Experiment Log

CogVideoX-2b: Text-to-Video Generation & Hyperparameter Ablation Studies

Claudio Dragotta, Alessandro Dascola, Gabriele Ciccolella

1. Objective

This project explores text-to-video generation using CogVideoX-2b, a 2-billion parameter diffusion model by THUDM (Tsinghua University). The goal is twofold: (1) demonstrate inference across diverse prompts, and (2) systematically study how each hyperparameter affects the generated output by running controlled ablation studies on a fixed prompt.

2. Setup

- Platform: WSL2 (Ubuntu) on Windows, NVIDIA GPU with CUDA 12.x
- Python 3.12 | PyTorch with CUDA backend
- Hugging Face Diffusers >= 0.30.1 | Transformers >= 4.44.2 | Accelerate >= 0.33.0
- Model: THUDM/CogVideoX-2b loaded in float16 precision
- Memory optimizations: sequential CPU offload, VAE tiling and slicing
- A Colab notebook is provided to reproduce all experiments on T4/A100 GPU runtimes

3. Pipeline Architecture

The generation script (run_cogvideox.py) was designed with a modular, file-based prompt system. Each prompt is stored as a separate .txt file in the prompts/ directory. The script reads the prompt file at runtime via a CLI argument. This means adding a new scenario requires zero code changes -- just create a new .txt file and run the script.

A batch script (run_all.sh) iterates over all .txt files in the prompts/ directory and generates a video for each one automatically. This was used to produce all baseline videos in a single batch run.

All generation hyperparameters are exposed as CLI arguments, making it easy to run experiments with different settings without modifying any source code:

Parameter	Default	Role
--model	2b	Model variant: 2b (float16) or 5b (bfloat16)
--steps	50	Number of diffusion inference steps
--guidance	6.0	Classifier-free guidance scale
--frames	49	Number of video frames to generate
--fps	8	Output video frame rate (playback only)
--seed	42	Random seed for reproducibility

Experiment output organization

A dedicated script (run_experiments.sh) generates all experiment videos in an organized folder structure, enabling easy comparison between parameter variations. The script supports skip logic: if a video already exists it is not regenerated, allowing safe interruption and resumption. The output layout is:

- outputs/baseline/ - 9 prompts with default parameters (steps=50, GS=6, seed=42, frames=49)
- outputs/exp1_guidance_scale/ - panda_guitar with GS = 1, 6, 12
- outputs/exp2_steps/ - panda_guitar with steps = 10, 25, 50
- outputs/exp3_seed/ - panda_guitar with seed = 42, 123, 999
- outputs/exp4_frames/ - panda_guitar with frames = 25, 49

Total: 20 videos (9 baseline + 11 experiment variations). Each subfolder isolates one parameter variation, making it straightforward to compare the visual effect of changing a single hyperparameter.

4. Hyperparameter Impact Overview

Before running experiments, we identified which parameters are worth studying. The table below summarizes the expected impact of each parameter:

Parameter	Impact on output	Impact on time	Worth testing?
guidance_scale	Quality / prompt adherence	Minimal	Yes (key param)
num_inference_steps	Quality vs speed	Linear (high)	Yes (key param)
seed	Generation diversity	None	Yes (easy)
num_frames	Video length / VRAM	Proportional	Yes
fps	Playback speed only	None	No
model (2b vs 5b)	Quality (5b better)	Higher for 5b	Depends on HW

Based on this analysis, we selected four parameters for controlled experiments: guidance_scale, num_inference_steps, seed, and num_frames. For each experiment, we use the same prompt (panda_guitar) and vary only one parameter at a time, keeping all others at their default values.

5. Experiments

5.1 Baseline: inference across 9 prompts

We generated 9 videos with default settings (steps=50, GS=6, seed=42, frames=49) to validate the pipeline. Each video takes approximately 10-15 minutes to generate. All 9 prompts are available in the prompts/ directory (panda_guitar, gatto_hacker, samurai_pioggia, astronauta_luna, ronaldo_pizza, persona_panino, cane_parco, citta_futuristica, oceano_balena). All videos were generated successfully with consistent quality across realistic, fantastical, and cinematic scenarios.

5.2 Experiment 1: guidance_scale (1 vs 6 vs 12)

Prompt: panda_guitar | Fixed: steps=50, seed=42, frames=49 | Varied: guidance_scale

The guidance_scale controls classifier-free guidance -- how strongly the model conditions on the text prompt during denoising.

