MOVING ALPHABET: A CONTROLLED STUDY OF TRAINING DATA FOR TEXT-TO-VIDEO GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Text-to-video generation has advanced significantly over the past five years through scaling of model size, data, and compute. Unlike model architecture, training data is often underexplored. Real-world data curation is complex and non-trivial, involving clip selection from raw videos and captioning to create video-text pairs for learning text-to-video mappings. We study how data distribution and caption quality impact text-to-video models. To enable controlled experiments, we introduce **Moving Alphabet**, a procedural testbed that renders letters with varying fonts, colors, sizes, and positions, moving in different directions and speeds against a black background. This design allows precise control over data distribution and caption quality by corrupting ground-truth metadata. Our experiments yield three findings: a) a diverse and balanced distribution of video content and duration is critical for generalization; b) caption quality significantly affects both model performance and training efficiency, suggesting that text-to-video models are bounded by video understanding capabilities; and c) recovery through clean-data fine-tuning depends on the severity and type of caption corruption. We believe these insights can inform the development of large-scale text-to-video models, and we advocate for greater attention to the science of pre-training data.

1 INTRODUCTION

Driven by architectural innovation, increased computational power, and the scaling of training datasets, Text-to-video (T2V) generation models have progressed from producing short artifact-laden clips (Singer et al., 2022; Ho et al., 2022) to synthesizing complex scenes with synchronized audio (OpenAI, 2024; Polyak et al., 2024) that are often indistinguishable from real footage. Much of the existing literature has focused on model and architectural innovations, such as diffusion models (Ho et al., 2020; Rombach et al., 2022), transformer architectures (Vaswani et al., 2017; Peebles & Xie, 2023), and scaling laws (Kaplan et al., 2020), to explain these improvements. However, what is less discussed is the critical role of training data quality and composition. As the saying goes: "garbage in, garbage out." In this paper, we study the effect of training data on the quality of video models, specifically focusing on how video data characteristics, *e.g.*, content complexity and caption quality, impact the performance of T2V models. We also investigate whether fine-tuning can mitigate the effects of low-quality training data. These studies then offer guidelines on what to prioritize when curating training data for video models.

Briefly, a video data pipeline typically consists of two main stages: (a) **sourcing and clipping** that selects videos and extracts short clips typically capped at 10-20s using model-driven heuristics; (b) **video captioning** based on vision-language models to annotate each clip with detailed descriptions. Despite substantial progress in multimodal understanding models (Yang et al., 2025; Comanici et al., 2025; Team et al., 2025) for clip selection and captioning, it remains unclear how the video distribution and caption quality affect the model performance. Importantly, how much headroom remains in training data to further advance T2V generation? This work focuses on studying the following aspects of the training videos:

- What is a suitable visual content complexity and duration distribution in the training data?
- How does caption quality impact model performance and training efficiency?
- Can fine-tuning on a small high-quality dataset recover performance from a weak pretraining?

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

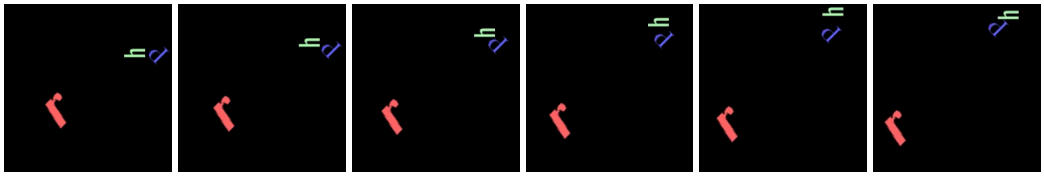


Figure 1: **3-letter example from Moving Alphabet.** Frames of the 17-frame clip (256×256, 8 fps), from left to right. Ground-truth caption: "A scene with 3 letters. Letter 1: A medium red lowercase r in decorative font, rotated 315 degrees. It starts at (0.25, 0.53) and moves at 149.0 degrees at speed 3.5, ending at (0.06, 0.64). Letter 2: A small light green lowercase h in condensed font, ..."

To systematically study these aspects, we need a testbed that allows precise control over video content complexity, duration, and caption quality. Working directly with real-world video data is impractical, so we introduce **Moving Alphabet**, a controlled video testbed where we render letters with varying fonts, colors, sizes, and positions, moving at different speeds and directions. See an example in Fig. 1. This design enables precise control over the data distribution and allows us to generate captions with varying precision and recall by manipulating ground-truth metadata. We also propose novel, fine-grained evaluation metrics for assessing visual quality and prompt adherence. **Our experiments yield the following recommendations for curating T2V training data**:

- **Prioritize diverse and balanced data distributions.** Models trained on an equal mix of 1-letter, 2-letter, and 3-letter scenes match or substantially outperform single-complexity specialists, even when evaluated on the specialists' own distributions. Mixing 2-, 4-, and 8-second videos yields the same trend. These results support diverse, balanced pre-training data across complexity and duration; we expect similar benefits on untested axes such as resolution.
- **Invest in high-quality captions from the start.** Performance drops sharply as precision (caption correctness) or recall (caption completeness) declines; reduced precision is more damaging, so prioritize correctness over completeness under a limited budget. Corrupted captions require up to 2–4× more compute to match the validation loss achieved with ground-truth captions. Perfect precision may be achievable at low recall, but full recall is ill-defined for real videos because the desired detail varies. Caption content and accuracy primarily bound T2V performance and controllability. Much recent progress in video generation can be traced to better captions from improved media-understanding models. While practitioners recognize captions' importance, our results quantify how caption quality affects generation performance and training efficiency.
- **Do not rely on post-hoc fixes for poor pre-training data.** Fine-tuning on a small clean dataset only partially restores generation quality in a representative precision-corruption case reported in §6. These results favor investing in caption quality during pre-training over recovering performance post-hoc.

While our findings are derived from a controlled testbed, we believe they offer valuable insights that can guide real-world T2V data curation strategies.

2 Related Work

Training data curation for video generation. DALL·E 3 (Betker et al., 2023) showed that recaptioning images improves text-image alignment, often more than dataset scaling. LAION-5B (Schuhmann et al., 2022) introduced large-scale filtering, and DataComp (Gadre et al., 2023) benchmarked data selection for CLIP training. For video, Panda-70M (Chen et al., 2024c) and InternVid (Wang et al., 2023) build large captioned video datasets using multiple cross-modality teachers, while ShareGPT4Video (Chen et al., 2024b) and LLaVA-NeXT (Liu et al., 2024) improve video understanding through better captions. Stable Video Diffusion (Blattmann et al., 2023) systematically ablates filtering and multi-stage curation, demonstrating their importance for downstream quality. VideoCrafter2 (Chen et al., 2024a) explicitly studies how to overcome data limitations for high-quality video diffusion. These studies use real-world data, where content, caption quality, and visual fidelity are entangled. Our controlled testbed varies each axis independently.

Curriculum and data mixing. Curriculum learning (Bengio et al., 2009) proposes training on easy examples first; data mixing studies how training composition affects performance. DoReMi (Xie et al., 2023) learns domain weights to speed up pretraining, The Pile (Gao et al., 2020) combines diverse text sources into a single training corpus, and Data Mixing Laws (Ye et al., 2024) formalize

how to predict performance from mixture proportions. Scaling laws (Kaplan et al., 2020) characterize how model and data size trade off, but say little about data *composition* at fixed scale. In video generation, progressive training from low to high resolution is common practice (OpenAI, 2024; Polyak et al., 2024), but the effect of mixing different scene complexities or video lengths has not been systematically studied. Our complexity and length mixing experiments (§4.1, §4.2) provide the first controlled evidence that equal mixing across content axes improves average performance over specialist or biased training.

Synthetic testbeds for understanding generative models. Controlled datasets with known ground truth have been instrumental for understanding representation learning. CLEVR (Johnson et al., 2017) tests compositional visual reasoning, dSprites (Matthey et al., 2017) and 3D Shapes (Kim & Mnih, 2018) provide disentanglement benchmarks with independently varying factors, and T2I-CompBench (Huang et al., 2023) evaluates compositional text-to-image generation. TinyStories (Eldan & Li, 2023) studies language learning from synthetic stories, while Physics of Language Models (Allen-Zhu & Li, 2024) uses synthetic data to study knowledge storage and extraction. Muennighoff et al. (Muennighoff et al., 2023) study data scaling laws for language models under controlled data constraints. Zhang et al. (Zhang et al., 2025) use a synthetic image–text benchmark to study bidirectional transfer between visual understanding and generation in unified VLMs. Our Moving Alphabet instead isolates how within-generation data properties affect text-to-video models.

3 Moving Alphabet: A Controlled Video Generation Testbed

Studying how training data influences T2V models requires precise control over video content and caption quality. These conditions are difficult to study with real-world data, where ground truth is ambiguous and confounding factors are unavoidable. We therefore introduce **Moving Alphabet**, a procedural testbed that renders letters moving against a near-black background. Because all attributes are sampled from known distributions, we can **(a)** introduce precise, measurable corruption into captions, **(b)** compose any target data distribution, and **(c)** evaluate models against exact per-video ground truth. Fig. 1 shows a three-letter clip and its ground-truth caption.

3.1 Data Generation

We construct the Moving Alphabet dataset by placing one to three letters with varying fonts, colors, sizes, and rotations on a black background, and animating them to move in different directions with different speeds. The main benefits of this procedural approach are that datasets can be scaled to arbitrary size and composed into controlled mixtures, enabling controlled experiment setups. We systematically vary video content complexity, duration, and caption quality, as detailed below.

