TEXT-TO-IMAGE GENERATIVE MODELING WITHOUT LANGUAGE PRETRAINING

Anonymous authors
Paper under double-blind review

ABSTRACT

Modern text-to-image (T2I) systems use internet-scale language pretraining for semantic conditioning. *We ask whether language pretraining is necessary for strong T2I generation.* We introduce `Xanthus`, a family of early-fusion multimodal autoregressive models pretrained entirely from scratch on GPIC, a dataset of 100M image-text pairs. `Xanthus`-0.52B outperforms LlamaGen-XL/8 (**48.0** vs. 63.5 gFD) with **18.2**× less pretraining compute. Further, `Xanthus`-1.6B outperforms JiT-T2I-XXL/8++ (**35.3** vs. 43.2 gFD) with **275**× less pretraining compute. Controlled data composition and scheduling experiments show that external text-only pretraining does not improve T2I generation. Under the same pretraining budget, `Xanthus` also learns image captioning from the paired image-text data while maintaining competitive T2I generation quality. We show that strong T2I systems can be trained from scratch without internet-scale language pretraining.

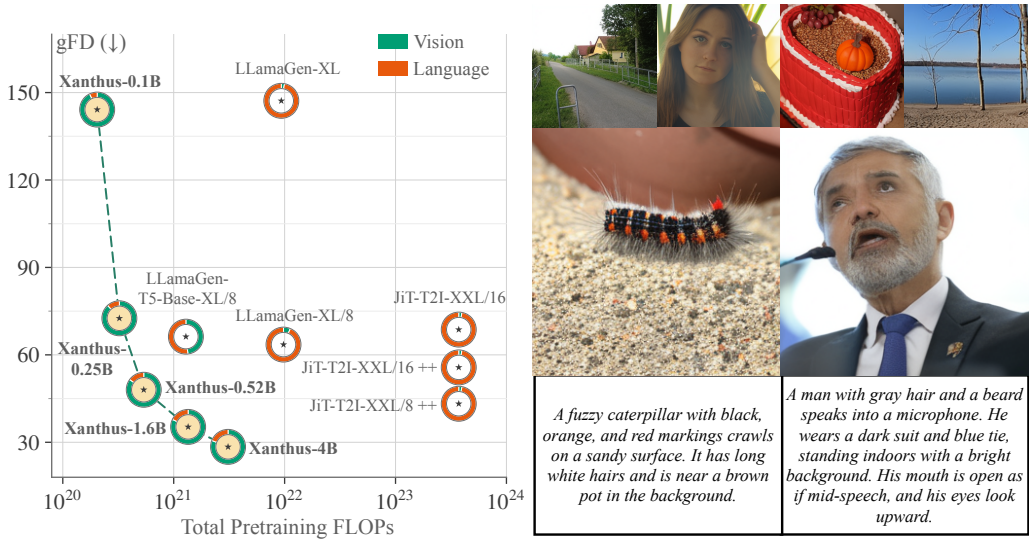

Figure 1: **Text-to-image generation without large-scale language pretraining. Left:** Generation quality versus total pretraining FLOPs. For each model, the ⋆ marks its total pretraining FLOPs and gFD, while the donut decomposes that compute into language and vision pretraining FLOPs. `Xanthus` (Ours) achieves lower gFD with substantially less total pretraining compute than LlamaGen baselines using FLAN-T5 encoders (0.11-1.22B) with 0.78-0.90B image generators, and JiT-T2I baselines using Qwen3-1.7B with a 1.10B image generator. **Right:** The collage shows 256×256 samples generated by `Xanthus`-4B. We show two generations with text prompts.

1 INTRODUCTION

Text-to-image (T2I) generation maps natural-language descriptions to images. Modern T2I systems have an ingrained reliance on internet-scale language pretraining [1] for semantic conditioning of image generators. This dependence is realized via two designs, distinguished by how and when text tokens interact with image tokens: (1) *Late fusion*, the predominant design, uses two separate models: a large model is pretrained on text, and the resulting text embeddings are used to condition a

[1] Throughout this paper, *language pretraining* refers to pretraining on web-scale text-only corpora, excluding image-paired captions.

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.

separate image generator through cross- or joint attention (Saharia et al., 2022; Chen et al., 2024; Black Forest Labs, 2025; Zhao et al., 2026); (2) *Early fusion* removes the separate text encoder while retaining language pretraining within a single multimodal model trained on text-only and image-text data, with captions used as a prefix to condition image generation (Team et al., 2024; Wang et al., 2024; Zhou et al., 2025; Liu et al., 2026). Across both designs, language pretraining dominates the compute budget, accounting for over 80% of total pretraining FLOPs in representative T2I systems. This raises three issues. First, its contribution to T2I generation has not been systematically quantified. Second, language pretraining cost must be accounted for when comparing the total cost of T2I systems, rather than treated as external to the system. Third, in late-fusion systems, large frozen language models add systems overhead both during T2I pretraining and inference.