- **GS=1:** The panda shape is present but severely distorted and unstable across frames. In early frames the figure is a chaotic blend of colors with limbs morphing; by the last frame a panda face emerges but the body remains blurry. No guitar is visible at any point. The forest background is recognizable but noisy. The model captures the general theme but fails to produce coherent details. File size (630 KB) is smaller than GS=6, suggesting less visual complexity.
- **GS=6 (default):** Excellent quality. The panda is clearly rendered in cartoon/animated style with a red jacket, sitting on a stool, strumming an acoustic guitar in a bamboo forest. Other smaller pandas are visible in the background. Colors are natural and balanced. Frame-to-frame motion is smooth -- the panda's head and arms move naturally. This is clearly the sweet spot (file size: 767 KB, the largest, confirming the highest visual complexity).
- **GS=12:** The panda and guitar are very clearly defined -- even more vivid than GS=6. Colors are brighter with higher contrast; the red jacket pops more, bamboo is a more intense green, and lighting appears more dramatic with visible sun rays. A smaller panda appears on the right side. The overall look is more '3D rendered' and slightly artificial compared to the softer GS=6 result. File size (714 KB) sits between GS=1 and GS=6.

Conclusion: guidance_scale is the most impactful parameter for visual quality. GS=1 fails to produce coherent details; GS=6 is the sweet spot with natural colors and smooth motion; GS=12 produces sharper but more artificial-looking

results. Generation time is practically identical across all three values.

5.3 Experiment 2: num_inference_steps (10 vs 25 vs 50)

Prompt: panda_guitar | Fixed: GS=6, seed=42, frames=49 | Varied: inference steps

The number of diffusion steps controls how many denoising iterations the model performs. More steps = more detail, but slower.

- **10 steps:** COMPLETE FAILURE. The output is a dark black blob on a bright green background with visible vertical scan-line artifacts. No recognizable content whatsoever -- no panda, no guitar, no forest. The denoising process was insufficient to produce any meaningful image. Ironically the file is the LARGEST (1.1 MB) because the noisy pattern creates high entropy that compresses poorly. This is the most dramatic finding of all experiments.
- **25 steps:** Surprisingly excellent quality. The panda is clearly visible in a red jacket, playing guitar in a bamboo forest -- very close to the 50-step version. The composition is similar (same seed), with the panda centered and bamboo behind. Minor differences: slightly less detail in the bamboo textures and the background pandas are less defined. For practical purposes, this is nearly indistinguishable from 50 steps at roughly half the generation time.
- **50 steps (default):** Full quality baseline. Panda with red jacket, guitar, bamboo forest, background pandas -- all rendered with maximum detail and smooth motion. File size (767 KB) is smaller than 10 steps, confirming that coherent images compress better than noise.

Conclusion: the jump from 10 to 25 steps is the single biggest quality leap in all experiments -- from completely unusable to near-perfect. The jump from 25 to 50 is marginal. This means 25 steps is an excellent default for rapid iteration, and the common assumption that 'more steps = proportionally better' is wrong. There is a critical threshold around 20-25 steps below which the model fails.

5.4 Experiment 3: seed variation (42 vs 123 vs 999)

Prompt: panda_guitar | Fixed: GS=6, steps=50, frames=49 | Varied: random seed

The seed controls the initial random noise tensor. Different seeds produce different "interpretations" of the same prompt.

- **Seed=42 (default):** The baseline version. Panda centered on a stool, guitar clearly visible, multiple smaller pandas watching on the right. Warm balanced lighting, dense bamboo forest backdrop. File size: 767 KB.
- **Seed=123:** Genuinely different composition -- not just a slight variation. The panda is positioned more to the left, the bamboo arrangement is sparser, and a smaller panda on the right appears to hold its own tiny guitar. The lighting is slightly darker and moodier. The overall style feels more 'cinematic'. File size: 751 KB.
- **Seed=999:** A third distinct interpretation. The camera angle is wider, the panda sits on a wooden bench/stool in a sunlit clearing. Rocks and a stream are visible in the background. Smaller pandas appear at the edges. The color palette is warmer with more golden-green tones. File size: 621 KB (less visual complexity).

Conclusion: the seed produces genuinely different artistic interpretations -- different camera angles, lighting, compositions, and background elements -- not just minor variations. Quality remains consistent across seeds. This makes seed variation a powerful tool: generate 3-5 versions and pick the best composition. Zero impact on generation time.