Content complexity. We refer to scenes containing one, two, or three letters as 1L, 2L, and 3L. We sample each letter's appearance and motion, using distinct colors within a scene. Letters move at constant speed, bounce at canvas boundaries, and can occlude one another in multi-letter scenes.

Temporal complexity. Video length is independently configurable to roughly 2, 4, or 8 seconds (17, 33, or 65 frames at 8 fps and 256×256 resolution). Longer videos contain more boundary bounces and more complex trajectories.

Captions. Each video is paired with a templated natural-language caption generated from its metadata. The captions describe all appearance attributes, start and end positions using normalized coordinates, movement direction as an exact angle, speed, and any boundary bounces. Fig. 1 illustrates this format. Appendix B provides the full generation specification and additional examples.

3.2 Model Architecture and Training

Architecture & Baseline. We adopt a standard 800M-parameter MMDiT architecture (Esser et al., 2024) trained with flow matching (Lipman et al., 2023). Text conditioning uses frozen T5-XXL (Raffel et al., 2020) and CLIP ViT-L/14 (Radford et al., 2021) encoders, and video latents come from a pretrained causal 3D VAE. This standard recipe (Kong et al., 2024; Yang et al., 2024; Polyak et al., 2024) keeps controlled experiments affordable while using components from current video generation models. We train the baseline model from scratch on $8\times$A100 GPUs. The base configuration uses 150K videos with 17 frames at 8 fps and 256×256 resolution, converging in approximately two days.

The baseline metrics plateau by epoch 100. Hence, we use this checkpoint for baseline evaluation. During training, we randomly drop the caption with a probability 0.1 to enable classifier-free guidance at inference. More details in Appendix C.

3.3 Evaluation

A held-out set of 6,000 videos (2,000 per complexity level) is used for evaluation. Unless otherwise specified, we report results at classifier-free guidance (CFG) scale 1.0 to exclude the effect of guidance scale. Appendix G examines how guidance scale interacts with caption quality. Standard video metrics such as Fréchet Video Distance (FVD) (Unterthiner et al., 2019) rely on I3D features pretrained on natural videos, which transfer poorly to our synthetic domain. Since every video instead comes with exact ground-truth metadata, we use the following metrics:

- **Pixel accuracy**: foreground PSNR (FG PSNR) and foreground MSE (FG MSE scaled by 10^3 for readability), computed on the letter region only. These capture both visual fidelity and semantic correctness, as incorrect colors or positions increase pixel errors relative to the GT rendering.
- **Attribute metrics**: we measure each attribute (A_{color}, A_{size}, A_{dir}, ρ_{speed}) from generated video pixels and compare against ground-truth metadata, using accuracy for color, size, and direction and correlation for speed. For multi-letter videos, letters are localized using ground-truth trajectory bounding boxes to reduce cross-letter interference.

We report attribute metric calibration and the VAE reconstruction ceiling in Appendix A.

4 Investigation on Data Distribution

We first investigate the data collection choices on how we should balance different types of training data. We study two axes: content complexity (simple vs. complex scenes) and video length (short vs. long clips). Throughout this section, we hold caption quality fixed at ground truth, so that the only variable across conditions is the data composition under study.

4.1 Content Complexity

Experimental setup. The choice of what data to collect is naturally tied to the target evaluation scenarios. However, even when the evaluation distribution is known, it remains unclear whether the training data should match that distribution or whether another data distribution may yield better performance. We study this question by constructing training sets with different mixtures of 1L, 2L, and 3L videos and evaluate on different evaluation sets while keeping the total training set size fixed.

Following the baseline setting of 150K 2-second videos with 17 frames at 8 FPS, and ground-truth captions, we construct 6 different training sets. Three single-complexity conditions using pure 1L, 2L, or 3L videos, and three mixtures. The equal mix contains 50K videos at each level. The Heavy-1L mix has 100K 1L with 25K each of 2L and 3L, while the Heavy-3L mix has 100K 3L and 25K each of 1L and 2L. We evaluate each model on held-out videos at all three complexity levels. The results are shown in Tab. 1 using our proposed evaluation metrics. We summarize our findings below.

Out-of-distribution collapse. As expected, models trained on a single complexity level fail when evaluated on unseen complexities. Pure 1L training achieves only 0.357 direction accuracy and 4.1 dB FG PSNR on 3L scenes, while Pure 3L achieves 0.485 direction accuracy and 8.2 dB FG PSNR on 1L scenes. The degradation is not graceful, which suggests that the compositional skills do not emerge from single-object training, and per-object accuracy is lost when training exclusively on complex scenes where multiple overlapping objects dilute the supervision signal for each object.

Equal mixing matches or exceeds specialists. Empirically, we found that Mix Equal not only generalizes across complexity levels but also matches or exceeds the specialist models when evaluated on their own distributions. On 3L evaluation, Mix Equal reaches 12.9 dB FG PSNR versus Pure 3L's 10.1 dB, a gap of nearly 2.8 dB (Tab. 1). We hypothesize that the cause of this is compositional transfer. That is, simple scenes provide clean, unambiguous supervision for per-object primitives like color, direction, and size, while complex scenes introduce spatial reasoning between objects. These primitives then compose into better multi-object generation. This suggests that maintaining a diverse and balanced datamix is important in real-world text-to-video generation.

Table 1: Effect of training data complexity mixing on attribute accuracy and pixel quality. Models are trained on different distributions over 1L/2L/3L scenes with the same total data budget and evaluated at all complexity levels with CFG = 1.0. **Bold**: best per column; underline: second best.

(A) ATTRIBUTE ACCURACY

	Eval: 1-Letter				*Eval: 2-Letter*				*Eval: 3-Letter*				*Avg*			
Training Data	Col↑	Dir↑	Size↑	Spd↑	Col↑	Dir↑	Size↑	Spd↑	Col↑	Dir↑	Size↑	Spd↑	Col↑	Dir↑	Size↑	Spd↑
Pure 1L	.979	.983	**1.00**	**.998**	.309	.467	.283	.450	.165	.357	.172	.282	.484	.602	.485	.577
Pure 2L	.610	.975	.941	.915	.878	.984	.972	.994	.532	.761	.635	.504	.673	.906	.849	.804
Pure 3L	.379	.485	.346	.281	.894	.982	.995	.992	.886	.980	.995	.993	.720	.816	.779	.755
Mix Equal	**.981**	**.987**	**1.00**	.997	**.950**	**.987**	**1.00**	**.997**	.918	.981	**1.00**	**.996**	**.950**	**.985**	**1.00**	**.997**
Heavy-1L	.970	.943	.999	.950	**.950**	.977	.998	.994	.918	.975	.999	.993	.946	.965	.998	.979
Heavy-3L	.939	.949	.919	.915	.946	.984	.998	**.997**	**.920**	**.982**	.995	**.996**	.935	.972	.970	.969

(B) PIXEL QUALITY

	Eval: 1-Letter		*Eval: 2-Letter*		*Eval: 3-Letter*		*Avg*	
Training Data	FG MSE↓	FG PSNR↑	FG MSE↓	FG PSNR↑	FG MSE↓	FG PSNR↑	FG MSE↓	FG PSNR↑
	FG MSE values are $\times 10^3$; FG PSNR in dB.							
Pure 1L	70.1	12.4	383.4	4.3	402.0	4.1	285.2	6.9
Pure 2L	168.5	8.6	85.8	11.7	265.3	6.1	173.2	8.8
Pure 3L	187.7	8.2	109.7	10.4	113.7	10.1	137.0	9.6
Mix Equal	**63.9**	**12.8**	**55.4**	**13.1**	**56.0**	**12.9**	**58.4**	**12.9**
Heavy-1L	136.0	9.6	104.9	10.4	90.1	10.8	110.3	10.3
Heavy-3L	68.9	12.5	62.2	12.6	59.8	12.7	63.6	12.6

Table 2: Effect of training video length distribution on pixel quality. All conditions use 150K videos but differ in total tokens due to varying durations. Metrics are averaged over 1L/2L/3L complexity levels (CFG = 1.0). **Bold**: best per column; underline: second best.

		Eval: 2s (17f)		*Eval: 4s (33f)*		*Eval: 8s (65f)*		*Avg*	
Training Data	Tok. (B)	FG MSE↓	FG PSNR↑	FG MSE↓	FG PSNR↑	FG MSE↓	FG PSNR↑	FG MSE↓	FG PSNR↑
		FG MSE values are $\times 10^3$; FG PSNR in dB.							
Pure 2s	19.2	**58.4**	**12.9**	354.9	4.7	370.1	4.5	261.1	7.4
Pure 4s	34.6	312.4	5.3	101.3	11.1	355.8	4.7	256.5	7.0
Pure 8s	65.3	297.6	5.5	282.8	5.8	101.6	10.9	227.3	7.4
Mix Equal	39.7	65.6	12.4	**67.1**	**12.3**	71.7	12.0	**68.1**	**12.2**
Heavy-2s	29.4	74.0	12.0	72.4	12.0	79.7	11.6	75.4	11.9
Heavy-8s	52.5	223.8	6.6	81.3	11.2	**64.6**	**12.5**	123.2	10.1

Balanced mixing outperforms biased mixing. Heavy-1L underperforms Mix Equal even on 1L evaluation despite containing more 1L data, with a 3.2 dB FG PSNR gap (Tab. 1). This suggests that oversampling a single complexity level does not translate to better performance, even on that level's own distribution. Heavy-3L rivals Mix Equal on 3L metrics but sacrifices 1L and 2L quality.

