F LITE

TECHNICAL REPORT

Simo Ryu Pengqi Lu Javier Martín Juan Iván de Prado Alonso*

ABSTRACT

We introduce F Lite, an open-source 10 billion parameter text-to-image Diffusion Transformer (DiT). Our architecture utilizes cross-attention for textual conditioning, selected based on favorable FLOP efficiency analyses for our scale and setup. F Lite incorporates architectural refinements like learnable register tokens, residual value connections, and employs μ-Parameterization for stable large-scale training. The model was trained on Freepik's internal dataset comprising approximately 80 million copyright-safe images, making it the first publicly available model of this scale trained exclusively on legally compliant and SFW content. The model was trained with a moderate compute budget—64 H100 GPUs over two months—offering a solid illustration of what is achievable within medium-range data and compute regimes. We release F Lite as a valuable baseline for researchers advancing large-scale diffusion models for text-to-image generation.

1 Introduction

Diffusion models [1, 2, 3, 4] combined with the Transformer architecture [5] have enabled significant advances in text-to-image generation. Diffusion Transformers (DiTs) [6] are a leading architecture, with variations in conditioning mechanisms. While recent models like Flux [7] and Stable Diffusion 3 (SD3) [8] utilize Multi-Modal Diffusion Transformer (MMDiT) blocks integrating text and image tokens early, alternatives based on cross-attention conditioning remain effective.

Based on computational resource constraints and internal FLOP efficiency analyses [9], we identified cross-attention as offering a potentially advantageous compute-to-performance ratio for our target scale and training infrastructure. This informed the design of F Lite, our 10 billion parameter open-source text-to-image model presented in this report.

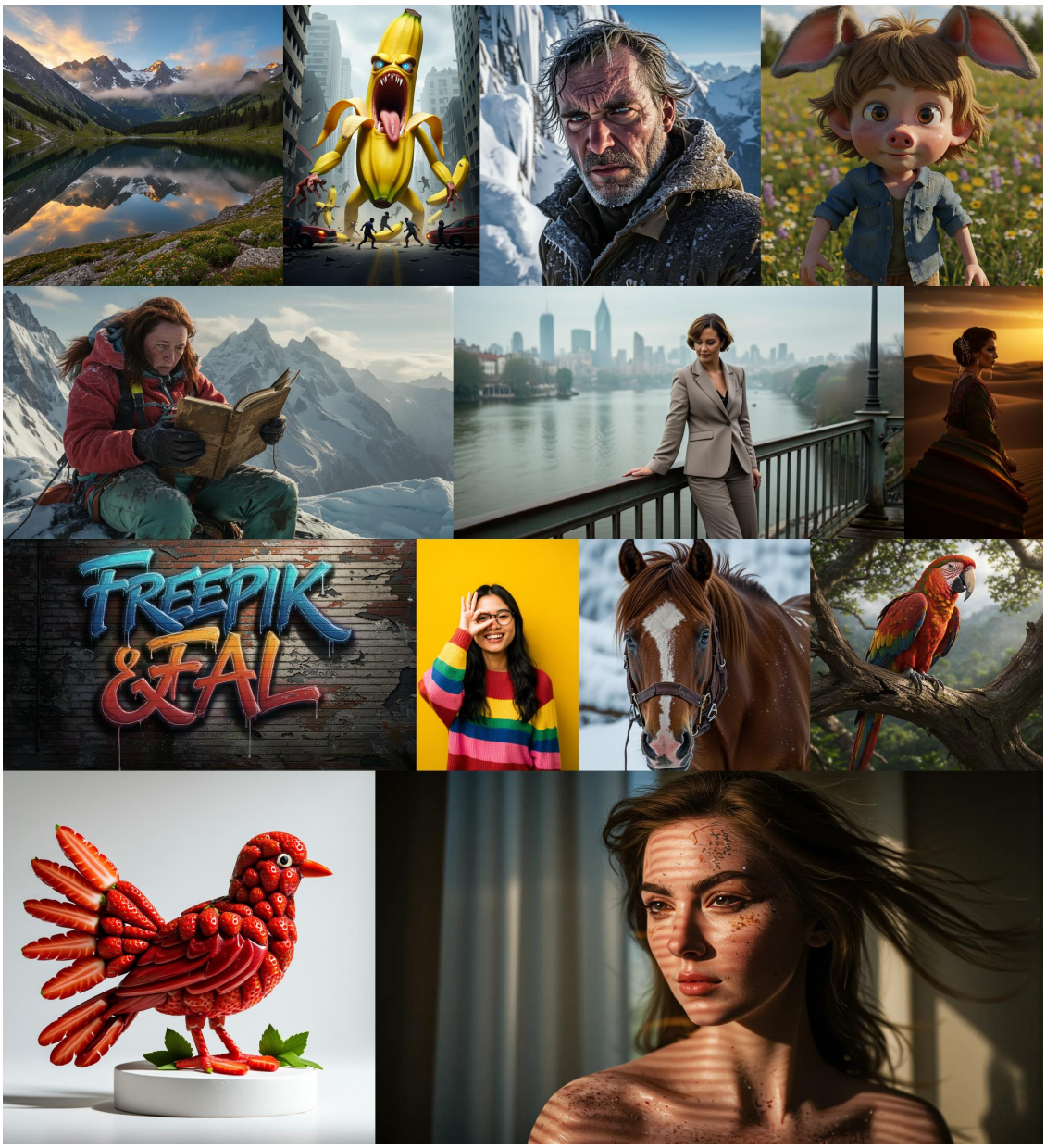
F Lite builds upon the DiT framework [6] but integrates text conditioning via cross-attention. We aim to provide a scalable, publicly available baseline incorporating established practices alongside modern architectural refinements and efficient, stable training strategies.

