomposes human judgments into several dimensions (e.g., alignment, composition, stability, dynamics) via binary checklists; these are linearly weighted and summed into a single interpretable score. We report the single overall VisionReward score in Sec. 4.

- *VisionReward - Overall score*: Videos are evaluated on multiple dimensions (*e.g.*, motion realism, camera motion, stability, physics) via checklist questions; answers are mapped to binary features and a linear regression produces the final score.

C. Additional results

In this section, we present and explain additional experimental results of ALG, as well as the information regarding the experimental results demonstrated in our main text. As explained in Fig. 4, all prompts and models used for generation of videos, including Fig. 4, can be found in our supplementary materials.

C.1. Qualitative examples

We first provide additional qualitative examples for ALG in the following sections as explained below. The qualitative example videos can be viewed in a playable video file format in our supplementary materials.

- Appendix C.1.1: Additional qualitative results of video generation.
- Appendix C.1.2: Visualization of the effects of the design choice for the first unconditional term in ALG on the visual quality of the generated video quality, which is discussed in Sec. 3.2. (Fig. 7)

C.1.1. Additional qualitative results

We provide additional qualitative examples for our experiments in Fig 6. More qualitative example videos can be seen in video format in our supplementary materials, along with their prompts and I2V models used for generation.

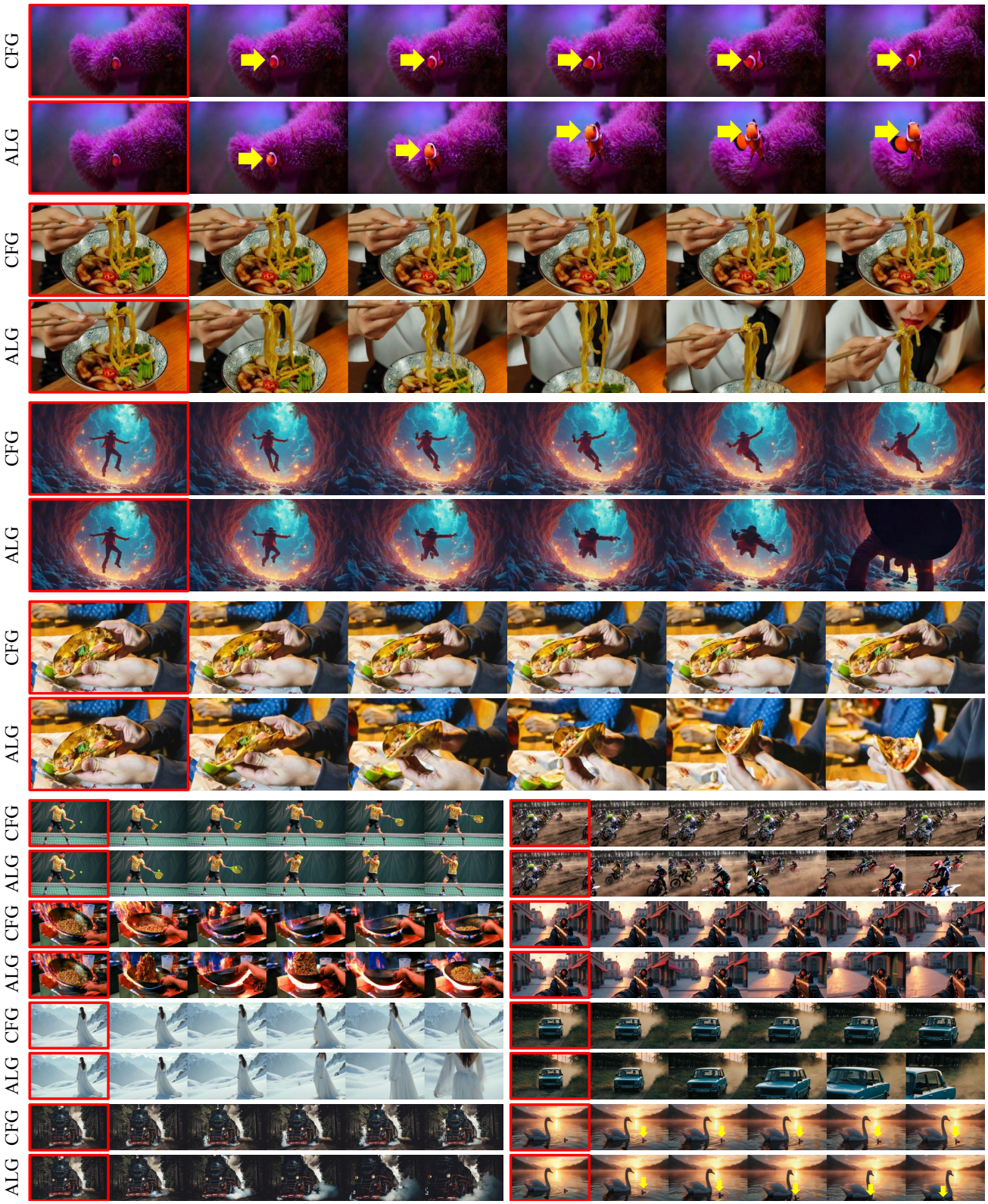

Figure 6. **Additional qualitative comparison between ALG and CFG.** We compare the videos generated by using default image-to-video generation method (CFG) and our method (ALG). The input conditioning frames are denoted with red outline. We observe that the videos using ALG show more dynamic motion (*e.g.*, larger object movement, animal movement, or human action, and more complex background movements). The list of prompts and models used for each video is included in the supplementary material.

C.1.2. Qualitative comparison for the design choice of ALG

We visualize the qualitative differences that arise when using the low-pass filtered latent for all unconditional terms in Eq. (3) (see Sec. 3.2 for more details) in Fig. 7. As shown, using the low-pass filtered latent for all unconditional terms often result in unstable video generation results, often characterized by distorted video frames or abrupt changes of scenes.

(a) Distortion of video frames