Takeaways. Training on a single scene complexity fails to generalize. Models collapse out-of-distribution, and compositional skills do not emerge from single-object training alone. Oversampling a single complexity level does not improve performance even on that level's own evaluation set. For practitioners, we recommend maintaining a balanced mix of simple and complex scenes rather than filtering toward either extreme. Intuitively, simple scenes provide clean per-object supervision, complex scenes build spatial reasoning, and together they produce stronger multi-object generation at no extra data cost.

4.2 VIDEO LENGTH

Experimental setup. Video datasets in the wild span a wide range of durations. Longer videos contain richer temporal dynamics but are more expensive to process; shorter clips may lack complex motion patterns. We study how mixing training video lengths affects quality across durations.

We train six variants with 150K videos each that vary the mixture of 2-second, 4-second, and 8-second clips. All conditions share the same data budget of 150K videos and 100 epochs, matching the number

of training clips each model sees rather than the token count or compute, which necessarily differ across clip lengths. Three models are trained on a single duration each. Pure 2s is trained on 19.2B tokens, whereas Pure 8s is trained on 65.3B tokens, a 3.4× difference. The three mixed models combine all three durations. Mix Equal uses an equal number of examples per duration, with 50K examples each, for a total of 39.7B tokens. Heavy-2s allocates 100K examples to 2-second videos and 25K examples each to 4- and 8-second videos, totaling 29.4B tokens. Heavy-8s uses the reverse allocation, totaling 52.5B tokens. All models are evaluated on all three video lengths. We report pixel quality (FG MSE and FG PSNR) rather than per-attribute accuracy, since direction is defined from net displacement and degenerates for longer clips where bounces return the letter near its origin. Pixel metrics stay length-comparable and already reflect attribute errors. Tab. 2 reports pixel quality metrics averaged over complexity levels at CFG = 1.0.

Length-specific training does not generalize. As expected, models trained on a single duration perform well only at their training duration. For example, pure 4s achieves 11.1 dB FG PSNR on 4-second videos but drops to 5.3 dB or lower on other durations (Tab. 2). The model memorizes training durations rather than learning temporal representations that transfer across durations.

Equal mixing provides balanced performance. Mix Equal achieves 12.0–12.4 dB FG PSNR across all three evaluation lengths. It trades 0.5 dB at 2s, where Pure 2s achieves 12.9 dB, for gains of 1.2 dB at 4s and 1.1 dB at 8s over the specialists (Tab. 2). Averaged across all three lengths, Mix Equal achieves 12.2 dB, clearly the best overall (next best is Heavy-2s at 11.9 dB). This is not simply a compute effect: Mix Equal uses 39.7B tokens, comparable to Pure 4s at 34.6B and far less than Pure 8s at 65.3B, yet outperforms both at 4s and 8s. Exposure to diverse temporal dynamics enables generalization across durations. Relative to content complexity, video length shows weaker cross-condition transfer: single-duration specialists lose about 6.5 dB out of domain, versus about 4.5 dB for single-complexity specialists, making a balanced mix even more important here.

Biased mixing excels on majority durations but drops a minority one. Heavy-2s degrades gracefully, maintaining FG PSNR of at least 11.6 dB at all lengths, including the minority 8-second duration. Heavy-8s achieves the best 8-second quality (12.5 dB), surpassing even the 8-second specialist Pure 8s (10.9 dB) by 1.6 dB. However, it catastrophically fails on 2-second videos (6.6 dB), the least-represented duration in its training mix. This instability suggests that heavy oversampling of long videos creates fragile optimization dynamics where the model cannot reliably learn all shorter durations. By contrast, Mix Equal and Heavy-2s are stable across all lengths.

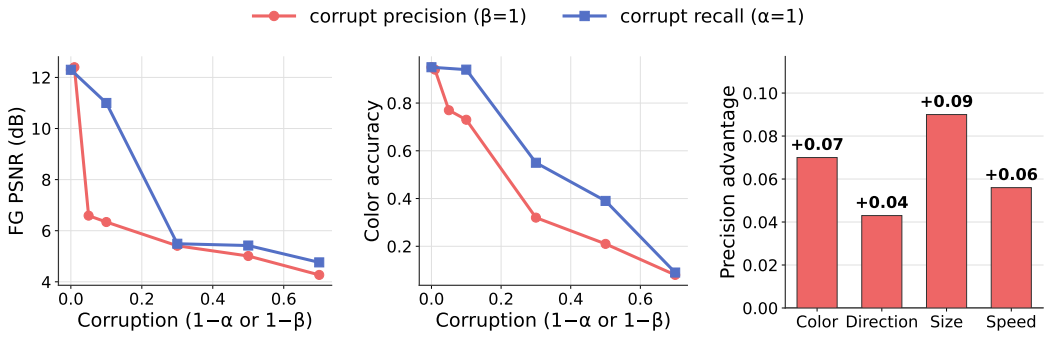
Takeaways. Training on a single video length fails to generalize: models memorize the training duration rather than learning transferable temporal representations. For real-world video datasets, we recommend sampling roughly equal numbers of short, medium, and long clips instead of duration-biased filtering; this generalizes broadly at lower token cost than training on long videos alone, and avoids the sharp short-clip degradation caused by oversampling long videos.

5 Investigation on Caption Quality

In practice, captions from VLMs or alt-text vary widely in accuracy and completeness, yet the effect of caption quality on generation performance is poorly understood. With our Moving Alphabet dataset, we control the precision and recall of the captions to study their impact on video generation.

5.1 Precision vs. Recall: Which Matters More?

Experimental setup. Based on the learnings from §4.1, we use Mix Equal in letter complexity as the default setting, and pure 2-second only for efficiency concern. To study caption quality in isolation from visual content, we degrade captions along two controlled axes while keeping the underlying videos the same. We define **precision** $\alpha \in [0, 1]$ as the fraction of stated attributes that are correct, and **recall** $\beta \in [0, 1]$ as the fraction of true attributes that are mentioned. For each training sample, we independently corrupt each attribute: with probability $1-\alpha$, a mentioned attribute is replaced with a random incorrect value; with probability $1-\beta$, a ground-truth attribute is omitted. Each of the 35 precision–recall conditions is trained on the same 50K-video subset for 200 epochs, with the full grid specified in Appendix D.1.



(a) FG PSNR (b) Color accuracy (c) Precision advantage

Figure 2: Effect of caption precision (α) and recall (β) on generation quality at CFG = 1.0. **(a, b)** FG PSNR and color accuracy when reducing precision (red, β=1) or recall (blue, α=1). **(c)** For each attribute, bars show the average performance difference between models with swapped precision and recall levels. Positive values favor higher precision over higher recall.

Fig. 2 reports results at CFG=1.0 on the fixed 6K held-out evaluation set described in §3.3. We have the following observations.

Both axes degrade pixel fidelity, and the drop is a sharp cliff. As Fig. 2a shows, FG PSNR falls along both axes: reducing recall from β=1.0 to 0.3 costs 7.6 dB (12.3 to 4.8), and reducing precision from α=1.0 to 0.3 costs 8.1 dB (12.3 to 4.3). Both matter because FG PSNR captures semantic correctness alongside visual quality: wrong and missing attributes both increase pixel error against ground truth. Crucially, the precision loss is not gradual but concentrated at high α: 5.8 of those 8.1 dB occur between α=0.99 and α=0.95 (12.4 to 6.6 dB), the steep initial drop of the red curve and a cliff from only a 4% change in precision. A safe zone exists only at $\alpha \geq 0.99$ with β=1.0.

Precision is the primary driver of attribute controllability. Fig. 2b shows color accuracy falling toward chance for $\alpha \leq 0.5$: at β=1.0 it drops from 0.95 at α=1.0 to 0.21 at α=0.5 and 0.08 at α=0.3, making color the most sensitive attribute to precision. This precision dominance extends beyond color: the mean precision advantage in Fig. 2c is positive for all four attributes (color, direction, size, and speed). Averaged across matched quality budgets, prioritizing correctness over coverage improves every attribute metric. Appendix F shows the individual comparisons. In short, *mentioning* an attribute helps pixel fidelity, but stating it *correctly* is what determines per-attribute controllability.

Takeaways. Reducing either precision (α) or recall (β) hurts generation quality, but precision is the more damaging axis: low precision degrades pixel fidelity more sharply than low recall and can collapse per-attribute accuracy even with complete captions. When forced to choose, *e.g.*, under a limited budget, correctness beats completeness: for practitioners curating training data, verifying that captioned attributes are accurate should take priority over maximizing caption length or coverage.

5.2 Caption Quality and Compute Efficiency

The previous section showed that caption corruption degrades final generation quality significantly. Here we ask: Does caption corruption also affect training efficiency? We measure the compute (in PFLOPs) needed to reach a fixed validation loss threshold, evaluated with ground-truth captions, to quantify how much harder it is for corrupted-caption models to learn the true data distribution. We select the threshold as the lowest validation loss that most of the 35 conditions can achieve within the 200-epoch training budget; unreachable cases are reported as lower bounds.

