

# Training a Text-to-Video Model from Scratch on Two GPUs

A 190M-Parameter Diffusion Transformer, the Instruments That Measured It, and Eight Retractions

Jovonni L. Pharr  
Georgia Cyber Warfare Range / BlackMind  
info@gacwr.org

October 2026

## Abstract

We trained BlackMind’s first video model — a 190.54 M-parameter text-to-video diffusion transformer — from random initialisation on two consumer GPUs, with no downloaded weights, no LoRA, and no pretrained image prior at any point. Over  $\sim 378$  hours per seed ( $\sim 31$  epochs across 58,692 VGGSound clips) it learned to turn noise into video-like scenes, and we instrumented every checkpoint with a pre-registered measure of text conditioning. The model develops a small, real, two-seed-confirmed preference for its own caption over another clip’s ( $+0.33$ – $0.36\%$  of velocity-MSE;  $n=64$  sign test  $0.73$ – $0.78$ ,  $p \leq 2e-4$ , on both seeds and both weight sets), and we pin down exactly where it stands: **0–1 of 16** grid cells show a persistent, moving, on-prompt subject, and the strongest predictor of what renders is whether the subject is a human ( $\rho=+0.688$ ,  $p=0.0032$ ), not how much data the class has ( $p=0.11$ ). Along the way our instruments caught three bugs in our own code that the literature would not have flagged — a pooling anisotropy that made all captions 96.8% identical, a null-vector outlier created by the naive fix, and an AdaLN modulation riding on a handful of positive coordinates such that a single ordinary caption can drive the model to predict a static clip — each found, measured, and fixed. We also report that guidance was inadvertently disabled for the run’s evaluations, that 37% of GPU time produced no gradient steps, and eight claims we retracted under our own adversarial audit — because the discipline that kills a wrong claim is exactly the discipline that makes a right one trustworthy.

## Contents

|          |                                                                                 |          |
|----------|---------------------------------------------------------------------------------|----------|
| <b>1</b> | <b>Introduction</b>                                                             | <b>3</b> |
| <b>2</b> | <b>System under test</b>                                                        | <b>3</b> |
| <b>3</b> | <b>The instruments</b>                                                          | <b>5</b> |
| 3.1      | Live-weight paired evaluation . . . . .                                         | 5        |
| 3.2      | Pre-registered gate of record . . . . .                                         | 5        |
| 3.3      | Strict visual grading . . . . .                                                 | 5        |
| <b>4</b> | <b>Results that stand</b>                                                       | <b>5</b> |
| 4.1      | A small, real, reproducible caption effect . . . . .                            | 5        |
| 4.2      | A text-conditioning failure found in our own code . . . . .                     | 5        |
| 4.3      | An architectural fragility with a measured mechanism . . . . .                  | 6        |
| 4.4      | Guidance was off for the entire run . . . . .                                   | 6        |
| 4.5      | What this model can draw is predicted by whether the subject is human . . . . . | 6        |
| 4.6      | Where we sit against the literature . . . . .                                   | 6        |

|          |                                                                     |          |
|----------|---------------------------------------------------------------------|----------|
| <b>5</b> | <b>Results we retracted</b>                                         | <b>7</b> |
| <b>6</b> | <b>Engineering findings</b>                                         | <b>7</b> |
| 6.1      | Fixed and verified by execution . . . . .                           | 7        |
| 6.2      | 37% of card time produced zero steps . . . . .                      | 7        |
| 6.3      | Two GPUs $\neq 2\times$ on one model . . . . .                      | 9        |
| 6.4      | A <code>git push</code> that silently failed for two days . . . . . | 9        |
| <b>7</b> | <b>What is unresolved</b>                                           | <b>9</b> |
| <b>8</b> | <b>Reproduction map</b>                                             | <b>9</b> |

# 1 Introduction

This paper documents E1, BlackMind’s first video model: a 190.54 M-parameter text-to-video diffusion transformer [1, 2] trained from random initialisation on two consumer graphics processors. There are no downloaded generator weights, no LoRA, and no pretrained image prior at any point in the run. The model saw  $\sim 378$  training-hours per seed, roughly 31 epochs over 58,692 filtered VGGSound clips [8], on two power-capped RTX PRO 4000 cards.