This technical report provides a comprehensive overview of F Lite's design and training. Key contributions and features detailed herein include:

- **Scalable Cross-Attention DiT:** Demonstrating the effectiveness of cross-attention conditioning in a $\sim$10B parameter DiT, highlighting its potential FLOP efficiency.
- **Text Encoder Optimization:** Observing that utilizing intermediate layers (specifically, the 17th layer) of the T5 text encoder [10] leads to noticeable improvements in convergence rate compared to using the final layer, a finding made possible through our online text encoding setup.
- **Architectural Enhancements:** Integration of learnable register tokens and residual value connections [11] for improved representation capacity and performance with minimal overhead.
- **Scalable Training Strategy:** Leveraging μ-Parameterization (μP) [12] principles for stable hyperparameter tuning across scales.
- **Medium-range Data and Compute Training:** Demonstrating the capabilities achievable when training a large-scale model with limited but high-quality data and modest computational resources.

By sharing F Lite, its underlying methodology, and our empirical findings, we aim to contribute a valuable resource for researchers and practitioners exploring the frontiers of large-scale generative modeling.

*simo@fal.ai, lpengqi@freepik.com, javier.martin@freepik.com, ideprado@freepik.com



2 Model Architecture

F Lite follows the latent diffusion paradigm [3], operating on latents from the pre-trained VAE from Flux Schnell [7]. Its core is a Transformer architecture [5, 6] with specific modifications for conditioning and performance.

2.1 Overall Structure

Input latent images z are patchified and flattened into a sequence of tokens. N_{reg} learnable *register tokens* are prepended to this sequence. The combined sequence passes through L Transformer blocks, each conditioned on timestep t and text embeddings c. A final linear layer projects the output tokens (excluding register tokens) back to the VAE latent dimension to predict the denoised latent $\hat{z}$.

2.2 Scaling Cross-Attention

Unlike MMDiT approaches [7, 8], F Lite uses separate self-attention (for image tokens) and cross-attention (image tokens attend to text tokens) layers within each block. This design was chosen for its favorable efficiency characteristics as suggested by [9].

Text conditioning c is derived from a pre-trained T5-XXL encoder [10]. A crucial finding from our experiments, enabled by online text encoding, is the superior performance achieved by extracting embeddings from an **intermediate layer**. In [13], they show that normalized MSE of hidden representations suddenly improves for final layers, suggesting that 'generality' of the features is lost in these layers.