In this work, we show that strong T2I generation can be achieved without language pretraining. We introduce `Xanthus`, a family of early-fusion autoregressive models that learn text conditioning directly from image–text pairs using next-token prediction. We quantify the contribution of language pretraining to T2I generation using GPIC (Chandrasegaran et al., 2026), a dataset of 100M image–text pairs, through three experimental axes: (1) **From-scratch pretraining:** we train all components from scratch, including the text and image tokenizers and the generative model; (2) **Data composition and scheduling:** under a fixed pretraining budget, we control the mixture and schedule of text-only, T2I, and I2T examples and measure their effect on T2I generation quality; and (3) **Compute comparisons:** we pretrain representative T2I baselines on GPIC and compare generation quality as a function of total pretraining compute, decomposed into language and vision. Through these experiments, we quantify the utility of language pretraining for T2I generation.

Our results show that `Xanthus` achieves strong T2I generation without language pretraining, with generation Fréchet distance (gFD) consistently improving as model size scales. Particularly, `Xanthus`-0.52B achieves 48.0 gFD, outperforming LlamaGen-XL/8 (FLAN-T5-XL 1.22B + 0.90B image generator; 63.5 gFD) using **18.2× less** total pretraining compute. `Xanthus`-1.6B achieves 35.3 gFD, outperforming JiT-T2I-XXL/8++ (Qwen3-1.7B + 1.10B image generator; 43.2 gFD) using **275× less** total pretraining compute. Our data composition and scheduling experiments further show that external text-only pretraining does not improve T2I generation. As a byproduct, `Xanthus` learns both T2I generation and image captioning directly from image-text pairs under the same pretraining budget while maintaining competitive T2I generation quality. Our work provides a simple recipe for pretraining T2I models without language pretraining and prompts rethinking the role of generic language pretraining for multimodal generation. Code and models will be released.

2 Related Work

Modern T2I systems use large-scale language pretraining through two designs. In *late fusion*, the predominant paradigm, a language model is pretrained separately on text-only data, and its embeddings condition a visual generator. Imagen (Saharia et al., 2022) and PixArt-α (Chen et al., 2024) use T5-XXL, LlamaGen-T2I (Sun et al., 2024) uses FLAN-T5-XL, SANA (Xie et al., 2025) uses Gemma-2B and PixelGen (Ma et al., 2026) uses Qwen3-1.7B. In *early fusion*, the separate text encoder is removed, but text-only pretraining is incorporated into multimodal pretraining. Chameleon (Team et al., 2024), Transfusion (Zhou et al., 2025), and Emu3 (Wang et al., 2024) train jointly on text-only and image–text data, while Tuna-2 (Liu et al., 2026) initializes from Qwen2.5-7B and continues training on a mixture that includes text-only data. Figure 2 compares the text and vision tokens used to pretrain these T2I systems. Systems with language pretraining consume $10^{12} \sim 10^{14}$ text tokens, often exceeding the vision budget. In contrast, `Xanthus` uses no text-only language pretraining; its only language supervision comes from caption tokens paired with images,

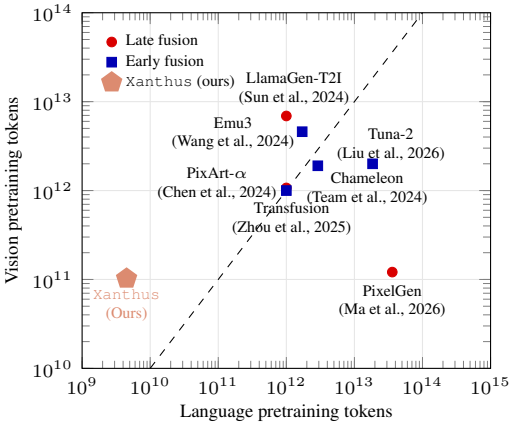

Figure 2: **Language vs. vision pretraining tokens.** Language tokens count text-only pretraining. Emu3 and Tuna-2 values are estimated.

more than two orders of magnitude lower than the language-pretraining budgets shown. We note that training T2I models without a pretrained language model is not new. Early systems such as DALL-E, CogView, and GLIDE learned text conditioning directly from paired image–text data (Ramesh et al., 2021; Ding et al., 2021; Nichol et al., 2022). Parti (Yu et al., 2022) observed that text-encoder pretraining marginally improves T2I generation performance, leaving the role of language pretraining as an open question. However, subsequent high-performing T2I systems largely shifted toward pretrained language models for text conditioning. We revisit this question and systematically quantify the role of language pretraining for modern T2I generation.

3 Approach

`Xanthus` is a decoder-only Transformer with a joint text–image vocabulary. We model text and image tokens as a single sequence using the next-token prediction, following prior multimodal early-fusion works (Team et al., 2024; Wang et al., 2024). All components, including the text tokenizer, image tokenizer, and multimodal generative model, are trained from scratch. Below, we describe the tokenizers, sequence format, training objective, and pretraining data composition and scheduling.

3.1 Tokenization

Images. We train a discrete image tokenizer from scratch on GPIC images, following the encoder–decoder design of Cosmos Tokenizer (Agarwal et al., 2025) with finite scalar quantization (FSQ) (Mentzer et al., 2024). The tokenizer downsamples images by a factor of 8 along each spatial dimension, with a codebook size of 64,000. Each 256×256 image is represented as $N = 1{,}024$ image tokens $\mathbf{x} = (x_1, \ldots, x_N)$, obtained by flattening the 32×32 grid in raster order.