Low recall imposes a consistent compute penalty. Fig. 3 shows a strong recall gradient: at β=0.3 (rightmost column), all conditions need 2.08 to over 3.91× the baseline compute, compared to 1.00–3.29× at β=1.0. At α=0.7, β=0.3, the model reaches the threshold at 2.15×, slower than most β=1.0 conditions. Precision also increases compute cost: at fixed β=0.9, reducing α from 1.0 to 0.3 raises cost from 1.41× to 2.35×, with the highest cost at α=0.5 (3.62×). We examine whether supervised finetuning on clean data can recover from this damage in §6. Both axes affect training efficiency, though recall shows a more consistent gradient across all precision levels.

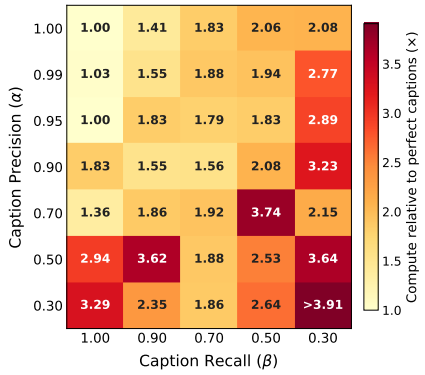


Figure 3: Compute to reach a validation loss of 0.035 (evaluated with ground-truth captions), shown relative to the ground-truth baseline (α=1, β=1, 66×10^3 PFLOPs $= 1.0\times$). Models are trained on corrupted captions at varying precision (α) and recall (β); higher values mean more compute to reach the same loss. Low recall (β=0.3) needs 2–4$\times$ more compute than ground-truth captions, and at α=0.3, β=0.3 the threshold is never reached within the 200-epoch budget ($> 3.9\times$).

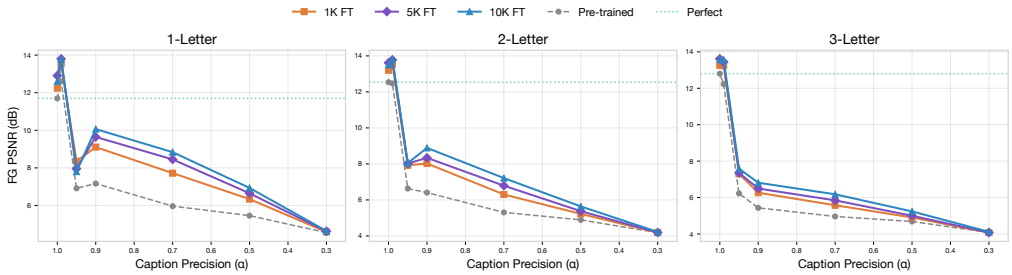


Figure 4: Finetuning recovery for FG PSNR at β=1.0, CFG = 1.0. Raw FG PSNR vs. caption precision α. Finetuned models exceed the clean-caption baseline at α=0.99, partially recover at tested precision levels in $[0.7, 0.95]$, and show only very limited improvement for $\alpha \leq 0.5$ within the tested finetuning budget. 10K outperforms 5K, which outperforms 1K, with diminishing returns.

Takeaways. Caption quality directly affects training efficiency. Corrupted captions require more compute to reach a given validation loss. In severe cases, the threshold is not reached within the training budget. Reducing precision or recall increases training cost, with low recall showing a particularly consistent penalty across all precision levels. For practitioners with a fixed compute budget, investing in higher-quality captions yields a larger return than training longer on noisy data.

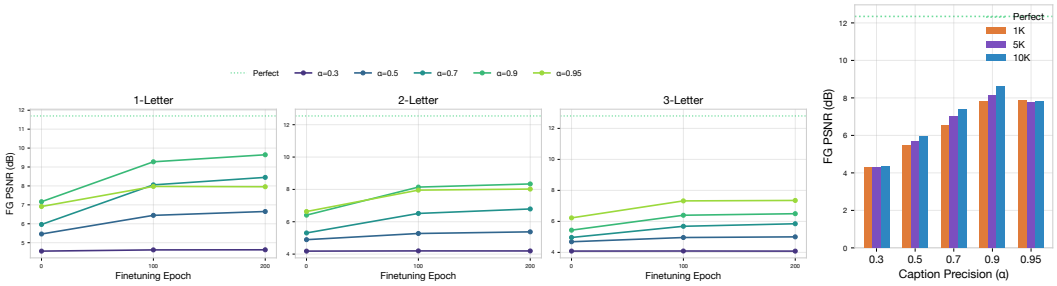
6 Can Finetuning Recover from Caption Corruption?

Experimental setup. The previous sections showed that caption corruption during pretraining degrades generation quality. We now ask whether finetuning on a small set of clean captions can recover this loss. We finetune each corrupted model on 1K, 5K, or 10K clean examples balanced across complexity levels, using ground-truth captions and a reduced learning rate. Fig. 4 and Fig. 6 show pixel-quality recovery for precision and recall corruption, respectively.

Mild corruption is partially recoverable. At tested precision levels in $[0.7, 0.95]$, finetuned models recover a meaningful fraction of lost FG PSNR but remain below the baseline. At α=0.9, finetuning with 5K samples raises 1L FG PSNR from 7.2 to 9.6 dB, a 55% recovery toward the 11.7 dB baseline. At α=0.7, 1L recovers 43%, 2L recovers 21%, and 3L only 11%.

Severe corruption is difficult to reverse. At $\alpha \leq 0.5$, finetuning yields only very limited improvement within the tested budget. At α=0.3, FG PSNR barely improves: 1L stays at 4.6 dB versus 4.6 dB pretrained, essentially unchanged; see Fig. 4. These results suggest that incorrect associations learned during pretraining are difficult to correct, even with 10K clean examples.

Recovery is harder for compositional scenes. A clear complexity gradient emerges: 1L scenes consistently achieve the highest recovery, followed by 2L, then 3L; see Fig. 5a. At α=0.7, 1L reaches 43% recovery while 3L reaches only 11% with 5K samples. Multi-object scenes require learning inter-object relationships like relative positions and independent motions, which are harder to reacquire from limited data.



(a) Recovery over finetuning epochs (b) 1,5,10K Finetuning

Figure 5: (a) FG PSNR recovery over finetuning epochs. Most gains occur within the first 100 epochs; severe corruption (α=0.3) shows very limited recovery. (b) 10K finetuning outperforms 5K and 1K for FG PSNR, with the largest margin at moderate corruption.

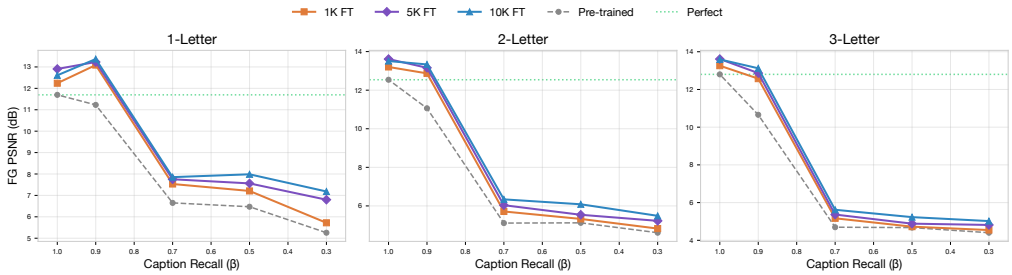


Figure 6: Finetuning recovery for FG PSNR vs. caption recall β at fixed α=1.0, CFG = 1.0. Mild recall corruption (β=0.9) is fully recoverable, with finetuned FG PSNR exceeding the ground-truth baseline. At lower recall ($\beta \leq 0.7$), finetuned models remain well below the ground-truth baseline.

More data helps, but with diminishing returns. Finetuning with more data consistently helps, as shown in Fig. 5b: 10K outperforms 5K, which outperforms 1K, with the largest gains in the moderate corruption regime at $\alpha \in [0.7, 0.95]$. For severe corruption at $\alpha \leq 0.5$, 10K samples still yield only limited recovery within the tested budget.

Recall corruption is far more recoverable than precision. Comparing Fig. 6 with Fig. 4 shows that recall corruption behaves very differently from precision. At β=0.9, finetuning fully recovers FG PSNR, matching or even exceeding the ground-truth baseline. At lower recall ($\beta \leq 0.7$), finetuning yields limited gains and does not close the gap to the ground-truth baseline. This mirrors the pretraining finding in §5.1: models trained on incomplete but accurate captions learn no false associations, so clean data fills in the gaps; hallucinated associations must first be unlearned.

Takeaways. Finetuning can fully recover from very mild precision corruption, but recovery remains partial at tested precision levels in $[0.7, 0.95]$ and limited under severe corruption ($\alpha \leq 0.5$) within the tested budget. Most recovery occurs within the first 100 epochs (Fig. 5a). The pretraining asymmetry persists: omission (low β) is much easier to repair than hallucination (low α), since the model has not learned wrong associations.

7 Conclusion

We introduce **Moving Alphabet**, a controlled testbed for studying how training data affects text-to-video generation. Our experiments yield three findings: balanced and diverse content complexity and clip duration are important for generalization; caption quality is crucial for both model performance and training efficiency; and recovery through clean-data fine-tuning depends on the severity and type of caption corruption. These results collectively recommend that practitioners prioritize balanced data curation and high-quality captions at the pre-training stage. This work also opens up a few interesting future directions, including how to measure diversity in real-world scenarios, and conditioning on non-textual modalities such as spatial layouts (Reve Team, 2026). We hope this work demonstrates the importance of training data in text-to-video generation and encourages community efforts to improve data curation and quality.

Ethics statement

This study uses procedurally generated videos of moving letters and does not involve human subjects or personal data. Its purpose is to improve understanding of training-data curation for text-to-video models. Although video-generation systems can carry broader risks of misuse, the experiments here study controlled data and training effects rather than introducing a new deployment capability.