This motivates the question of which layer contains the most 'general' features. As shown in Figure 1, we found that using the hidden state from the **13 to 17th layer** (middle layer) yielded a **25-30% improvement in training efficiency** (faster convergence) compared to using the final layer. This suggests that intermediate representations may be more suitable for conditioning diffusion models in this setup.

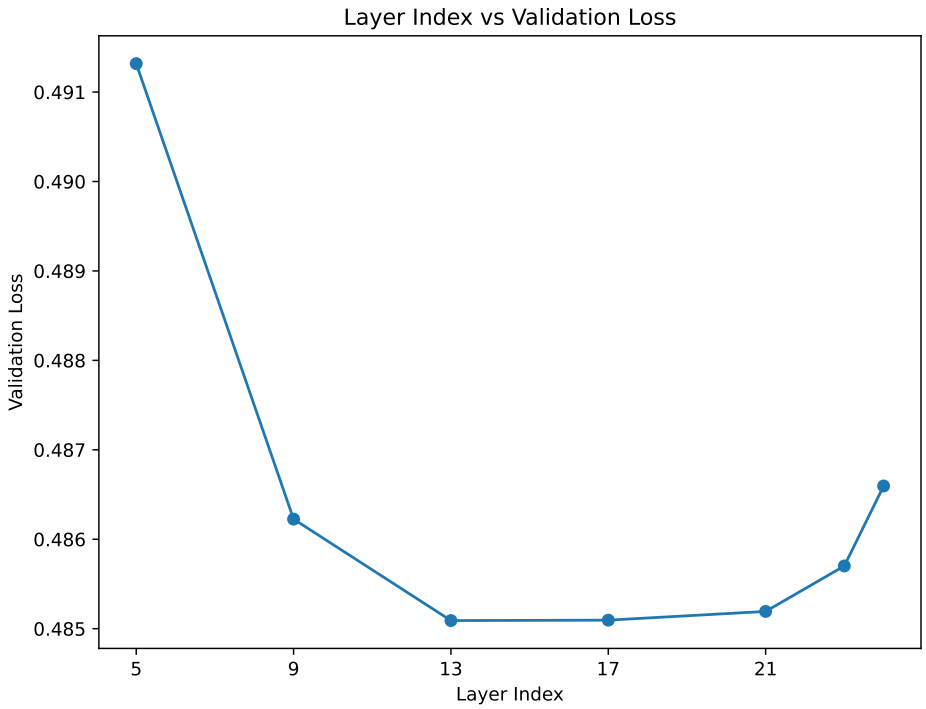


Figure 1: Validation loss across different T5 encoder layers. The x-axis shows the layer index (with layer 0 being the first layer and layer 24 being the final layer). The y-axis shows the final validation loss. The U-shaped curve demonstrates that intermediate layers (particularly around layer 17) provide more effective conditioning features than either early or final layers.

2.3 Positional Encoding

We employ 2D Rotary Positional Embeddings (RoPE) [14] for image patch tokens to encode spatial information robustly across varying resolutions and aspect ratios. Register tokens receive identity RoPE embeddings. We empahsize the use of (0, 1) for sin/cos values to preserve identity features for Q, K values, avoiding possible mistake like using (1, 1) or (0, 0).

2.4 Architectural Refinements

F Lite incorporates several modern components:

- **Register Tokens:** N_{reg} (e.g., 16) learnable tokens prepended to the image sequence serve as auxiliary capacity, participating in self-attention but discarded before output projection.
- **Residual Value Connection:** Following [11], the attention value projection v_l is a learned convex combination of the current layer's projection and the first layer's value projection v_0, providing consistent small gains with negligible compute cost.
- **Normalization Options:** We optionally include RMSNorm applied independently to queries and keys (QK Normalization) [7] before attention score computation for potential stability benefits, particularly at high resolutions.

Normalization within the Transformer blocks uses RMSNorm (non-learnable RMSNorm used in final configuration). We refer to the table 1 for the validation loss of different configurations.

Configuration	Validation Loss
Scale + Shift + Gate (False, False, False)	0.52815
Scale + Shift + Gate (False, True, False)	0.52864
Scale + Shift + Gate (True, False, False)	0.52641
Scale + Shift + Gate (True, True, True)	0.52237
Value Residual	0.52164
Unlearnable RMSNorm with Bias	**0.52084**

Table 1: Comparison of validation loss for different architectural configurations. Lower scores indicate better performance. The first four rows show variations of Scale, Shift, and Gate parameters.