Text. We train a byte-level BPE tokenizer (Sennrich et al., 2016) from scratch on GPIC captions and OpenWebText, with a vocabulary size of 16,384. Each caption is represented as $\mathbf{c} = (c_1, \ldots, c_L)$.

Joint vocabulary. Text tokens, image tokens, and special tokens form a single vocabulary of size 80,640, comprising 16,384 text tokens, 64,000 image tokens, and 256 special tokens. The model uses a shared embedding table and output head. We use `[SOV]` and `[EOV]` to mark the beginning and end of an image, `[EOS]` to mark the end of a sequence, and `[PAD]` for sequence padding.

3.2 Sequence Format and Training Objective

We use four sequence formats for pretraining, shown in Figure 3. Text-only sequences are packed, while image-containing sequences are padded to a fixed length with `[PAD]` tokens, which are excluded from the training loss. We do not pack multiple image–caption pairs into a sequence. In image-only (I) sequences, image tokens are predicted from `[SOV]`. In image-to-text (I2T) sequences, image tokens are predicted first, followed by caption tokens conditioned on the image. In text-to-image (T2I) sequences, image tokens are predicted conditioned on the caption.

For classifier-free guidance (CFG) (Ho & Salimans, 2022), we adopt the standard practice of dropping the caption from a fraction p_{drop} of T2I examples to train the unconditional branch. At infer-

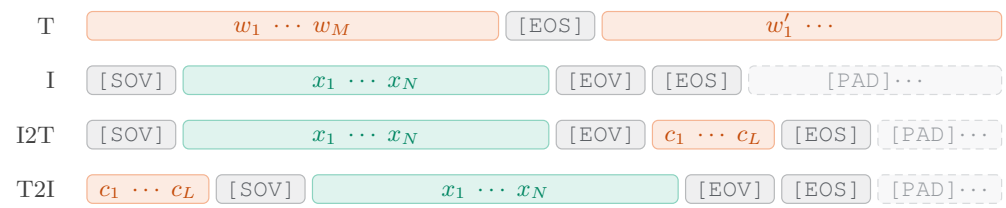

Figure 3: **Sequence formats.** Text-only pretraining tokens, caption tokens, and image tokens occupy one joint vocabulary alongside special tokens. The top two formats are unimodal, while the bottom two represent paired image-caption examples. Rows I and I2T share the same `[SOV]` $\mathbf{x}$ `[EOV]` prefix, which is also obtained when dropping the caption from a T2I sequence used for classifier-free guidance. All three therefore train unconditional image generation and differ only in what follows `[EOV]`. Image-caption sequences are padded to a fixed length, with `[PAD]` positions excluded from the loss. Box widths are schematic and not proportional to token counts.

ence, the conditional and unconditional image-token logits are combined as $\boldsymbol{\ell}_{\text{cfg}} = \boldsymbol{\ell}_{\text{u}} + \gamma(\boldsymbol{\ell}_{\text{c}} - \boldsymbol{\ell}_{\text{u}})$, where γ is the guidance scale. We note that I2T sequences share the same image prefix as unconditional T2I sequences, and therefore also train the unconditional image branch used for CFG. We use weighted next-token cross-entropy, $\mathcal{L} = (\mathcal{L}_{\text{text}} + \lambda_{\text{img}}\mathcal{L}_{\text{img}})/M$, where $\mathcal{L}_{\text{text}} = -\sum_{i\in\mathcal{T}} \log p_\theta(s_i \mid s_{<i})$, $\mathcal{L}_{\text{img}} = -\sum_{i\in\mathcal{I}} \log p_\theta(s_i \mid s_{<i})$, and $M = |\mathcal{T}| + |\mathcal{I}|$. Here, $\mathcal{T}$ and $\mathcal{I}$ denote the non-padding text and image token positions, respectively, with special tokens included in $\mathcal{T}$. We set $\lambda_{\text{img}} = 0.5$ to prevent image tokens dominating the loss similar to Emu3 (Wang et al., 2024).

3.3 Pretraining Data Composition and Scheduling

To study the utility of language pretraining for T2I generation, we vary the composition and scheduling of text-only, T2I, and I2T examples under a **fixed** pretraining budget. Figure 4 illustrates the five configurations. Unless otherwise specified, caption dropout is applied to T2I examples at rate p_{drop}.

- **Baseline T2I.** We train exclusively on T2I examples.
- **Staged: Language → T2I.** We allocate the first α fraction of training to text-only pretraining, followed by T2I for the remaining $1 - \alpha$ fraction. This is the only configuration that uses external text-only data for pretraining.
- **Staged: I2T → T2I.** We allocate the first β fraction of training to I2T, followed by T2I for the remaining $1 - \beta$ fraction.
- **Mixing: T2I + I2T.** We interleave T2I and I2T examples throughout training, allocating a fraction η to I2T and $1 - \eta$ to T2I.
- **Mixing: T2I without caption dropout + I2T.** We set $\eta = p_{\text{drop}}$ and disable caption dropout. This preserves the baseline's conditional and unconditional image-training tokens while additionally providing I2T supervision.