(b) Abrupt scene change

Figure 7. **Visual examples that warrant the design choice of ALG.** Using low-pass filtered input image for all terms of classifier-free guidance (denoted LP) often results in (a) distorted video frames, or (b) abrupt scene changes. ALG avoids this issue by grounding the generation in the original image's precise details and simultaneously providing motion guidance from the filtered image to enhance motion. This ensures both stability and visual integrity (see Sec. 3.2) while enhancing video dynamism.

Benchmark	Model	Method	Dynamic Degree	VBench Avg.	QS	I2V	TF	AQ	SC	IQ	MS	DOVER Overall	DOVER Aes.	DOVER Tech.	Vision Reward
VBench [24]	Wan 2.2	CFG	31.7	79.6	**85.4**	98.5	**98.2**	63.2	**96.2**	69.9	**98.9**	0.635	0.768	0.509	**0.183**
		ALG (Ours)	**39.0**	**80.5**	85.2	**98.5**	97.8	**63.5**	95.8	**70.0**	98.7	**0.637**	**0.781**	**0.530**	0.182
	Wan 2.1	CFG	28.9	79.1	**85.3**	**98.3**	**98.2**	**63.2**	**96.2**	**69.9**	**98.9**	**0.618**	**0.767**	**0.509**	**0.179**
		ALG (Ours)	**39.4**	**80.0**	84.5	98.0	98.0	62.4	95.0	68.3	98.8	0.614	0.761	0.508	0.176
	LTX-Video	CFG	15.5	77.8	**85.9**	**99.1**	**99.4**	**62.4**	**98.2**	**70.0**	**99.6**	0.625	0.755	**0.527**	0.175
		ALG (Ours)	**21.5**	**78.2**	85.4	98.9	99.2	61.6	97.1	69.6	99.6	**0.626**	**0.765**	0.522	**0.175**
PVD [4]	Wan 2.2	CFG	65.0	79.4	79.6	94.2	**97.2**	49.5	**89.7**	62.0	**98.4**	0.484	0.631	0.389	0.145
		ALG (Ours)	**69.0**	**80.3**	**79.6**	**95.0**	96.8	**50.0**	89.7	**63.0**	98.3	**0.512**	**0.660**	**0.415**	**0.145**
	Wan 2.1	CFG	66.0	79.4	**79.2**	94.0	**97.3**	**49.7**	**88.3**	62.2	**98.4**	0.492	0.645	0.393	**0.145**
		ALG (Ours)	**74.0**	**80.3**	78.8	**94.2**	97.0	49.4	86.1	**63.3**	98.2	**0.529**	**0.676**	**0.428**	0.141
	LTX-Video	CFG	66.7	80.0	**79.4**	**96.2**	**98.6**	49.4	**91.4**	58.5	**99.4**	0.428	0.490	0.391	**0.123**
		ALG (Ours)	**77.0**	**81.3**	79.3	95.3	98.2	**49.9**	89.0	**60.3**	99.2	**0.446**	**0.533**	**0.393**	0.122
VidProM [57]	Wan 2.2	CFG	27.3	79.1	85.6	**98.2**	**98.7**	**67.9**	**96.3**	**66.2**	**99.2**	**0.658**	**0.787**	0.560	0.096
		ALG (Ours)	**30.5**	**79.5**	**85.6**	98.0	**98.7**	67.7	96.2	66.1	99.2	0.657	0.785	**0.560**	**0.096**
	Wan 2.1	CFG	27.8	79.1	**85.6**	**98.1**	**98.8**	**67.6**	**96.2**	**66.0**	**99.3**	**0.652**	**0.776**	**0.558**	**0.093**
		ALG (Ours)	**36.3**	**80.0**	85.1	97.8	98.7	67.0	95.4	65.3	99.2	0.648	0.774	0.554	0.092
	LTX-Video	CFG	18.2	78.1	**85.9**	**99.2**	99.5	**67.6**	**97.8**	**64.7**	99.6	**0.627**	**0.764**	**0.527**	**0.089**
		ALG (Ours)	**23.0**	**78.7**	85.7	99.1	**99.5**	67.6	97.5	64.5	**99.6**	0.625	0.763	0.523	0.088