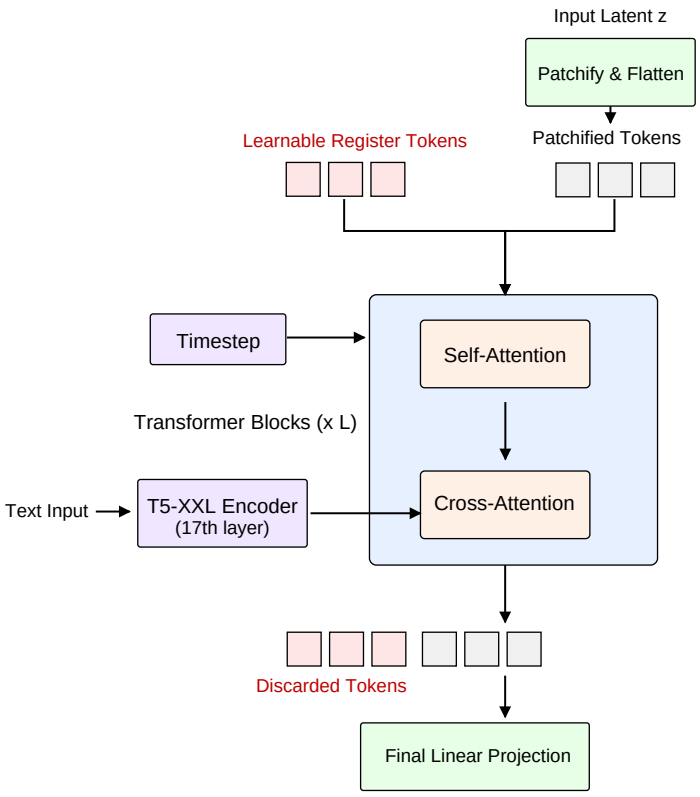


Figure 2: F Lite architecture overview, showing the cross-attention conditioning mechanism and key architectural components.

3 Training Strategy

Training F Lite involved a multi-stage approach, stable optimization techniques, and efficient data handling.

3.1 Multi-Stage Training Approach

Training progressed through stages of increasing resolution and complexity:

1. **Low-Resolution Stage (256×256 pixels, 512×512 pixels):** Focused on learning core concepts and text alignment on center-cropped images. This constituted the bulk of pre-training compute.
2. **High-Resolution Stage ($\geq 1024 \times 1024$ pixels):** Fine-tuning on higher resolution images, incorporating dynamic aspect ratios and resolutions within batches

3.2 Optimizer and Stability

- **Optimizer & μP:** We used AdamW [15] with hyperparameters set according to μ-Parameterization (μP) principles [12]. This involved scaling learning rates and weight decay based on parameter tensor dimensions (fan-in/fan-out) and using distinct settings for weights, biases, and normalization parameters, ensuring hyperparameter stability across scales.
- **Learning Rate Schedule:** The Warmup-Stable-Decay (WSD) schedule was employed (warmup, long stable phase, final decay) [16], offering a flexible scheduling framework that helped us tracking the training progress by regular cooldown phases.

3.3 Pretraining Techniques

Several techniques were employed to maximize training efficiency and stability.

Resolution-Aware Timestep Sampling: We implemented an adaptive timestep sampling strategy. A base lognormal-like distribution [4] was used to sample timesteps σ. The noise level was shifted based on image resolution using a time shift parameter α, calculated as $\alpha = \sqrt{\text{num\_image\_tokens}/K}$ (where K is a constant, e.g., 32, resulting in $\alpha \approx 2$ for 512px and $\alpha \approx 4$ for 1024px), pushing higher resolution images towards higher noise levels initially: $\sigma_{\text{shifted}} = \sqrt{\sigma^2 + \alpha^2}$. Critically, to counteract timestep "starvation" at low noise levels (small σ) caused by this shifting, we incorporated **10% uniform sampling** across the entire timestep range, as suggested by [17]. This ensures that the model receives sufficient gradients for fine details, which was found crucial for stability and quality, especially at high resolution.