4 Experiments

4.1 Experimental Setup

Dataset and pretraining protocol. We pretrain on GPIC (Chandrasegaran et al., 2026), a dataset of 100M image-text pairs at 256×256 resolution. All configurations use the same pretraining budget: 97,632 iterations at a global batch size of 1,024, corresponding to one epoch on GPIC, with an identical warmup-stable-decay (WSD) learning-rate schedule (Hu et al., 2024). We study the effect of data composition and schedules on T2I performance under fixed pretraining budget. For language pretraining, we use ClimbMix (Diao et al., 2026), packed without padding. Model dimensions, peak learning rates, and additional training details are provided in Appendix A.

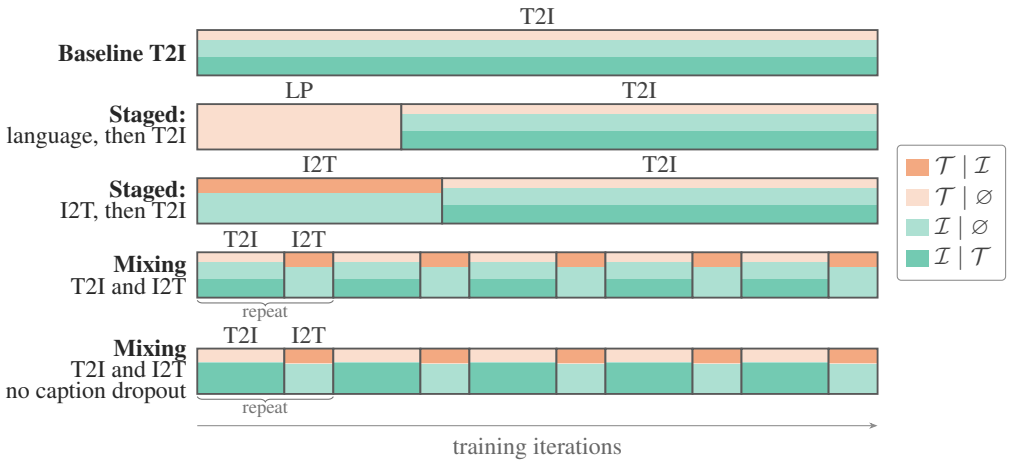

Figure 4: **Pretraining data composition and scheduling described in Section 3.3.** The four token types are: image tokens conditioned on text ($\mathcal{I} \mid \mathcal{T}$); image tokens without text conditioning ($\mathcal{I} \mid \varnothing$); text tokens conditioned on images ($\mathcal{T} \mid \mathcal{I}$); and text-only tokens ($\mathcal{T} \mid \varnothing$). Vertical boundaries denote training phases, while horizontal bands show the token composition within each phase. Bands represent interleaved examples. Band height and width are schematic.

Table 1: **Image tokenizer reconstruction quality on GPIC Test-50K at 256×256 resolution.** Our tokenizer, trained from scratch on GPIC, outperforms pretrained Cosmos-DI-8x8 across all reconstruction metrics. Real-vs-Real reports reference scores between GPIC Test-50k and Test-1M real image sets.

Tokenizer	rFD ↓	Precision ↑	Recall ↑	Density ↑	Coverage ↑	MMD ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Cosmos-DI-8x8	37.926	0.943	0.912	1.011	0.953	0.036	25.380	0.776	0.153
`Xanthus`	**18.852**	**0.948**	**0.936**	**1.029**	**0.961**	**0.012**	**25.614**	**0.792**	**0.062**
Real-vs-Real	7.593	0.955	0.949	1.021	0.972	0.0	–	–	–

Tokenizer training and evaluation. • *Image tokenizer.* We train our image tokenizer for one epoch using reconstruction and perceptual losses, followed by an additional half epoch with a PatchGAN adversarial loss. Compared to Cosmos-DI-8x8 (Agarwal et al., 2025), we increase the decoder width from 128 to 192. To assess reconstruction quality, we compare our tokenizer with pretrained Cosmos-DI-8x8 and include Real-vs-Real scores as a reference. Our tokenizer outperforms pretrained Cosmos-DI-8x8 across all reconstruction metrics (Table 1). As with prior discrete tokenizers, reconstruction fidelity is lower for small text and fine glyphs (Team et al., 2024).

• *Text tokenizer.* Despite having a $9.3\times$ smaller vocabulary than Qwen3, our tokenizer achieves comparable compression on GPIC test captions (4.774 vs. 4.763 bytes/token), outperforming the FLAN-T5 tokenizer (4.152 bytes/token; Table 2). On held-out OpenWebText, our tokenizer achieves 4.086 bytes/token. OpenWebText is used only to train the BPE tokenizer, not the multimodal generative models. Full tokenizer training details are provided in Table 10.

Model configurations. We train `Xanthus` at five sizes: 0.1B, 0.25B, 0.52B, 1.6B, and 4B parameters. Model architecture details are in Table 9. We use a sequence length of 1280 for all experiments. We use $p_{drop} = 0.1$ for all experiments unless otherwise specified. All T2I generators are pretrained on GPIC for one epoch at 256×256 resolution.

Table 2: **Text tokenizer comparison on GPIC test captions.** Higher bytes/token and lower tokens/caption indicate better compression.