Table 9. **Comparison of CFG and ALG across all three models and benchmark datasets**. ALG consistently improves Dynamic Degree while maintaining video quality (all metrics except Dynamic Degree), leading to higher VBench average score (VBench-Avg.) in all cases.

C.2. Evaluation results

In this section, we present additional quantitative experimental results.

C.2.1. Full experimental results

We report the full experimental results for all three models (Wan 2.1/2.2 [55], LTX-Video [16]), and for all three benchmark datasets (VBench [24], PVD [4], VidProM [57]) in Table 9. Note that in this result, we show individual scores of each metric suite (VBench and DOVER). Specifically, for VBench, we report Temporal Flickering (TF), Aesthetic Quality (AQ), Subject Consistency (SC), Imaging Quality (IQ), and Motion Smoothness (MS) alongside the aggregate scores. For DOVER, we provide the breakdown into Aesthetic (Aes.) and Technical (Tech.) metric scores in addition to the Overall score. Consistent with the results reported in Sec. 4 and Sec. 5, ALG improves Dynamic Degree across all models and benchmarks while maintaining quality metrics. As a result, VBench average score (VBench-Avg.) increases in ALG for all cases in Tab. 9.

C.2.2. Applying low-pass filter to the input image with CogVideoX

In order to show that the dynamism gap between text-to-video and image-to-video model variants shown in Sec. 3.1 is not a phenomenon limited to Wan [55] models, we report the diagnostic results with an additional open-source video model that has both text-to-video and image-to-video model variation, namely, CogVideoX [63]. We show the enhancement of dynamic motion upon applying low-pass filtering with varying strengths, similarly to Fig. 3a (Wan 2.1 results). The results are visualized in Fig. 8. As shown, we observe that stronger low-pass filtering results in enhanced dynamic degree of the generated videos and a loss of per-frame video quality (aesthetic quality). This finding with CogVideoX is aligned with the results with Wan 2.1 presented in Fig. 3a of Sec. 3.1, and further supports our claim that low-pass filtering the input image results mitigates the motion suppression effect in I2V models.

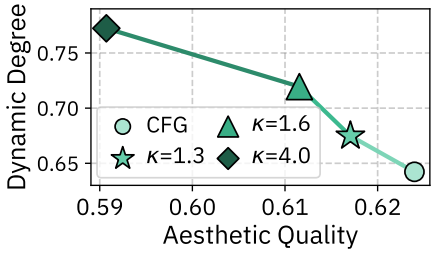

Figure 8. Low-pass filtering input image enhances motion in CogVideoX.

C.3. Feature map visualization

We provide additional results of feature map visualization as shown in Fig. 2 of Sec. 3.1. For Fig. 2, similarly to DINOv2 [40] and REPA [67], we inspect the middle layers of the DiT denoiser of Wan 2.1 by selecting the 5th frame of the intermediate activation at this layer. We provide additional visualizations for more diverse prompts, DiT layers, and t values in Fig. 9 and Fig. 10. Additionally, we include the feature map visualization for our method (ALG), which exhibits a similar behavior to the mitigation of the shortcut effect seen in naïve low-pass filtering, as ALG applies low-pass filtering at the early stages of the denoising process (where shortcut effect occurs predominantly).