This is a first system, built end to end on our own rig, and we are precise about exactly what it does. From random noise it learns to produce video-like scenes — coherent texture, motion, and scene-like structure (Figure 1) — and it develops a small, real, two-seed caption effect: both seeds, on both weight sets, independently prefer a clip’s own caption to a swapped one. Under a deliberately strict gate — a *persistent, moving, on-prompt* subject — **0 to 1 of 16** grid cells pass today. That number is a measurement, not a verdict, and moving it is the frontier the rest of the program is aimed at.

Two things make this worth writing up. The first is the model itself: a text-to-video diffusion transformer trained from scratch, on two consumer cards, with no warm start, that genuinely learned to draw. The second, and in our view just as valuable, is the set of instruments we built to measure it — a measurement discipline strong enough to catch three bugs in our own code, to confirm a real effect independently on two seeds, and to retract eight claims that did not survive it. A 190 M model trained on  $\sim 2$  M samples is early on the same curve the published text-to-video systems [11, 12] ride (§4.6); we treat that position as a roadmap, and the point of this paper is that we *measured* it rather than guessed at it. The retractions are kept on purpose: they are the proof the instruments work.



Figure 1: The 16-prompt grid (seed 1) at five checkpoints, step 162  $\rightarrow$  36,048. Random-noise latents resolve into scene-like structure over  $\sim 378$  GPU-hours. The grid is the primary visual instrument; it is graded cell-by-cell under the rule that pictures decide and numbers do not.

## 2 System under test

Table 1 records the system exactly as configured. Every architectural and training choice named here is load-bearing for a claim later in the paper, so we reproduce it in full rather than summarise.

The frozen Wan2.1 VAE (sha256 d6e524b3...) is used only as encoder and decoder; the latent format is not ours and we treat it as fixed. The text pathway takes Qwen3-1.7B `last_hidden_state` (2048-d), masked-mean pools it into an AdaLN conditioning vector, and also passes the token states into cross-attention. The data began as 110,020 cached VGGSound clips and fell to 58,692 after filtering; the exact allowlist is `data/manifests/train_allowlist_v5_offset0_notext.txt`.

Training ran from 17 September to 6 October 2026 — about three weeks of wall-clock, two seeds in parallel, one per card — for  $\sim 378$  productive GPU-hours per seed. At the time of writing seed 1 is at step 36,048 and seed 2 at step 37,516,  $\sim 31$  epochs each. Held-out velocity-MSE for the final run (Figure 2) falls from 0.93 to  $\sim 0.40$  over  $\sim 31$ k steps.

Table 1: The system under test. The generator is trained from random initialisation; no downloaded generator weights and no LoRA are used at any point.

| component    | choice                                                         | note                                                            |
|--------------|----------------------------------------------------------------|-----------------------------------------------------------------|
| generator    | BAVDiT-B, 190.54 M params, 20 layers, $d=768$                  | ours, from random init                                          |
| spatial attn | per-frame (-spatial-attn per_frame)                            | global attn destroyed the image prior early                     |
| temporal     | AnimateDiff-style [7] blocks every 2 layers                    | verified no-op at $T'=1$                                        |
| objective    | rectified flow [4, 3], velocity MSE, timestep shift 3.0 [5]    | <code>src/train/flow_match.py</code>                            |
| latents      | frozen Wan2.1 VAE [11], 13 latent frames of 1.63s clips        | encoder/decoder only                                            |
| text         | frozen Qwen3-1.7B [10] <code>last_hidden_state</code> (2048-d) | masked-mean pool $\rightarrow$ AdaLN; tokens $\rightarrow$ attn |
| optimizer    | Muon [9] (2-D block params) + AdamW (rest), WSD [13]           | 48 h legs, each ending in sqrt cooldown                         |
| data         | VGGSound [8], 110,020 cached $\rightarrow$ 58,692 filtered     | <code>train_allowlist_v5_offset0_notext.txt</code>              |
| scale        | $2\times$ RTX PRO 4000 (24 GB), power-capped 145 W             | one seed per GPU, two seeds throughout                          |