Tokenizer	Vocab.	Bytes/token ↑	Tokens/caption ↓
FLAN-T5	32,100	4.152	51.1
Qwen3	151,669	4.763	44.6
`Xanthus`	**16,384**	**4.774**	44.5

Late-fusion methods. Late fusion is the predominant design in modern T2I systems. Selecting baselines for this study is non-trivial: existing systems differ in pretraining objective, language conditioning, image representation, and training setup, while our study requires public pretraining code and models that enable controlled interventions to study the role of language pretraining. We select two representative families, LlamaGen (Sun et al., 2024) and JiT-T2I (Li & He, 2026; Ma et al., 2026). They cover (i) autoregressive and diffusion-based generation, (ii) encoder-decoder and decoder-only language models of different sizes, and (iii) image sequence lengths of 256 and 1,024. For both families, we use the authors' official codebases and follow their training configurations unless stated otherwise. Below, we describe our modifications and summarize them in Table 3.

• **LlamaGen (Sun et al., 2024).** This method uses a FLAN-T5-XL text encoder for language modeling, a pretrained discrete image tokenizer, and an autoregressive image generator.

- **Default (LLamaGen-XL).** We use the original LlamaGen configuration: FLAN-T5-XL, the pretrained image tokenizer producing 256 image tokens from a codebook of 16,384 entries, and the 0.78B image generator.
- **Use `Xanthus` image tokenizer (LlamaGen-XL/8).** Starting with the default configuration, we replace the image tokenizer with our GPIC-trained tokenizer. This increases the visual vocabulary size by $4\times$, the image sequence length from 256 to 1,024, and the image generator size from 0.78B to 0.90B.
- **Use smaller text encoder (LlamaGen-XL-T5-Base/8).** Starting with LlamaGen-XL/8, we replace FLAN-T5-XL with FLAN-T5-base to study the effect of language model size.

• **JiT-T2I (Li & He, 2026; Ma et al., 2026).** This method uses pretrained Qwen3-1.7B (Yang et al., 2025) as the language model and a pixel-space flow-matching image generator.

Table 3: **Experiment configurations.** Three configurations each of LlamaGen and JiT-T2I, varying the language model, image sequence length, and perceptual supervision. LlamaGen and Xanthus are autoregressive; JiT is diffusion-based. Xanthus is trained without a pretrained language model. Pretraining tokens report the total language and vision tokens used during training; for Xanthus, all language tokens come from GPIC captions. FLAN-T5 counts the encoder parameters only.

	Language	Params		Pretraining tokens		Perceptual
Configuration	model	Language	Vision	Language	Vision	loss
LlamaGen-XL	FLAN-T5-XL	1.22B	0.78B	∼1T	25.6B	✓
LLamaGen-XL/8	FLAN-T5-XL	1.22B	0.90B	∼1T	102.4B	✓
LLamaGen-XL-T5-Base/8	FLAN-T5-Base	0.11B	0.90B	∼1T	102.4B	✓
JiT-T2I-XXL/16	Qwen3	1.70B	1.10B	36T	25.6B	
JiT-T2I-XXL/16 ++	Qwen3	1.70B	1.10B	36T	25.6B	✓
JiT-T2I-XXL/8 ++	Qwen3	1.70B	1.10B	36T	102.4B	✓
Xanthus	—	0.1–4B		4.5B	102.4B	✓

- **Default (JiT-T2I-XXL/16).** We use the original JiT-T2I-XXL/16 configuration: Qwen3-1.7B and no image tokenizer, with 16×16 image patches producing 256 image tokens.
- **Add perceptual loss (JiT-T2I-XXL/16 ++).** Starting with the default configuration, we add LPIPS loss (Zhang et al., 2018) to train the image generator. This tests the effect of perceptual supervision used to also train the image tokenizers in Xanthus and LlamaGen.
- **Use 8×8 patches (JiT-T2I-XXL/8 ++).** Starting with JiT-T2I-XXL/16 ++, we reduce the image patch size from 16×16 to 8×8, increasing the image sequence length from 256 to 1,024, matching Xanthus and the modified LlamaGen configurations.

Early-fusion methods. Xanthus is an early-fusion multimodal model. Reproducing existing early-fusion methods including Chameleon (Team et al., 2024) and Emu3 (Wang et al., 2024) is not feasible as the pretraining code and data are not public. Therefore, we study the role of language pretraining within Xanthus by varying the composition and scheduling of text-only, T2I, and I2T sequences under the same pretraining budget described in Section 3.3.

Evaluation protocol. We strictly follow the GPIC evaluation protocol to evaluate T2I generation and use generation Fréchet distance (gFD), Precision, Recall, Density, Coverage and MMD metrics. All models generate images at 256×256 resolution using the same set of 50K evaluation captions. We report the sampling and guidance settings used for each model. We also report total pretraining FLOPs, decomposed into language and vision, including the pretraining of language models, image tokenizers, and image generators.

Table 4: **Comparison of Xanthus with late-fusion T2I methods on GPIC.** Without language pretraining, Xanthus achieves stronger T2I generation than the late-fusion baselines: Xanthus-0.52B outperforms LlamaGen-XL/8 (48.02 vs. 63.52 gFD), while Xanthus-1.60B outperforms JiT-T2I-XXL/8++ (35.28 vs. 43.22 gFD). All image generators are trained on GPIC for one epoch at 256×256 resolution. CFG scales for each model are reported in the Appendix.