Sequence Dropout: We implemented an option to randomly drop a fraction of image patch tokens (excluding register tokens) during the forward pass. Using 50% token dropout allowed us to effectively double the batch size without increasing step time while simultaneously improving convergence. This technique significantly enhanced training efficiency and was extensively utilized during high-resolution training. Following the approach suggested by [18], we included a final phase of unmasked fine-tuning to ensure the model could properly process complete sequences.

4 Post Training Alignment

To enhance image quality and alignment with user preferences, we applied a two-stage post-training process.

4.1 Supervised Fine-Tuning (SFT)

The pre-trained model underwent SFT on a curated dataset of $\sim$100K high-quality images. This stage used the same optimization strategy but focused on this higher-quality data distribution, often emphasizing higher resolutions.

4.2 Reinforcement Learning from Human Feedback (RLHF)

We further aligned the model using RLHF on a dataset of human preference pairs (winning/losing images for the same prompt).

- **Initial Approach (MaPO):** We started with Margin-aware Preference Optimization (MaPO) [19], using its loss formulation combining reconstruction and preference margin terms.

- **Adapted GRPO for Improved Stability:** We subsequently adapted concepts from DeepSeek's Group Relative Policy Optimization (GRPO) [20]. While GRPO was designed for LLMs, we applied its principle of batch-relative normalization to the text-to-image RLHF setting. Specifically, we computed the MaPO-style log-odds preference signal ($\text{log\_odds} = f(\mathcal{L}^w_{\text{recon}}) - f(\mathcal{L}^l_{\text{recon}})$) and then normalized it using batch statistics (mean $\mu_{\text{log\_odds}}$ gathered across all distributed workers):

$$\text{relative\_log\_odds} = \text{log\_odds} - \mu_{\text{log\_odds}} \tag{1}$$

  This centered log-odds value was then used in the ratio loss term: $\text{ratio} = \log \sigma(\text{relative\_log\_odds} \cdot \alpha)$.
- **Benefits:** This GRPO adaptation provided more stable gradients, implicitly adjusted for batch difficulty, and yielded continued improvement in image quality and preference scores where MaPO had started to plateau.

The RLHF phase demonstrably improved automatic metrics (CLIP score, aesthetic score) and human preference ratings.

5 Experimental Results and Analysis

5.1 Training Setup

F Lite ($\sim$10B parameters, 40 layers, hidden size 3072, 16 heads) was trained on an internal dataset of $\sim$80M filtered high-quality images from the Freepik catalog. Compute involved up to 64 H100 GPUs over $\sim$1.5-2 months ($\sim$10e22 BF16 FLOPs effective compute, about twice the compute of SD3 [8]). We used the Flux Schnell VAE and T5-XXL (17th layer output) for text encoding.

5.2 Qualitative Analysis

Qualitative samples generated by F Lite demonstrate the ability to produce diverse, high-fidelity images adhering well to complex prompts. The model shows particular strength in generating illustrative and vector styles, likely reflecting the training data distribution.

However, certain limitations were observed:

- **High-Frequency Detail:** Some images, particularly photorealistic ones, lacked fine-grained textures (e.g., skin pores, fabric detail). This might be improvable with further high-resolution training.
- **Anatomy and Complexity:** Complex scenes or intricate anatomy sometimes resulted in malformations, a common challenge in generative models, though potentially exacerbated by specific training dynamics.
- **Short Prompt Performance:** The model performed significantly better with long, descriptive prompts (as used in training) and struggled with very short prompts. Fine-tuning on shorter captions could address this.
- **Text Rendering:** While capable of generating text-like elements, accurate rendering of specific text within images remained limited.

Despite F Lite's capability to generate high-quality images with impressive aesthetics and composition, we observe anatomical malformations and generation errors frequently. We believe our architectural choices and training approach are fundamentally sound, and that these issues could be substantially mitigated through extended training using more compute over a larger dataset. This hypothesis is supported by observations from scaling laws in diffusion models [9], which indicate that both model and data scaling contribute significantly to generation quality improvements.