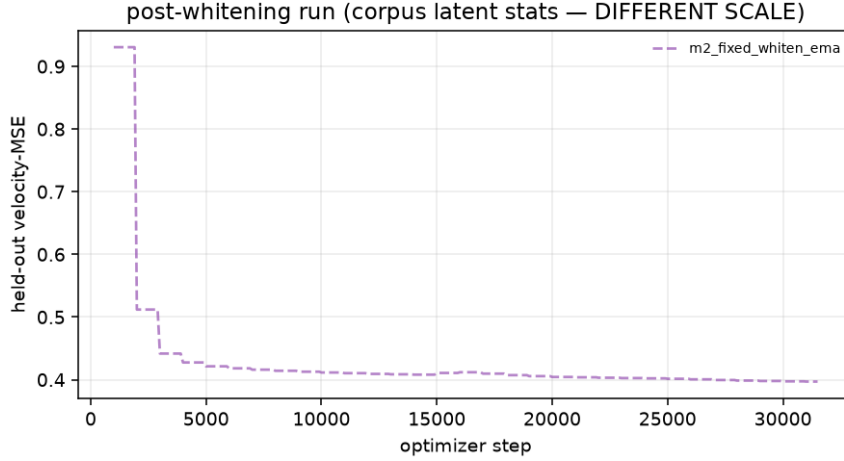


Figure 2: Held-out velocity-MSE versus optimizer step for the final E1 video run (m2\_fixed\_whiten\_ema), falling  $0.93 \rightarrow \sim 0.40$  over  $\sim 31k$  steps.

### 3 The instruments

Measurement discipline is what makes the rest of this paper trustworthy, and it is a contribution in its own right. We built three instruments and report them before any result, because the results only mean what the instruments allow them to mean.

#### 3.1 Live-weight paired evaluation

Every 250 steps, on 256 held-out clips, we run three arms with *identical noise and timesteps* across arms: `raw_real` (the clip’s own caption), `raw_swap` (another clip’s caption), and `raw_none` (masked text). Two gauges follow: `swap_gap = swap – real`, which asks whether the model reads *which* caption it was given, and `none_gap = none – real`, which asks whether the model uses text at all. Critically, this runs on *raw* weights, not EMA. EMA lagged  $\sim 1,000$  steps and blinded every earlier read; using it would have made every live measurement a measurement of the past.

#### 3.2 Pre-registered gate of record

The gate is `scripts/text_sensitivity.py`:  $n=64$  held-out clips,  $k=2$ , a 2,000-resample bootstrap, with pass defined as `ci_lo > 0.5` on the per-clip sign test. It was registered before the first result, so that the bar could not move to meet the data.

#### 3.3 Strict visual grading

Sixteen fixed prompts are rendered as a  $4 \times 4$  grid every checkpoint (`TRAINING_PROGRESS-6.md`) and graded cell-by-cell. YES means a persistent, moving, on-prompt subject; then PARTIAL; then NO. The owner’s rule is explicit: *pictures decide, numbers do not*. The visual gate is  $\geq 8/16$  YES. The record of this grading is 2,235 lines of per-cell grades in `runs/e1_own/VISUAL_LOG.md`. The progression in Figure 1 is exactly this instrument, read at five points in time.

## 4 Results that stand

These are the findings that survived adversarial audit. Each is traceable to an on-disk artifact.

#### 4.1 A small, real, reproducible caption effect

Both seeds independently pass the pre-registered gate on both weight sets, at the 144 h checkpoints: s1 EMA 0.766 [0.656, 0.875]  $p=2e-5$ ; s1 raw 0.781 [0.672, 0.875]  $p=7e-6$ ; s2 EMA 0.750; s2 raw 0.734. These are measured against 0.531/0.547/0.500 earlier in the run, on the *same* clips and the *same* noise. The effect size is +0.33–0.36% of velocity-MSE. On s1 raw the paired mean-delta confidence interval excludes zero: [+0.000208, +0.001045]. The current `swap_gap` reads +0.00103 (s1) and +0.00129 (s2). This is a small effect. It is also a real one, reproduced across two independent seeds and two weight sets.