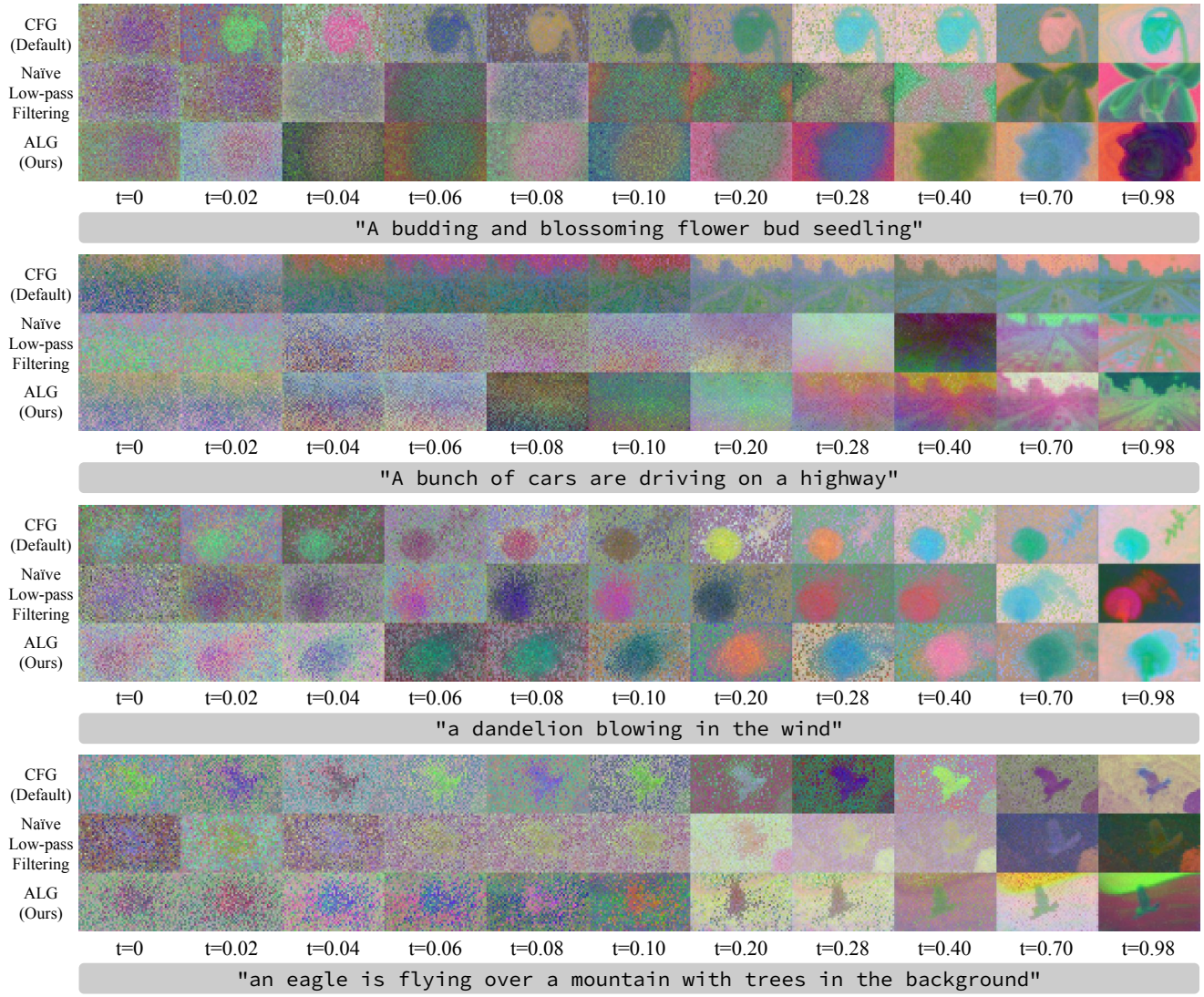

Figure 9. **Visualization of shortcut effect in I2V generation for the 15th layer of the DiT backbone.** For all default video generation results, we observe a premature refinement of the feature maps similar to Fig. 2. Low-pass filtering the input image avoids the shortcut effect and get refined more gradually. We observe similar effects in the case of our method (ALG), as it applies low-pass filter in the early stages of the sampling process. Best viewed in zoomed and colored monitor.