Reproducibility statement

Section 3.1 describes the procedural data construction and controlled factors, Section 3.2 specifies the model and baseline training setup, and Section 3.3 defines the evaluation protocol. The appendix reports the complete attribute space, additional training and metric details, convergence evidence, and expanded experimental results. The supplementary material will include any additional configurations and artifacts released for reproducing the experiments.

AI use statement

Generative AI tools were used solely to assist with grammar and language editing. They were not used to generate data, formulate hypotheses, design experiments, implement methods, analyze data, or interpret results. All AI-assisted edits were reviewed by the authors. The authors take responsibility for the final content of this work.

References

Zeyuan Allen-Zhu and Yuanzhi Li. Physics of language models: Part 3.1, knowledge storage and extraction. In *Proc. ICML*, 2024. 3

Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proc. ICML*, 2009. 2

James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. 2023. OpenAI technical report. 2

Andreas Blattmann, Tim Dockhorn, Sumith Kulal, Daniel Mendelevitch, Maciej Kilian, Dominik Lorber, Yam Levi, Zion English, Vikram Voleti, Adam Letts, et al. Stable video diffusion: Scaling latent video diffusion models to large datasets. *arXiv preprint arXiv:2311.15127*, 2023. 2

Haoxin Chen, Yong Zhang, Xiaodong Cun, Menghan Xia, Xintao Wang, Chao Weng, and Ying Shan. VideoCrafter2: Overcoming data limitations for high-quality video diffusion models. *arXiv preprint arXiv:2401.09047*, 2024a. 2

Lin Chen, Xilin Wei, Jinsong Li, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Zehui Chen, Haodong Duan, Bin Lin, Zhenyu Tang, Li Yuan, Yu Qiao, Dahua Lin, Feng Zhao, and Jiaqi Wang. ShareGPT4Video: Improving video understanding and generation with better captions. *arXiv preprint arXiv:2406.04325*, 2024b. 2

Tsai-Shien Chen, Aliaksandr Siarohin, Willi Menapace, Ekaterina Deyneka, Hsiang-wei Chao, Byung Eun Jeon, Yuwei Fang, Hsin-Ying Lee, Jian Ren, Ming-Hsuan Yang, and Sergey Tulyakov. Panda-70M: Captioning 70M videos with multiple cross-modality teachers. *arXiv preprint arXiv:2402.19479*, 2024c. 2

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*, 2025. 1

Ronen Eldan and Yuanzhi Li. TinyStories: How small can language models be and still speak coherent English? *arXiv preprint arXiv:2305.07759*, 2023. 3

Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, Dustin Podell, Tim Dockhorn, Zion English, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis. In *Proc. ICML*, 2024. 3, 16

Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. DataComp: In search of the next generation of multimodal datasets. In *Proc. NeurIPS*, 2023. 2

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800GB dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020. 2

Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*, 2022. 15, 18

Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *Proc. NeurIPS*, 2020. 1

Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, and Tim Salimans. Imagen video: High definition video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022. 1

Kaiyi Huang, Kaiyue Sun, Enze Xie, Zhenguo Li, and Xihui Liu. T2I-CompBench: A comprehensive benchmark for open-world compositional text-to-image generation. *arXiv preprint arXiv:2307.06350*, 2023. 3

Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C Lawrence Zitnick, and Ross Girshick. CLEVR: A diagnostic dataset for compositional language and elementary visual reasoning. In *Proc. CVPR*, 2017. 3

Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child, Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language models. *arXiv preprint arXiv:2001.08361*, 2020. 1, 3

Hyunjik Kim and Andriy Mnih. Disentangling by factorising. In *Proc. ICML*, 2018. 3

Weijie Kong, Qi Tian, Zijian Zhang, Rox Min, Zuozhuo Dai, Jin Zhou, Jiangfeng Xu, Bohao Pang, Hao Liu, et al. HunyuanVideo: A systematic framework for large video generative models. *arXiv preprint arXiv:2412.03603*, 2024. 3, 16

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. Flow matching for generative modeling. In *Proc. ICLR*, 2023. 3, 16

Haotian Liu, Chunyuan Li, Yuheng Li, Bo Li, Yuanhan Zhang, Sheng Shen, and Yong Jae Lee. LLaVA-NeXT: Improved reasoning, OCR, and world knowledge. 2024. Blog post. 2

Loic Matthey, Irina Higgins, Demis Hassabis, and Alexander Lerchner. dSprites: Disentanglement testing sprites dataset. 2017. https://github.com/deepmind/dsprites-dataset/. 3

Niklas Muennighoff, Alexander M Rush, Boaz Barak, Teven Le Scao, Aleksandra Piktus, Nouamane Tazi, Sampo Pyysalo, Thomas Wolf, and Colin Raffel. Scaling data-constrained language models. *arXiv preprint arXiv:2305.16264*, 2023. 3

OpenAI. Sora: Creating video from text. 2024. Technical report. 1, 3

William Peebles and Saining Xie. Scalable diffusion models with transformers. In *Proc. ICCV*, 2023. 1

Adam Polyak, Amit Zohar, Andrew Brown, Andros Tjandra, Animesh Sinha, Ann Lee, Apoorv Vyas, Bowen Shi, Chih-Yao Ma, Ching-Yao Chuang, et al. Movie Gen: A cast of media foundation models. *arXiv preprint arXiv:2410.13720*, 2024. 1, 3, 16

Thomas Porter and Tom Duff. Compositing digital images. In *Proc. SIGGRAPH*, 1984. 14

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *Proc. ICML*, 2021. 3, 16

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *JMLR*, 2020. 3, 16

Reve Team. The layout bet, June 2026. URL https://reve.com. Accessed: 2026-06-03. 9

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proc. CVPR*, 2022. 1

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. LAION-5B: An open large-scale dataset for training next generation image-text models. In *Proc. NeurIPS*, 2022. 2

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, Devi Parikh, Sonal Gupta, and Yaniv Taigman. Make-A-Video: Text-to-video generation without text-video data. *arXiv preprint arXiv:2209.14792*, 2022. 1

Kimi Team, Angang Du, Bohong Yin, Bowei Xing, Bowen Qu, Bowen Wang, Cheng Chen, Chenlin Zhang, Chenzhuang Du, Chu Wei, et al. Kimi-VL technical report. *arXiv preprint arXiv:2504.07491*, 2025. 1

Thomas Unterthiner, Sjoerd van Steenkiste, Karol Kurach, Raphaël Marinier, Marcin Michalski, and Sylvain Gelly. FVD: A new metric for video generation. 2019. 4

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Proc. NeurIPS*, 2017. 1

Yi Wang, Yinan He, Yizhuo Li, Kunchang Li, Jiashuo Yu, Xin Ma, Xinhao Li, Guo Chen, Xinyuan Chen, Yaohui Wang, et al. InternVid: A large-scale video-text dataset for multimodal understanding and generation. *arXiv preprint arXiv:2307.06942*, 2023. 2

Sang Michael Xie, Shibani Santurkar, Tengyu Ma, and Percy Liang. DoReMi: Optimizing data mixtures speeds up language model pretraining. In *Proc. NeurIPS*, 2023. 2

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. 1

Zhuoyi Yang, Jiayan Teng, Wendi Zheng, Ming Ding, Shiyu Huang, Jiazheng Xu, Yuanming Yang, Wenyi Hong, Xiaohan Zhang, Guanyu Feng, et al. CogVideoX: Text-to-video diffusion models with an expert transformer. *arXiv preprint arXiv:2408.06072*, 2024. 3, 16

Jiasheng Ye, Peiju Liu, Tianxiang Sun, Yunhua Zhou, Jun Zhan, and Xipeng Qiu. Data mixing laws: Optimizing data mixtures by predicting language modeling performance. *arXiv preprint arXiv:2403.16952*, 2024. 2

Jihai Zhang, Tianle Li, Linjie Li, Zhengyuan Yang, and Yu Cheng. Are unified vision-language models necessary: Generalization across understanding and generation. *arXiv preprint arXiv:2505.23043*, 2025. 3

APPENDIX

A EVALUATION CALIBRATION AND METRIC DETAILS

This appendix validates the evaluation metrics introduced in §3.3 and specifies their implementation.

Table 3: **(a)** Metric calibration: attribute accuracy evaluated on raw ground-truth videos, confirming that our classifiers are near-perfect on uncompressed data. **(b)** VAE reconstruction ceiling: metrics after an encode→decode round-trip, establishing upper bounds for generated video evaluation.

(A) METRIC CALIBRATION (RAW)

	1L	2L	3L
Color ↑	0.995	0.975	0.956
Dir ↑	0.996	0.992	0.990
Size ↑	1.000	1.000	1.000
Speed ↑	0.996	0.995	0.994

(B) VAE RECONSTRUCTION CEILING

	1L	2L	3L
Color ↑	0.987	0.967	0.946
Dir ↑	0.996	0.993	0.990
Size ↑	1.000	1.000	1.000
Speed ↑	0.998	0.996	0.995
FG MSE ($\times 10^3$) ↓	5.6	4.7	4.6
FG PSNR (dB) ↑	22.5	23.2	23.4
PSNR (dB) ↑	41.1	38.1	36.2

Our per-attribute scores are produced by automated classifiers, which we apply to generated videos to measure what the model actually produced. A generated video carries no attribute label of its own, which must be measured from the pixels, and an unreliable classifier would make a low score reflect the metric rather than the model. We therefore calibrate the classifiers on ground-truth videos with known attributes, thereby demonstrating the classifiers' correctness. As shown in Tab. 3(a), the classifiers recover the true attributes near-perfectly, the lowest being color at 0.956 on 3L scenes. These calibration scores provide a reference for interpreting the scores on generated videos, including residual measurement error. The residual drop comes from the inter-letter occlusion described in Appendix B.1: where letters overlap, the occluding letter's pixels enter the occluded one's ROI and bias the median color. The effect grows with the number of letters, with the color ceiling decreasing from 0.995 (1L) to 0.975 (2L) and 0.956 (3L).