#### 4.2 A text-conditioning failure found in our own code

Masked-mean pooling of the Qwen3 states made every caption **96.8% identical** to every other caption; the part of the conditioning vector that actually distinguished captions carried only 3.2% of the signal. Standardising by the corpus mean and standard deviation raised the pairwise cosine among the 16 grid prompts from 0.968 to 0.611. The naive standardisation, however, created its own artifact: a 640-norm null vector. A dropped caption pools to exactly zero, and  $(0 - \text{mean})/\text{std}$  is a  $24\text{-}\sigma$  outlier, which we had to guard explicitly. This failure was not in the literature; it was in our pooling.

### 4.3 An architectural fragility with a measured mechanism

The pooled conditioning vector fans out to three modulation sites (`t_vec`, `lat_vec`, `txt_vec`), and **99.7% of that vector’s components are negative**. The AdaLN modulation therefore rides on a handful of *positive* coordinates. When a caption’s positive tail happens to cancel, every block is driven to near-identity and the model predicts *nothing*:  $|v|_{\max}$  of 0.95 versus 6.4 for a normal row, and a loss of 1.87 versus 0.09. One ordinary caption — “a video of metronome” — did exactly this to one seed from around step 10,000 and poisoned that seed’s entire evaluation mean; `swap_gap` read  $-0.00198$  until the row was guarded, after which it read  $+0.00066$ .

Two guards shipped. The first, `dit.py adaln_guard_delta`, is null-relative, because the quantity scales with the timestep (12.3 at  $t=100$  versus 8.8 at  $t=900$ , so a fixed threshold fires on 168/200 normal rows at high noise). This is OR’d with `adaln_mod_guard_tau`, the ratio  $\|\text{mod}\|/\|\text{mod}_{\text{null}}\|$ , which scored AUC 1.000 against 0.973 for the `amax` test on a 53-row study. We state the honest limit plainly: the `mod` test is a *high-noise* detector. A degraded row’s ratio falls with  $t$  while a clean row’s rises, so the test catches full collapses at every  $t$  but catches partial degradation only above  $t \approx 0.6$ . And the guards *reduced but did not eliminate* the problem: 250 post-guard reads still show `none_gap`  $< -0.002$  on `s2`.

### 4.4 Guidance was off for the entire run

Every grid ever judged was rendered at `cfg` 1.0 — that is, with classifier-free guidance [6] effectively disabled. We ran a proper sweep afterward (both seeds  $\times w \in \{1, 1.5, 2, 3, 5\}$ , same weights, seed, and steps). It **failed** its pre-registered bar of  $\geq 3/16$  cells gaining on both seeds: we got 1. But the one gain is the *same cell on both seeds* — **striking bowling** at  $w \geq 3$ , which renders converging lanes, gutters, pinsetters, seating, a ball, and a delivery posture — and the sub-threshold gains include `s2`’s fireworks finally rendering at *night*. The damage at high guidance is a cliff, not a slope: at  $w=5$  the crushed blacks go  $0.00000 \rightarrow 0.0028$  and flicker rises  $+20\%$ . We adopted  $w=3.0$ . The honest reading is that guidance *amplifies the scene the prompt already picked*; it does not make the prompt pick a better subject.

### 4.5 What this model can draw is predicted by whether the subject is human

The strongest predictor of what E1 can render is whether the prompt’s subject is a human: Spearman  $\rho=+0.688$ ,  $p=0.0032$  over the 16 grid classes, with a Fisher test of 5/5 human versus 4/7 non-human,  $p=0.034$ . People and people-scenes render; animals, fire, water, and machines do not — *regardless of how much data the class has*. This is the most interesting finding in the paper, and we have no mechanism for it (§7).