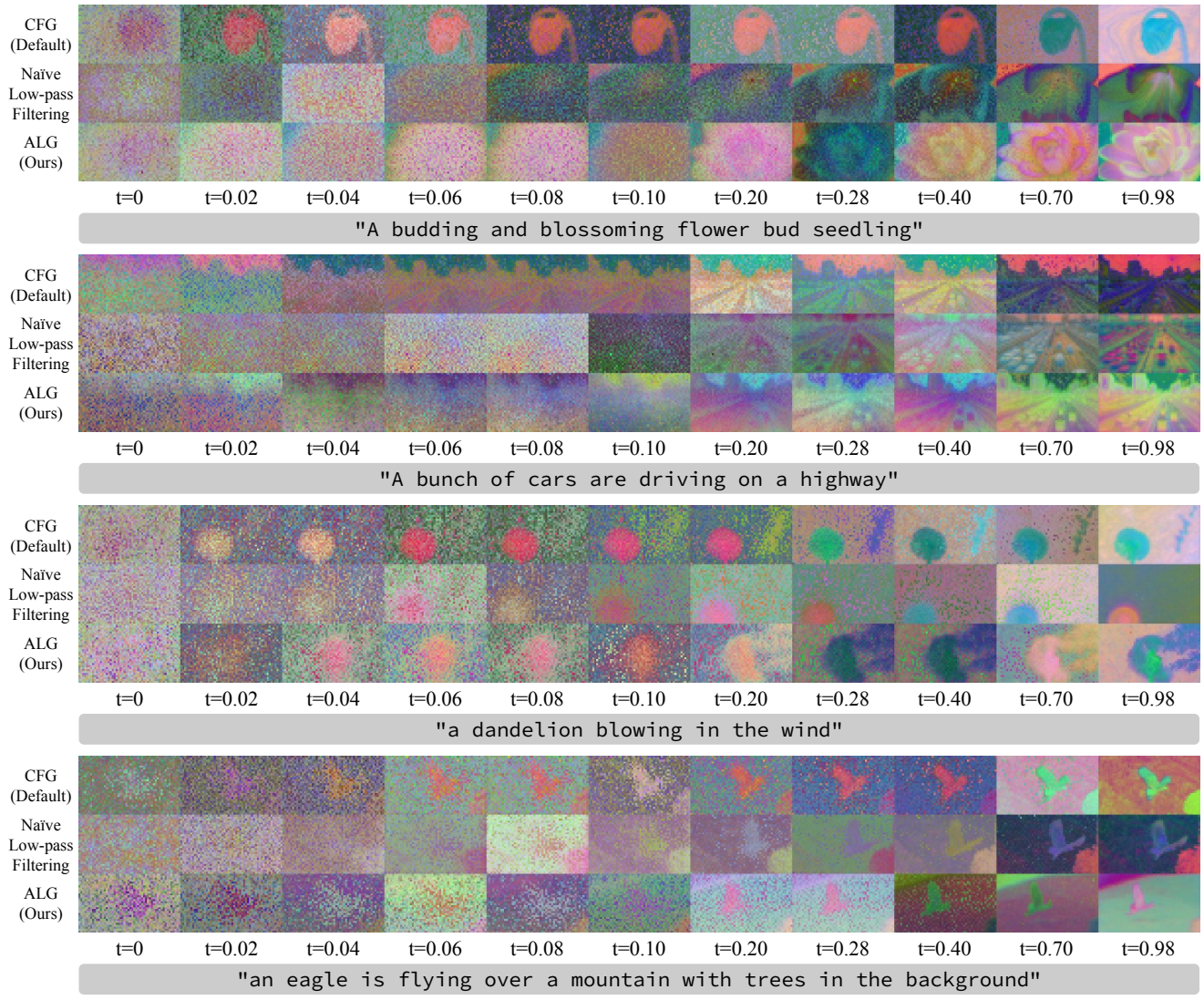

Figure 10. **Visualization of shortcut effect in I2V generation for the 22th layer of the DiT backbone.** Similar to Fig. 9, we observe that baseline method suffers a premature refinement of feature maps while low-pass filtering mitigates this effect, resulting in a more gradual refinement. Additionally, we observe similar mitigation in the case of our method (ALG), as it applies low-pass filter in the early stages of the sampling process. Best viewed in zoomed and colored monitor.