5.5 Experiment 4: num_frames (25 vs 49)

Prompt: panda_guitar | Fixed: GS=6, steps=50, seed=42 | Varied: frame count

Frame count determines video length (at 8 FPS: 25 frames = ~3s, 49 frames = ~6s). More frames require more GPU memory.

- **25 frames / ~3s:** Per-frame quality is identical to 49 frames (same seed, same denoising). The panda is clearly visible with guitar and red jacket. However, the clip feels abrupt -- the motion starts but cuts off before completing a natural cycle. File size: 337 KB (roughly half of the 49-frame version).
- **49 frames / ~6s (default):** Full motion cycle with the same per-frame quality. The panda picks up the guitar, strums, head and body move naturally, background pandas react. The extra duration allows the scene to develop into a more complete 'story'. File size: 767 KB.

Conclusion: frame count affects only duration and VRAM, NOT per-frame quality. The file size scales almost linearly (337 KB vs 767 KB). 49 frames is recommended for final outputs to allow complete motion cycles; 25 frames is useful for quick previews since the visual quality per frame is identical.

6. Cross-Experiment Analysis

By running all four experiments on the same prompt (panda_guitar), we can directly compare each parameter in isolation:

Rank	Parameter	Changes what?	Time impact	Recommendation
1	guidance_scale	Quality & adherence	None	Keep at 6 (default)
2	num_inference_steps	Quality & speed	Linear	50 final, 25 preview
3	seed	Composition & style	None	Try 3-5, pick best
4	num_frames	Duration & VRAM	Proportional	49 final, 25 preview

Key insights from the combined analysis:

- **Most surprising finding:** 10 inference steps produces completely unusable output (black blob on green noise), while 25 steps produces near-perfect results. There is a critical threshold around 20-25 steps -- not a gradual degradation. This disproves the assumption that quality degrades linearly with fewer steps.
- **guidance_scale vs steps:** these two parameters affect different dimensions. guidance_scale controls WHAT the model generates (prompt adherence), while steps controls HOW WELL it generates (detail and refinement). They are largely independent and can be tuned separately.
- **File size as proxy:** an unexpected finding: file size correlates with visual quality. The noisy 10-step video is 1.1 MB (high entropy = poor compression), while the clean 50-step version is only 767 KB. Coherent images compress better than noise.

Recommended workflow: (1) write the prompt, (2) generate a quick preview with steps=25 (half the time, nearly identical quality), (3) if the composition is good, re-run with steps=50 for maximum quality, (4) if the composition is not ideal, try 3-5 different seeds with steps=25, pick the best, then run the selected seed at steps=50 for the final output.

7. Conclusions & Reflections

Key takeaways from this project:

- **CogVideoX-2b is remarkably capable:** despite only 2B parameters, it generates coherent short videos from complex prompts in 10-15 minutes on consumer hardware. The quality spans from photorealistic (dog jumping, person eating) to stylized animation (panda, cat hacker) to cinematic compositions (samurai in rain, cyberpunk city, whale at sunset).
- **Critical step threshold discovered:** 10 inference steps produces completely black/green noise, while 25 steps produces near-perfect results. This non-linear behavior is the most important practical finding: never go below ~20 steps, and 25 steps at half the time of 50

is the ideal setting for rapid iteration.

- **guidance_scale=6 confirmed as optimal:** after visual analysis, GS=6 produces the best balance of prompt adherence and natural appearance. GS=1 causes distortion and morphing artifacts, GS=12 produces sharper but slightly artificial results. The default is well calibrated.
- **Seeds unlock diversity:** different seeds produce genuinely different interpretations -- camera angles, lighting, compositions -- not just noise variations. This makes seed exploration the cheapest way to improve output quality: generate 3-5 candidates and pick.
- **Known limitation -- real people:** the 2b model struggles with specific real-world individuals (e.g. ronaldo_pizza produced a generic young man, not recognizably Ronaldo). The model excels with generic or fictional subjects.
- **Memory management is essential:** without sequential CPU offload and VAE tiling/slicing, the model would not fit in 15 GB VRAM. These optimizations are critical for accessibility on consumer GPUs.

Future directions: (1) test CogVideoX-5b for higher quality (requires more VRAM); (2) LoRA fine-tuning for domain-specific styles; (3) explore the 20-25 step threshold more precisely to find the minimum viable step count; (4) quantitative evaluation using FVD (Frechet Video Distance) or CLIP-based similarity metrics.