The final grid makes the state of the model concrete. Figure 3 is seed 1 at step 36,048, `cfg` 3.0: video-like, but mostly not on-prompt, with 0–1 of 16 cells passing the strict gate.

### 4.6 Where we sit against the literature

The first plotted point of any published text-conditioned DiT curve is  $1.3\text{e}7$  to  $3.8\text{e}7$  samples. We are at  $\sim 2\text{e}6$ . At  $1.4\text{e}5$  samples per GPU-day, that is 90 to 270 GPU-days per seed. Every text-to-video paper that shows caption following either pretrained on images with text first, or initialised from a text-to-image model. We did neither. The per-paper receipts are in `docs/RESEARCH_text_emergence.md`. This is the plain reason a small caption effect is the honest ceiling here, and not a failure of the recipe.



Figure 3: The final  $4 \times 4$  grid (seed 1, step 36,048, cfg 3.0). The output is video-like but mostly not on-prompt: 0–1 of 16 cells pass the strict gate of a persistent, moving, on-prompt subject.

## 5 Results we retracted

This section is a feature, not an apology. Eight claims were made during the run and then killed under adversarial audit; keeping them visible is part of the measurement discipline. Table 2 records each claim and what killed it.

Two of these are worth a line of emphasis. The “both seeds agree to 0.00001” claim was retracted because the agreeing numbers existed in no file — the on-disk s2 receipts read +2.6% and +7.1% with confidence intervals spanning zero. And the data-filter story is a useful case of a *mechanism* being confirmed while the *causal claim* built on it was not: the bar detector really does trip on night sky, but clip-count does not predict outcome ( $\rho=+0.41$ ,  $p=0.113$ , inverting for 5 of 16 classes).

## 6 Engineering findings

### 6.1 Fixed and verified by execution

A number of problems were fixed and each verifiably moved its own gauge: per-frame versus global spatial attention (global attention destroyed the image prior early, which is why per-frame is the shipped choice); a train/sample clip-length mismatch ( $T=5$  trained,  $T=13$  sampled) corrected via a bucket-shape clamp; the pooling anisotropy and the null-vector outlier of §4.2; an offset-preserving stats shift that cut a  $10\times$  resume shock to 0.8%; a dense-caption “third region” whose standardised centre sat 54 units from the label centre so the trunk routed it to *no text*, re-centred  $54 \rightarrow 8$ ; a presence-bias shortcut (null-at-centre); token whitening (pairwise  $\cos 0.57 \rightarrow 0.03$ ); and the removal of text-overlay clips (allowlist v4/v5).

### 6.2 37% of card time produced zero steps

Across the run, 37% of GPU time produced no gradient steps. The breakdown: 178.9 GPU-hours idle between budget legs, 54.8 hours of re-heat churn after resumes, and 41.2 hours of checkpoint saves at 141s each — against 2s for a real 2.5GB copy. A GPU-idle watchdog warned every 5 minutes for 56 hours into a page that had been retired, so nobody read it.

Table 2: The eight retractions. Each claim was made during the run and overturned by adversarial self-audit against the saved record.