Next, we also consider the effect of VAE on the metrics. Every generated video is decoded by the same frozen, lossy VAE, which caps the pixel quality any model can reach, so FG PSNR is only meaningful relative to that cap. Tab. 3(b) measures every metric on the VAE encode–decode of the ground-truth videos, establishing this ceiling for FG PSNR of 22–23 dB and for FG MSE of 4.6–5.6×10^{-3}. Our best models reach FG PSNR of 13–14 dB, 9–10 dB below this ceiling; hence, our experiments are therefore not bottlenecked by VAE quality, and the differences we observe across settings reflect the quality of the diffusion model itself.

A.1 EVALUATION METRIC DETAILS

In all cases we first obtain the letter pixels with a simple foreground mask: a pixel is foreground if its brightest RGB channel exceeds $40/255$, which cleanly separates the letters from the near-black background. For 1L videos this mask isolates the single letter directly. For multi-letter videos, we additionally localize each letter by its ground-truth trajectory: at every frame, we use the trajectory metadata described in Appendix B.1 to place a rectangular region of interest (ROI) at the letter's known position and size it to its rendered glyph. We intersect this ROI with the foreground mask and read that letter's attributes only from the pixels inside. Separating letters by location rather than appearance reduces cross-letter interference, although overlapping ROIs remain subject to the occlusion effects noted above. Because each ROI is anchored to a specific ground-truth letter, each prediction is aligned to its corresponding letter without any matching step, and we average per-attribute accuracy over all letters across all videos at each complexity level. From these per-letter pixels, we measure:

- *Color accuracy* (A_{color}): We extract the brightest foreground pixels with intensity greater than 200, where anti-aliasing effects are negligible, compute their median RGB value, and

assign the nearest ground-truth color based on Euclidean distance in the RGB space. The predicted color is correct if it matches the ground-truth color.

- *Size accuracy* (A_{size}): we measure the foreground bounding box's maximum dimension (median over the first 5 frames) and bin it into small ($<$44 px), medium (44–68 px), or large ($\geq$68 px), corresponding to GT render sizes of 32, 56, and 80 px. Correct if the bin matches GT.
- *Direction accuracy* (A_{dir}): We compute the displacement vector between the first and last detected centroids, convert it to an angle using atan2, and assign it to one of four 90° bins centered on right, down, left, or up. A prediction is correct if it matches the GT direction.
- *Speed correlation* (ρ_{speed}): we compute the median per-frame centroid displacement (in pixels) as the predicted speed, then report the Pearson correlation between predicted and GT speeds across all videos in the evaluation set.

B Attribute Specifications

Tab. 4 lists all attributes used in the Moving Alphabet dataset and their possible values.

Table 4: Moving Alphabet attributes and their value sets. Appearance attributes and initial motion parameters are sampled uniformly, with distinct colors within each scene. Caption direction is derived from the resulting trajectory.

Category	Attribute	Values
Appearance	Letter identity	52 classes (26 uppercase A–Z, 26 lowercase a–z)
	Font	8 typefaces (serif, sans-serif, monospace variants)
	Color	10 options (red, orange, yellow, green, cyan, blue, purple, magenta, white, light green)
	Size	3 levels: small (32 px), medium (56 px), large (80 px)
	Rotation	8 angles (0°, 45°, 90°, 135°, 180°, 225°, 270°, 315°)
Motion	Direction	Net-displacement angle 0–360° (binned to 4 cardinal directions for evaluation)
	Speed	Continuous range 3–10 pixels per frame (sampled uniformly)

B.1 Data Generation Details

Content complexity. Each video contains one to three letters, which we refer to as 1-letter (1L), 2-letter (2L), and 3-letter (3L) scenes. Each letter is described by seven attributes covering both appearance and motion:

- **Appearance**: letter identity is sampled from 52 classes (26 uppercase and 26 lowercase letters); font is sampled from 8 typefaces, color from 10 options, size from 3 levels (small, medium, and large), and rotation from 8 angles. See Appendix B for attribute specifications.
- **Motion**: each letter is initialized at a random position with a speed sampled uniformly in $[3, 10]$ pixels per frame and a heading sampled uniformly in $[0^\circ, 360^\circ)$. The letter moves at constant speed and reflects off the canvas boundaries, with every bounce recorded in the metadata. We summarize the motion by a single direction attribute, the angle of the net start-to-end displacement; since it is measured from the realized trajectory, it already accounts for any bounces rather than reflecting the initial heading.

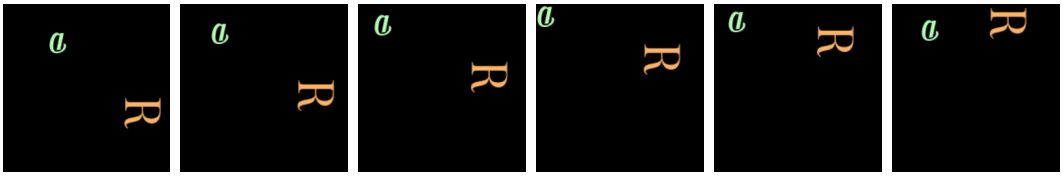
Colors are sampled without replacement within each scene; the remaining appearance attributes and initial motion parameters are sampled independently. Increasing the number of letters introduces overlapping trajectories and shared canvas space; where letters overlap, they are alpha-composited (Porter & Duff, 1984) in a fixed index order, so a higher-indexed letter occludes a lower-indexed one.

Temporal complexity. Video length is independently configurable to roughly 2, 4, or 8 seconds, corresponding to 17, 33, or 65 frames[1] at 8 fps and 256×256 resolution. Longer videos typically contain more boundary bounces and exhibit more complex trajectory patterns.

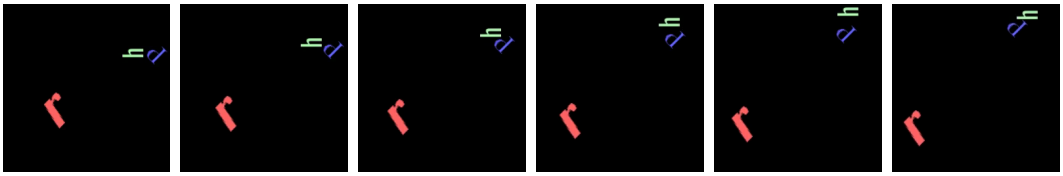
[1]The causal temporal VAE compresses frames $4\times$ in time and encodes the first frame independently, so it expects $4n{+}1$ input frames; we therefore use 17, 33, and 65 frames rather than 16, 32, and 64.



(a) **1-letter.** "A medium light green lowercase x in handwritten font, rotated 315 degrees. It starts at (0.29, 0.19) and moves at 291.4 degrees at speed 4.3, ending at (0.34, 0.07). It bounces off the top wall."



(b) **2-letter.** "A scene with 2 letters. Letter 1: A medium orange uppercase R in serif font, rotated 90 degrees. It starts at (0.73, 0.56) and moves at 254.3 degrees at speed 9.5, ending at (0.57, 0.01). It bounces off the top wall. Letter 2: A small light green uppercase D in decorative font, rotated 180 degrees. It starts at (0.28, 0.17) and moves at 216.8 degrees at speed 8.9, ending at (0.21, 0.11). It bounces off the left and top walls."



(c) **3-letter.** "A scene with 3 letters. Letter 1: A medium red lowercase r in decorative font, rotated 315 degrees. It starts at (0.25, 0.53) and moves at 149.0 degrees at speed 3.5, ending at (0.06, 0.64). Letter 2: A small light green lowercase h in condensed font, rotated 270 degrees. It starts at (0.71, 0.27) and moves at 278.4 degrees at speed 5.2, ending at (0.74, 0.06). It bounces off the top wall. Letter 3: A small blue uppercase P in serif font, rotated 225 degrees. It starts at (0.86, 0.25) and moves at 222.0 degrees at speed 4.5, ending at (0.67, 0.08). It bounces off the right wall."

Figure 7: Example videos from the Moving Alphabet dataset at three complexity levels, one per row. Frames are shown left to right at t=0, 3, 6, 9, 12, 15 of the 17-frame clips (256×256, 8 fps). Each row's ground-truth caption below describes every letter's appearance and motion.

Captions. Each video is paired with a templated natural-language caption generated from its metadata. The captions describe all appearance attributes, start and end positions using normalized coordinates, movement direction as an exact angle, speed, and any boundary bounces. See Fig. 7 for examples across complexity levels.

B.2 Caption Examples

Fig. 7 extends the example in Fig. 1 to scenes containing one, two, and three letters, pairing each video with its ground-truth caption. These captions are generated from scene metadata and describe each letter's appearance and motion, illustrating how the same template handles increasing scene complexity.