Model	Language	Vision	gFD ↓	Precision ↑	Recall ↑	Density → 1	Coverage ↑	MMD ↓
LlamaGen-XL	1.22B	0.78B	147.17	0.917	0.705	0.930	0.830	0.173
LLamaGen-XL/8	1.22B	0.90B	63.52	0.947	0.826	1.040	0.926	0.042
LLamaGen-XL-T5-Base/8	0.11B	0.90B	66.24	0.944	0.821	1.054	0.924	0.043
JiT-T2I-XXL/16	1.70B	1.10B	68.67	0.935	0.793	0.982	0.915	0.052
JiT-T2I-XXL/16 ++	1.70B	1.10B	55.70	0.948	0.848	1.034	0.938	0.025
JiT-T2I-XXL/8 ++	1.70B	1.10B	43.22	0.948	0.883	1.036	0.945	0.017
Xanthus	0.10B		144.16	0.924	0.636	1.029	0.853	0.131
	0.25B		72.55	0.946	0.796	1.050	0.922	0.050
	0.52B		48.02	0.953	0.858	1.064	0.938	0.027
	1.60B		35.28	0.955	0.891	1.071	0.955	0.019
	4.00B		28.43	0.951	0.905	1.034	0.960	0.014

4.2 Results: Comparison with Late-fusion T2I Baselines

We report quantitative results in Table 4 and show samples in Figure 7 and Appendix D. Figure 1 compares generation quality as a function of total pretraining FLOPs, with the exact compute breakdown reported in Table 11. Below, we discuss the key empirical findings:

Effect of image representation and perceptual supervision. For LlamaGen-XL, replacing its original image tokenizer with our GPIC-trained tokenizer increases the image sequence length from 256 to 1,024 and reduces gFD (147.2 → 63.5). For JiT-T2I-XXL/16, adding LPIPS perceptual supervision reduces gFD (68.7 → 55.7). Reducing the image patch size from 16×16 to 8×8 further increases the image sequence length from 256 to 1,024 and reduces gFD (55.7 → 43.2). We use these modified configurations for the subsequent comparisons with `Xanthus`.

Effect of language encoder size. We study whether a larger pretrained language encoder improves T2I generation. For LlamaGen, replacing FLAN-T5-XL with FLAN-T5-base reduces the language encoder size from 1.22B to 0.11B while retaining the same image tokenizer and 0.90B image generator. The resulting gFD changes only marginally (63.5 → 66.2), despite a 13.5× reduction in language-model training compute[2]. We next investigate whether strong T2I generation can be achieved without language pretraining altogether.

T2I generation without language pretraining. We now address the central question of this work: can a T2I model trained entirely from scratch match or surpass strong late-fusion systems that rely on pretrained language models? `Xanthus`-0.52B, a single multimodal model trained entirely from scratch, achieves 48.0 gFD, outperforming LlamaGen-XL/8 with pretrained FLAN-T5-XL (63.5) and approaching JiT-T2I-XXL/8++ with pretrained Qwen3-1.7B (43.2). Scaling `Xanthus` to 1.6B further reduces gFD to 35.3, surpassing both late-fusion methods. At 4B parameters, gFD decreases to 28.4. `Xanthus`-1.6B achieves lower gFD than both baselines while using 275× less total pretraining compute than JiT-T2I-XXL/8++.

> **Takeaway.** `Xanthus` demonstrates that T2I generative models can be trained entirely from scratch and outperform strong late-fusion methods, while requiring 275× less total pretraining compute than the strongest JiT-T2I baseline.

4.3 Results: Pretraining Data Composition and Scheduling

Q1. Does external language pretraining improve T2I generation?

No. Under a fixed pretraining budget, allocating α=0.1, 0.2, 0.5 of training to text-only data worsens gFD from 48.0 to 51.6, 52.8, and 65.1 (Table 5); both gFD and conditional image NLL degrade as paired image–text training shrinks. Even when language pretraining is *added* on top of a full T2I epoch (1.5× compute), gFD doesn't improve (48.0 vs. 49.2).

Q2. How do I2T supervision and scheduling affect T2I generation?

Staging favors generation; mixing favors captioning. We report results in Table 6. At small I2T fractions (0.1), scheduling barely helps (48.4 vs. 48.2 gFD). At 0.5, both schedules halve conditional image tokens (92.2B → 46.1B), but staged training substantially outperforms mixing on T2I (54.2 vs. 76.0 gFD), while mixing learns captioning better (conditional text NLL 1.009 vs. 20.437).

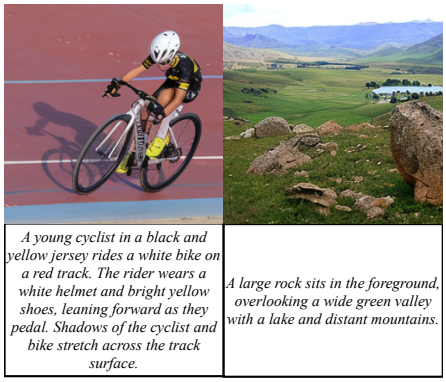

Figure 5: **Image captioning with `Xanthus`.** Examples from `Xanthus`-4B trained with I2T supervision. The same model performs both T2I generation and image captioning.