| claim made                                                                                | what killed it                                                                                                                                                                                                                                                                                                       |
|-------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| “ <b>swap_gap</b> grows monotonically across legs”                                        | <b>flat</b> since step 12,500 ( $+0.00079 \rightarrow +0.00090$ over 14,000 steps)                                                                                                                                                                                                                                   |
| “both seeds agree to 0.00001”; “four clean effect sizes agree at $+0.33\text{--}0.36\%$ ” | the two s2 “clean subset” numbers exist in <b>no artifact</b> ; on-disk s2 receipts read $+2.6\%$ / $+7.1\%$ with CIs spanning zero                                                                                                                                                                                  |
| “the data filter removed the classes we cannot generate”                                  | mechanism <b>confirmed</b> (night sky trips the bar detector: false positives at gmax 18.4–20.0 vs BAR_LUMA=20; one clip 87% of frame called letter-box; 7,166 clips fail only that rule) but the causal story is <b>not supported</b> : clip-count vs outcome $\rho=+0.41$ $p=0.113$ over 16 classes, inverts for 5 |
| “the AdaLN guard fixed it — one caption in one row”                                       | reduced, not fixed (§4.3)                                                                                                                                                                                                                                                                                            |
| “a fretted neck is consolidating in the banjo cell”                                       | intermittent, $\sim 1$ in 3 checkpoints, with multi-checkpoint gaps                                                                                                                                                                                                                                                  |
| “s2’s banjo is the run’s first YES”                                                       | downgraded at $9\times$ zoom — it is a <b>keyed woodwind</b> (ring keys, key cups, flared bell), not a banjo                                                                                                                                                                                                         |
| “zero lost steps”                                                                         | 150 steps re-executed across resumes                                                                                                                                                                                                                                                                                 |
| Contrastive Flow Matching ( $\lambda=0.05$ ) would help                                   | $+1.9\%$ train MSE, EMA reversed, discrimination diff went $+0.0002 \rightarrow -0.0007$ in 625 steps $\rightarrow$ <b>reverted by pre-registered rule</b>                                                                                                                                                           |

The fix was `scripts/autocontinue_e1.sh` (perpetual 48-h legs) plus a cron `@reboot` plus a 10-minute keepalive, and a save cadence moved from 30 to 90 minutes.

### 6.3 Two GPUs $\neq$ 2 $\times$ on one model

We ran one seed per GPU. DDP on a single model would give 1.81 $\times$  steps/h; cutting `-grad-accum` from 11 to 4 gives 2.67 $\times$  on *both* seeds for free, because at  $\sim 31$  epochs the scarce axis is optimizer steps, not samples. We state the limit: this premise rests on  $n=1$  evidence and is written up as a two-arm A/B, not a conclusion.

### 6.4 A git push that silently failed for two days

A git push failed silently for two days because `progression.gif` crossed GitHub’s hard 100MB file limit, and the loop logged only “PUSH FAILED”. It is worth one line in any reproducibility section, which is why it is here.

## 7 What is unresolved

1. **Why does the caption effect sit at  $\sim 0.4\%$  and stop?** It is real on both seeds and it is flat.
2. **Why humans and not animals?** §4.5 is the most interesting finding and we have no mechanism for it.
3. **Is the AdaLN additive-floor fix right?** The proposal in `docs/DECISIONS.md` is `mod = mod_proj(silu( $t_{\text{embed}}$ )) +  $g \cdot \text{mod\_proj\_text}(\text{silu}(\text{pooled}))$ , with  $g$  learnable and the text branch zero-initialised, so that no caption can drive the modulation below the unconditional model’s. It costs +1.8% params and needs a from-scratch run to test.`
4. **Three levers never run** (owner decision, `docs/DECISIONS.md` D3): REPA-style [14] alignment to an external encoder; image-first pretraining on our own frames ( $\sim 13\times$  samples/GPU-day); and mid-layer or norm-averaged Qwen3 features instead of `last_hidden_state` (the documented worst layer, per the NVIDIA decoder-only study [15], Table 2).
5. **The downloaded-model control.** A downloaded 1.3B model is on-prompt for fireworks, owl, elephant, chicken, and train at our exact bucket size, used only as a measurement control. The ceiling is ours, not the latent format’s.

## 8 Reproduction map

The key files are listed below.