C Training Details

Pretraining uses AdamW with a learning rate of 5×10^{-4}, linearly warmed up from 0 over the first 500 steps and held constant thereafter. We use a per-GPU batch size of 24, bf16 mixed precision, and gradient clipping with a maximum norm of 1.0. During training, we apply 10% text dropout to enable classifier-free guidance during inference (Ho & Salimans, 2022).

C.1 Model Architecture

Architecture. We adopt a standard MMDiT architecture (Esser et al., 2024) trained with flow matching (Lipman et al., 2023). The model contains 800M parameters, consisting of 36 transformer blocks, with a hidden dimension of 1024 and 16 attention heads. For text conditioning, we use two frozen text encoders: T5-XXL (Raffel et al., 2020) with 4096-dimensional token embeddings and CLIP ViT-L/14 (Radford et al., 2021) with 768-dimensional pooled embeddings. The projected token embeddings are jointly attended to with video tokens, while the pooled CLIP embedding contributes to the conditioning vector that modulates the transformer blocks. Video latents are obtained using a pretrained causal 3D VAE that encodes the first frame independently and applies $8\times$ spatial and $4\times$ temporal compression, followed by spatial patchification with a 2×2 patch size and temporal patch size as 1. We adopt this standard recipe (Kong et al., 2024; Yang et al., 2024; Polyak et al., 2024) so our findings transfer to deployed systems, at a moderate 800M scale that keeps hundreds of controlled experiments affordable.

Baseline. Pretraining starts from scratch on $8\times$A100 GPUs. The base configuration uses 150K videos with 17 frames at 8 fps and 256×256 resolution, converging in approximately two days. The baseline metrics plateau by epoch 100 (Fig. 8); we use this checkpoint for baseline evaluation. Caption-corruption and finetuning protocols are given separately in Appendix D.1 and Appendix D.3. During training, we randomly drop the caption with a probability 0.1 to enable classifier-free guidance at inference. Full training hyperparameters are in Appendix C.

D Additional Experimental Details

D.1 Caption Corruption: Grid and Training Protocol

When α=0.7, 30% of claims are replaced with random incorrect values, simulating hallucination. When β=0.3, only 30% of attributes appear in the caption, simulating omission.

We train separate models on a grid of 7 precision values, $\alpha \in \{1.0, 0.99, 0.95, 0.9, 0.7, 0.5, 0.3\}$, and 5 recall values, $\beta \in \{1.0, 0.9, 0.7, 0.5, 0.3\}$, for a total of 35 conditions. Training the full grid requires 35 separate models, one per condition. To keep this tractable, we train each on a 50K subset (2-second, equal-complexity mix) rather than the full 150K dataset, using the same 200-epoch budget for every condition. Threshold attainment under this budget is analyzed separately in §5.2. The attribute-corruption procedure is described in §5.1.

D.2 Precision Advantage: Iso-budget Comparison

The precision-advantage statistic in Fig. 2c is defined as follows.

Precision advantage per attribute is an iso-budget swap. Let $M(\alpha, \beta)$ be the attribute metric: accuracy for color, direction, and size, or correlation for speed. For every pair of corruption levels $x{>}y$ shared by both axes, we compare giving the higher level to precision, $M(\alpha{=}x, \beta{=}y)$, against giving it to recall, $M(\alpha{=}y, \beta{=}x)$; the bar is the mean of $M(\alpha{=}x, \beta{=}y){-}M(\alpha{=}y, \beta{=}x)$ over all such pairs. It is positive for all four attributes, so prioritizing precision rather than recall improves every attribute metric on average across the matched budgets, confirming that precision is the primary driver of attribute controllability.

D.3 Finetuning Protocol

We take each caption-corrupted model from §5.1 and finetune it for 200 epochs with ground-truth captions at a reduced learning rate of 10^{-4}, down from 5×10^{-4}, using 1K, 5K, or 10K stratified samples balanced across complexity levels. We evaluate at finetuning epochs 100 and 200. Fig. 4 shows FG PSNR at β=1.0 across α values.

Fig. 6 shows recovery across recall levels; both figures accompany the main analysis in §6.

D.4 Compute Efficiency: Validation Protocol

Experimental setup. We train the same 35 $\alpha \times \beta$ conditions from §5.1, but now evaluate validation loss every epoch using *ground-truth* captions, measuring how well the model has learned the true data distribution regardless of its training captions. We report the compute needed to first reach a validation loss of 0.035, normalized to the ground-truth baseline (66×10^3 PFLOPs). We select this threshold by sweeping in increments of 0.005: at 0.030 most conditions fail to converge within the training budget, making 0.035 the boundary at which the majority of settings can be compared.

E Training Convergence

Fig. 8 shows three pixel-quality metrics (FG PSNR, BG PSNR, and FG MSE) as a function of training step for the Mix Equal condition, evaluated across all three complexity levels. All metrics plateau by step 60K–70K, which corresponds to roughly 100 epochs with 150K training samples. We select epoch 100 for the baseline and data-distribution experiments; the caption-corruption and finetuning schedules are specified in Appendix D.

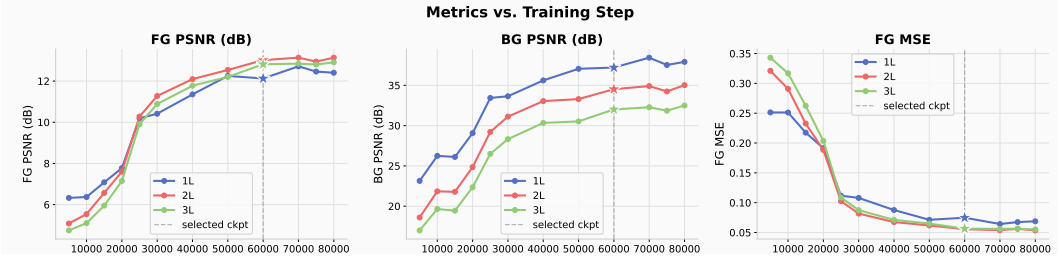


Figure 8: Pixel-quality metrics (FG PSNR, BG PSNR, and FG MSE) vs. training step for the Mix Equal condition. All metrics plateau by step 60K–70K. The dashed line marks the selected checkpoint.

F Precision Advantage: Pairwise Comparisons

Fig. 9 provides the per-pair detail behind the precision advantage in Fig. 2c: for every pair of corruption levels, we compare assigning the higher level to precision against assigning it to recall. Precision-favored allocations win for the majority of pairs across all four attributes, with the largest margins for color and size.

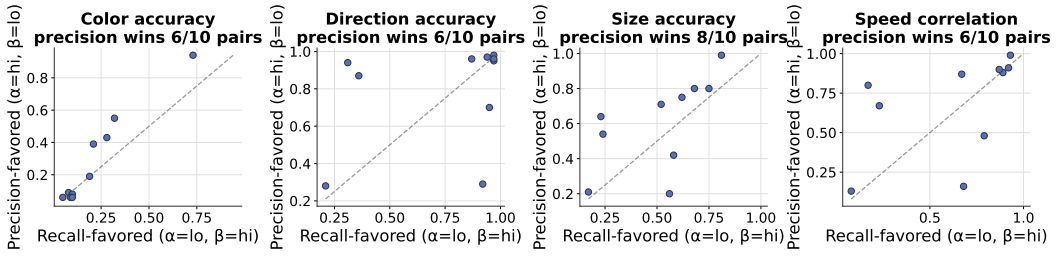
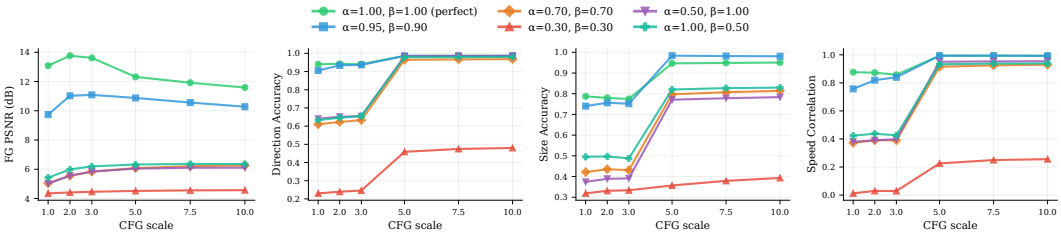


Figure 9: Per-pair detail behind the precision advantage in Fig. 2c. For every pair of corruption levels, we plot the metric when the higher level is assigned to precision (y, α=hi, β=lo) against the recall-favored assignment (x, α=lo, β=hi); points above the diagonal favor precision. Precision-favored allocations win in the majority of pairs for every attribute, and the wins are large for color and size; direction and speed are noisier (single run, CFG = 1.0).

G How Does CFG Interact with Caption Quality?

Experimental setup. Classifier-free guidance (CFG) (Ho & Salimans, 2022) amplifies the conditional signal at inference time and is widely used in practice. We study how CFG interacts with training data quality by evaluating every model from the caption corruption grid in §5.1 at six CFG scales from 1.0 to 10.0, keeping all other inference settings fixed at 50 steps and 2000 samples per complexity level. Fig. 10 shows four metrics, FG PSNR, direction, size, and speed, averaged across complexity levels for six representative conditions spanning ground-truth to severe corruption.



(a) FG PSNR (b) Direction (c) Size (d) Speed

Figure 10: Effect of CFG scale on generation quality for models trained with varying caption corruption. Each line is one corruption condition averaged across 1L/2L/3L evaluation. CFG affects attribute faithfulness and pixel quality differently, with the trade-off depending on caption corruption.