[2]Pretraining and finetuning the full FLAN-T5-XL encoder–decoder model requires 9.06×10^{21} FLOPs vs. 6.69×10^{20} FLOPs for FLAN-T5-base (Chung et al., 2024)

Table 5: **Effect of language pretraining on `Xanthus-0.52B`.** Under a fixed pretraining budget, allocating more compute to external text-only pretraining consistently worsens T2I generation. The † row adds $0.5\times$ the baseline budget of language pretraining before a full T2I epoch, using $1.5\times$ total pretraining compute, but does not improve gFD. Token counts exclude padding; NLL and gFD are evaluated on GPIC Test-50K.

		Text tokens (B)		Image tokens (B)		NLL ↓				
Configuration	α	$\mathcal{T} \mid \varnothing$	$\mathcal{T} \mid \mathcal{I}$	$\mathcal{I} \mid \varnothing$	$\mathcal{I} \mid \mathcal{T}$	$\mathcal{T} \mid \varnothing$	$\mathcal{T} \mid \mathcal{I}$	$\mathcal{I} \mid \varnothing$	$\mathcal{I} \mid \mathcal{T}$	gFD ↓
Baseline	–	4.05	0	10.24	92.16	**1.433**	**18.414**	**8.428**	**8.366**	**48.02**
Staged: Language → T2I	0.1	16.44	0	9.22	82.94	1.447	19.098	8.442	8.382	51.56
	0.2	28.83	0	8.19	73.73	1.451	17.197	8.445	8.388	52.76
	0.5	65.80	0	5.12	46.08	1.454	22.850	8.505	8.451	65.13
Staged: Language → T2I†	0.5	68.03	0	10.24	92.16	1.445	18.809	**8.428**	8.371	49.22

Table 6: **Effect of I2T supervision and scheduling on `Xanthus-0.52B`.** At a fixed I2T fraction, staged and mixed training use identical token counts and differ only in scheduling. Staged training better preserves T2I generation, while mixing improves conditional text NLL. Token counts exclude padding; NLL and gFD are evaluated on GPIC Test-50K.

		Text tokens (B)		Image tokens (B)		NLL ↓				
Configuration	β (or η)	$\mathcal{T} \mid \varnothing$	$\mathcal{T} \mid \mathcal{I}$	$\mathcal{I} \mid \varnothing$	$\mathcal{I} \mid \mathcal{T}$	$\mathcal{T} \mid \varnothing$	$\mathcal{T} \mid \mathcal{I}$	$\mathcal{I} \mid \varnothing$	$\mathcal{I} \mid \mathcal{T}$	gFD ↓
Baseline	–	4.05	0	10.24	92.16	**1.433**	18.414	8.428	8.366	**48.02**
Staged: I2T → T2I	$\beta = 0.1$	3.65	0.45	19.46	82.94	1.440	18.415	8.426	8.366	48.35
	$\beta = 0.5$	2.03	2.25	56.32	46.08	1.435	20.437	8.421	8.370	54.22
Mixing: T2I + I2T	$\eta = 0.1$	3.65	0.45	19.46	82.94	1.443	1.125	**8.414**	**8.362**	48.18
	$\eta = 0.5$	2.03	2.25	56.32	46.08	1.475	**1.009**	8.453	8.417	75.97

Q3. Can I2T supervision replace caption dropout?

Yes. We set $\eta{=}p_{\text{drop}}{=}0.1$ and disable caption dropout. We report results in Table 7. Since I2T sequences share the $[\texttt{SOV}]\, x\, [\texttt{EOV}]$ prefix, this preserves conditional and unconditional image-token counts while adding captioning supervision. Across all models, conditional text NLL drops by over an order of magnitude (e.g., 18.41 → 1.10 at 0.52B, Figure 5), while gFD (Table 7) and text–image alignment (Table 12) remain comparable.

4.4 Analysis: Classifier-Free Guidance and Caption Length

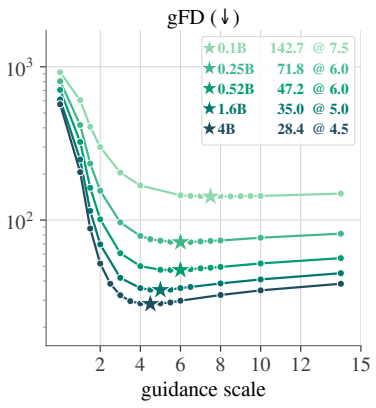

Figure 6: CFG sweep on GPIC validation. The optimal scale drops from 7.5 (0.1B) to 4.5 (4B) as gFD improves from 142.7 to 28.4; these scales are used on Test-50K.

Model size vs. classifier-free guidance (CFG). We evaluate the effect of CFG on T2I generation across all five `Xanthus` model sizes, using guidance scales from 1 to 14 (Figure 6). As model size increases from 0.1B to 4B, the guidance scale achieving the lowest gFD decreases from 7.5 to 4.5, while gFD improves from 142.7 to 28.4. This trend suggests the possibility that larger models may require no CFG (CFG = 1), eliminating the unconditional forward pass during inference.