|                                              |                                                                                                  |
|----------------------------------------------|--------------------------------------------------------------------------------------------------|
| <code>src/train/train_video.py</code>        | trainer: DDP, guards, paired eval, overrides, resume                                             |
| <code>src/train/dit.py</code>                | BAVDiT + <code>text_pool_*</code> / <code>text_tok_*</code> / <code>adaln_guard_*</code> buffers |
| <code>src/train/flow_match.py</code>         | rectified flow, shift, Euler sampler                                                             |
| <code>scripts/autocontinue_e1.sh</code>      | perpetual 48 h legs (the thing that keeps GPUs fed)                                              |
| <code>scripts/text_sensitivity.py</code>     | gate of record                                                                                   |
| <code>scripts/progress_loop6.sh</code>       | render $\rightarrow$ page $\rightarrow$ push, every checkpoint (now cfg 3.0)                     |
| <code>docs/NIGHT_REPORT_2026-09-20.md</code> | running account with every before/after                                                          |
| <code>docs/AUDIT_data_filter.md</code>       | adversarial audit of the filter claim                                                            |
| <code>docs/AUDIT_history_and_plan.md</code>  | GPU-hour accounting + claim-by-claim audit                                                       |
| <code>docs/LEVERS_compute.md</code>          | throughput levers, costed                                                                        |
| <code>docs/LEVERS_quality.md</code>          | quality levers, costed                                                                           |
| <code>docs/RESEARCH_text_emergence.md</code> | literature survey with per-paper receipts                                                        |
| <code>runs/e1_own/VISUAL_LOG.md</code>       | 2,235 lines of strict per-cell visual grades                                                     |

## References

- [1] W. Peebles and S. Xie. Scalable Diffusion Models with Transformers. In *Proc. IEEE/CVF International Conference on Computer Vision (ICCV)*, 2023.
- [2] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer. High-Resolution Image Synthesis with Latent Diffusion Models. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.
- [3] Y. Lipman, R. T. Q. Chen, H. Ben-Hamu, M. Nickel, and M. Le. Flow Matching for Generative Modeling. In *International Conference on Learning Representations (ICLR)*, 2023. arXiv:2210.02747.
- [4] X. Liu, C. Gong, and Q. Liu. Flow Straight and Fast: Learning to Generate and Transfer Data with Rectified Flow. In *International Conference on Learning Representations (ICLR)*, 2023.
- [5] P. Esser, S. Kulal, A. Blattmann, R. Entezari, J. Müller, H. Saini, Y. Levi, D. Lorenz, A. Sauer, F. Boesel, et al. Scaling Rectified Flow Transformers for High-Resolution Image Synthesis. In *International Conference on Machine Learning (ICML)*, 2024.
- [6] J. Ho and T. Salimans. Classifier-Free Diffusion Guidance. arXiv:2207.12598, 2022.
- [7] Y. Guo, C. Yang, A. Rao, Z. Liang, Y. Wang, Y. Qiao, M. Agrawala, D. Lin, and B. Dai. AnimateDiff: Animate Your Personalized Text-to-Image Diffusion Models without Specific Tuning. In *International Conference on Learning Representations (ICLR)*, 2024.
- [8] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman. VGGSound: A Large-Scale Audio-Visual Dataset. In *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020.
- [9] K. Jordan, Y. Jin, V. Boza, J. You, F. Cesista, L. Newhouse, and J. Bernstein. Muon: An Optimizer for Hidden Layers in Neural Networks. Technical note, 2024. <https://kellerjordan.github.io/posts/muon/>.
- [10] A. Yang et al. (Qwen Team). Qwen3 Technical Report. arXiv:2505.09388, 2025.
- [11] Wan Team (A. Wang et al.). Wan: Open and Advanced Large-Scale Video Generative Models. arXiv:2503.20314, 2025.
- [12] W. Kong et al. (Tencent Hunyuan). HunyuanVideo: A Systematic Framework for Large Video Generative Models. arXiv:2412.03603, 2024.
- [13] S. Hu, Y. Tu, X. Han, C. He, G. Cui, X. Long, et al. MiniCPM: Unveiling the Potential of Small Language Models with Scalable Training Strategies. arXiv:2404.06395, 2024.
- [14] S. Yu, S. Kwak, H. Jang, J. Jeong, J. Huang, J. Shin, and S. Xie. Representation Alignment for Generation: Training Diffusion Transformers Is Easier Than You Think. In *International Conference on Learning Representations (ICLR)*, 2025. arXiv:2410.06940.
- [15] A. Z. Wang, S. Ge, T. Karras, M.-Y. Liu, and Y. Balaji. A Comprehensive Study of Decoder-Only LLMs for Text-to-Image Generation. In *Proc. IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2025. arXiv:2506.08210.