6 Conclusion

We presented F Lite, a 10 billion parameter open-source text-to-image Diffusion Transformer employing cross-attention conditioning. Our work provides a robust baseline incorporating modern architectural refinements and scalable, efficient training strategies.

Key contributions include the validation of cross-attention's effectiveness at scale (evaluated by FLOPs), the novel and significant finding that utilizing intermediate T5 layers boosts training efficiency by 25-30%, and the detailed documentation of practical techniques like resolution-aware timestep sampling with uniform sampling correction and the successful application of the WSD scheduler.

By open-sourcing F Lite, we hope to accelerate research in large-scale generative modeling and provide a valuable tool for the AI community. We encourage collaboration and further exploration based on this work.

Acknowledgments

We sincerely thank the entire team involved in the F Lite project for their dedication and contributions. We are also deeply grateful to the broader open-source AI community, whose libraries, research, and discussions significantly influenced and enabled this work. We acknowledge the compute resources provided by Nebius which were essential for conducting the large-scale training experiments.

References

[1] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020.

[2] Yang Song, Jascha Sohl-Dickstein, Diederik P. Kingma, Abhishek Kumar, Stefano Ermon, and Ben Poole. Score-based generative modeling through stochastic differential equations, 2021.

[3] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models, 2022.

[4] Tero Karras, Miika Aittala, Timo Aila, and Samuli Laine. Elucidating the design space of diffusion-based generative models, 2022.

[5] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023.

[6] William Peebles and Saining Xie. Scalable diffusion models with transformers, 2023.

[7] Black Forest Labs. Flux. `https://github.com/black-forest-labs/flux`, 2023.

[8] 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, Kyle Lacey, Alex Goodwin, Yannik Marek, and Robin Rombach. Scaling rectified flow transformers for high-resolution image synthesis, 2024.

[9] Zhengyang Liang, Hao He, Ceyuan Yang, and Bo Dai. Scaling laws for diffusion transformers, 2024.

[10] 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. *Journal of machine learning research*, 21(140):1–67, 2020.

[11] Zhanchao Zhou, Tianyi Wu, Zhiyun Jiang, Fares Obeid, and Zhenzhong Lan. Value residual learning, 2025.

[12] Greg Yang, Edward J. Hu, Igor Babuschkin, Szymon Sidor, Xiaodong Liu, David Farhi, Nick Ryder, Jakub Pachocki, Weizhu Chen, and Jianfeng Gao. Tensor programs v: Tuning large neural networks via zero-shot hyperparameter transfer, 2022.

[13] Leo Gao, Tom Dupré la Tour, Henk Tillman, Gabriel Goh, Rajan Troll, Alec Radford, Ilya Sutskever, Jan Leike, and Jeffrey Wu. Scaling and evaluating sparse autoencoders. *arXiv preprint arXiv:2406.04093*, 2024.

[14] Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. Roformer: Enhanced transformer with rotary position embedding. *Neurocomputing*, 568:127063, 2024.

[15] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017.

[16] Kaiyue Wen, Zhiyuan Li, Jason Wang, David Hall, Percy Liang, and Tengyu Ma. Understanding warmup-stable-decay learning rates: A river valley loss landscape perspective, 2024.

[17] Yoav HaCohen, Nisan Chiprut, Benny Brazowski, Daniel Shalem, Dudu Moshe, Eitan Richardson, Eran Levin, Guy Shiran, Nir Zabari, Ori Gordon, Poriya Panet, Sapir Weissbuch, Victor Kulikov, Yaki Bitterman, Zeev Melumian, and Ofir Bibi. Ltx-video: Realtime video latent diffusion, 2024.

[18] Hongkai Zheng, Weili Nie, Arash Vahdat, and Anima Anandkumar. Fast training of diffusion models with masked transformers, 2023.

[19] Jiwoo Hong, Sayak Paul, Noah Lee, Kashif Rasul, James Thorne, and Jongheon Jeong. Margin-aware preference optimization for aligning diffusion models without reference, 2024.

[20] Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, Y. K. Li, Y. Wu, and Daya Guo. Deepseekmath: Pushing the limits of mathematical reasoning in open language models, 2024.