Pixel fidelity and attribute agreement favor different CFG scales. For the model trained on ground-truth captions, FG PSNR peaks at CFG = 2 at approximately 13.8 dB, then declines to approximately 11.6 dB at CFG = 10 (Fig. 10a). The mild corruption condition (α=0.95, β=0.9) shows a similar peak at CFG = 2–3. For these models, higher CFG improves direction, size, and speed metrics even as FG PSNR declines (Fig. 10). The preferred guidance scale therefore depends on whether pixel fidelity or attribute agreement is the priority.

CFG improves attribute agreement but leaves a large pixel-quality gap. For moderately corrupted models (α=0.7, β=0.7), raising CFG from 1 to 5 improves direction accuracy from approximately 0.61 to 0.96, and size accuracy from approximately 0.42 to 0.80 (Fig. 10b, Fig. 10c). However, FG PSNR increases only from approximately 5.0 to 6.3 dB over the full CFG sweep and remains well below the model trained on ground-truth captions (Fig. 10a). Thus, substantial recovery of individual attributes does not restore overall pixel fidelity within the tested CFG range.

Severe corruption limits recovery under CFG. At α=0.3, β=0.3, FG PSNR varies by less than 0.3 dB across the full CFG range (Fig. 10a). Direction accuracy nevertheless rises from approximately 0.23 at CFG = 1 to 0.48 at CFG = 10 (Fig. 10b). Size and speed also improve, but all three attribute metrics remain well below those of the model trained on ground-truth captions (Fig. 10). Stronger guidance therefore provides partial attribute recovery even under severe corruption, while pixel quality remains low.

Precision and recall interact differently with CFG. Comparing the α-only (α=0.5, β=1.0) and β-only (α=1.0, β=0.5) conditions isolates the two caption failure modes. Both have similar FG PSNR (approximately 5.0 vs. 5.4 dB at CFG = 1), but their attribute profiles differ under CFG. At higher CFG, the β-only model has slightly higher FG PSNR and size accuracy, whereas the α-only model has slightly higher direction accuracy and speed correlation (Fig. 10). The relative benefit of guidance for precision and recall corruption therefore depends on the metric.

Takeaways. Within the tested CFG range, guidance improves attribute agreement but does not close the pixel-quality gap between corrupted- and clean-caption models. CFG = 2–3 favors FG PSNR for clean or mildly corrupted captions (Fig. 10a), while higher guidance benefits attribute metrics. The preferred CFG scale and the relative effects of precision and recall corruption depend on the metric. These results favor improving pretraining captions over relying on CFG for full recovery.

H What Happens When Attributes Are Never Mentioned?

Real-world captions rarely describe every visual attribute. Even detailed captions cannot cover all aspects of a video: a caption might describe the main subjects and actions, but omit details like the exact colors of background objects, the precise speed of each motion, or the size relationships between elements. Since achieving perfect recall is practically impossible, attributes are systematically omitted from training captions. When an attribute is systematically absent from all training captions, the model must decide what to generate. We study whether models learn to produce diverse values or collapse to a default. We examine this question in two complementary settings: first with other attributes varying as usual, then in a single-attribute control that compares conditional and unconditional models.

H.1 Systematic Omission from Otherwise-Complete Captions

Experimental setup. We drop one attribute from otherwise-complete captions: for each target attribute (size, direction, or color), we train a model on captions that are identical to the ground-truth captions except that every mention of the target attribute is removed, while all other attributes remain fully specified and vary across videos. We repeat each condition across 4 random seeds and evaluate at CFG = 1.0.

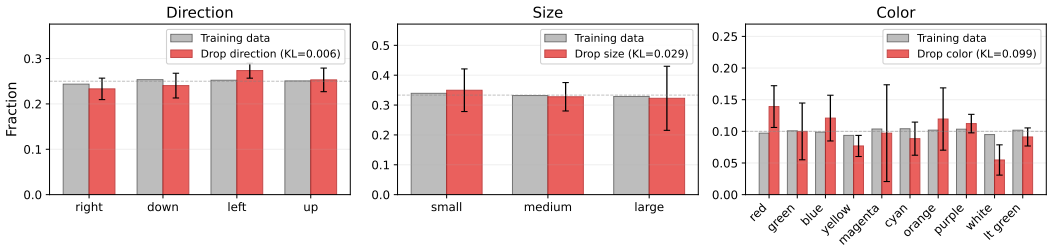


Figure 11: Drop-one marginal distributions: GT training data vs. generated when one attribute is dropped from otherwise-complete captions. Bars show mean over 4 seeds; error bars show ± 1 std. Direction stays near-uniform; color and size develop seed-dependent biases.

Size and color develop biases; direction is resilient. When one attribute is dropped from otherwise-complete captions (Fig. 11), size develops a bias (mean KL = 0.029), though with high seed-dependent variance (0.003–0.056). Color has a larger mean bias, with each seed favoring different colors (mean KL = 0.099). Direction, by contrast, stays near-uniform across all seeds (mean KL = 0.006). The reason is that direction is redundantly specified by other fields: when the direction phrase is dropped, the start and end positions (and any bounce events) remain in the caption, and together they largely determine the net direction even without stating it explicitly.

H.2 Single-Attribute Control

Experimental setup. Unlike the drop-one setting above, the single-attribute setting varies only the target attribute while holding the remaining appearance and initial motion parameters fixed. Conditional-drop captions omit the target attribute; unconditional models receive no text conditioning.

Conditional omission produces stronger size bias. Here, the collapse is far more extreme. Tab. 5 shows that conditional drop models have 13.7× higher KL divergence than unconditional models for size, and Fig. 13 reveals the mechanism: each seed collapses to a different dominant size, while all unconditional models maintain roughly uniform distributions. The text encoder's null representation for size acts as a strong but arbitrary default that varies with random initialization. For color (1.6×) and direction (1.9×), the gap between conditional drop and unconditional is small, and both conditions show similar bias patterns (Fig. 12).

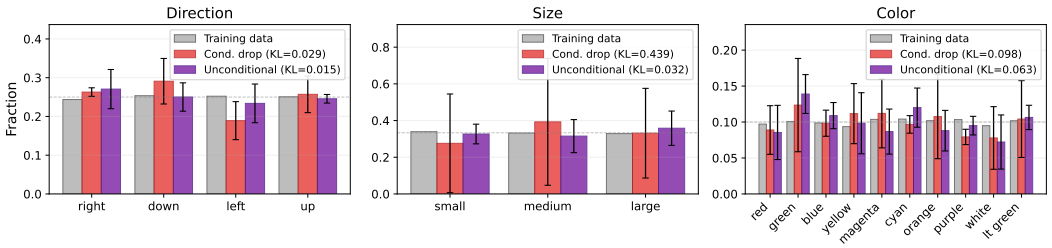


Figure 12: Single-attribute setting: GT vs. conditional drop vs. unconditional for each attribute. Bars show mean over 4 seeds; error bars show ± 1 std. The conditional drop model develops large, high-variance biases for size, while the unconditional model stays near-uniform.

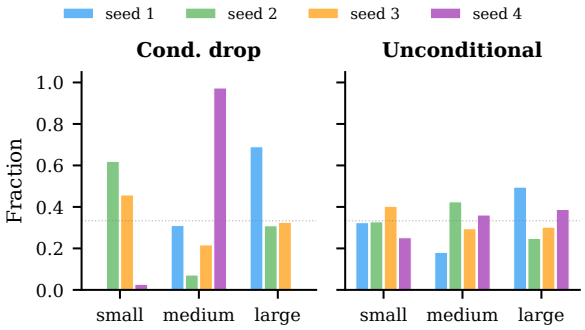


Figure 13: Single-attribute setting: per-seed size distributions for conditional drop vs. unconditional models. Each conditional drop seed converges to a different dominant size, while unconditional models preserve diversity.

Table 5: KL divergence from the training distribution when an attribute is never mentioned in captions. **Cond. drop**: text-conditioned model with the attribute omitted from all captions. **Unconditional**: model trained without any text conditioning. Values are mean $\pm$ std over 4 random seeds (CFG = 1.0).

Attribute	Cond. drop	Unconditional	Ratio
Size	0.44 ± 0.35	0.03 ± 0.03	$13.7\times$
Color	0.10 ± 0.04	0.06 ± 0.04	$1.6\times$
Direction	0.03 ± 0.03	0.01 ± 0.01	$1.9\times$

Takeaways. Whether a missing attribute is still generated correctly depends on whether it remains recoverable from the rest of the caption. In the drop-one setting, direction is resilient because it stays pinned down by the start and end positions left in the caption when the direction phrase is dropped; size and color have no such redundant cue and instead develop biases, with each model drifting toward a default value whose choice varies unpredictably across seeds. For size, the single-attribute control also shows that omitting an attribute from conditional captions can produce stronger bias than training without text conditioning. This matters for real data. Web captions rarely mention attributes like object size ("a cat on a table" does not say whether the cat is large or small). Omitting such an attribute does not preserve its natural distribution; instead, each model develops its own run-dependent bias, making the attribute uncontrollable rather than reliably generated. Similarly, captions for dance or sports videos describe the activity ("people dancing") without specifying speed, pose, or spatial arrangement, so these attributes converge to per-model defaults. The practical fix is to explicitly caption the attributes that matter for generation, unless they are already implied by other fields that remain in the caption.

I LIMITATIONS

Our study focuses on a controlled synthetic setting designed for precise factor isolation. Future work could extend the evaluation to more diverse video domains, model configurations, and training scales.