Effect of caption length on T2I generation. In `Xanthus`, captions are used as a prefix to condition image generation. We investigate how T2I generation quality varies with caption length and model size. We evaluate all five model sizes using short, medium, and long captions from GPIC Test-1M (Table 8). As model size increases from 0.1B to 4B, gFD decreases across all caption types. Medium and

Table 7: **Replacing caption dropout with I2T supervision across model sizes.** Replacing caption-dropped T2I examples with I2T preserves the same pretraining budget and conditional/unconditional image-token counts. It significantly improves conditional text NLL while maintaining comparable T2I generation across model sizes.

			NLL ↓				T2I generation					
Size	η	p_{drop}	$\mathcal{T} \mid \varnothing$	$\mathcal{T} \mid \mathcal{I}$	$\mathcal{I} \mid \varnothing$	$\mathcal{I} \mid \mathcal{T}$	gFD ↓	Prec.↑	Rec.↑	Den.→ 1	Cov.↑	MMD↓
0.1B	–	0.1	1.601	19.066	8.774	8.727	144.16	0.924	**0.636**	**1.029**	**0.853**	**0.131**
	0.1	0	**1.598**	**1.410**	**8.771**	**8.724**	**144.14**	**0.926**	0.631	1.057	0.841	0.132
0.25B	–	0.1	1.497	21.472	8.566	8.509	72.55	0.946	**0.796**	**1.050**	0.922	**0.050**
	0.1	0	**1.491**	**1.218**	**8.559**	**8.503**	**70.44**	**0.947**	0.793	1.082	**0.928**	**0.050**
0.52B	–	0.1	**1.433**	18.414	8.428	8.366	48.02	0.953	0.858	**1.064**	0.938	**0.027**
	0.1	0	1.435	**1.102**	**8.419**	**8.358**	**46.48**	**0.956**	**0.860**	1.074	**0.944**	0.028
1.6B	–	0.1	**1.370**	20.426	8.303	8.235	**35.28**	**0.955**	0.891	1.071	**0.955**	**0.019**
	0.1	0	1.372	**0.977**	**8.292**	**8.228**	35.91	**0.955**	**0.893**	**1.052**	**0.955**	0.020
4B	–	0.1	1.338	16.738	**8.193**	**8.122**	**28.43**	0.951	**0.905**	**1.034**	0.960	**0.014**
	0.1	0	**1.331**	**0.898**	8.204	8.135	30.35	**0.957**	0.903	1.073	**0.964**	0.016

long captions consistently achieve lower gFD than short captions, with comparable performance between medium and long captions from 0.52B onward. Thus, T2I generation improves with model size across caption lengths, while more detailed text conditioning yields better generation quality than shorter captions.

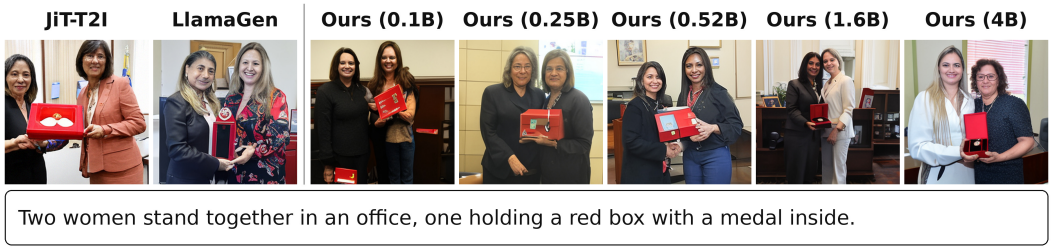

Figure 7: **Comparison of generated images.** From left to right: JiT-T2I-XXL/8++, LlamaGen-XL/8, and `Xanthus` models from 0.1B to 4B parameters. Image quality improves as `Xanthus` scales up. Additional samples are provided in Appendix D.

5 Discussion and Conclusion

We show that strong text-to-image generation does not require internet-scale language pretraining. `Xanthus` is pretrained entirely from scratch on paired image–text data, yet outperforms strong late-fusion baselines while using substantially less total pretraining compute; `Xanthus`-1.6B surpasses the strongest JiT-T2I baseline with 275× less pretraining compute. Our controlled data composition experiments further show that using external text-only pretraining does not improve T2I generation under the evaluated budgets, while replacing caption-dropped T2I examples with I2T supervision enables image captioning with comparable generation quality under the same budget. These results suggest that visually grounded language supervision from image–text pairs can be sufficient for learning strong text conditioning, and that internet-scale language pretraining need not be treated as a prerequisite for T2I generation. Our study is limited to GPIC, 256×256 generation, and the model families evaluated here; extending these findings to higher resolutions and larger-scale training is a valuable direction for future work.

Table 8: **Effect of caption length on T2I generation.** We report gFD for `Xanthus` across model sizes using 50K examples with short, medium, and long captions from GPIC Test-1M. Test-50K results are included for comparison.

Model size	Test-50K	Short	Medium	Long
0.1B	144.16	149.03	139.27	147.85
0.25B	72.55	77.18	70.14	72.53
0.52B	48.02	51.99	46.12	46.83
1.6B	35.28	38.59	34.21	34.33
4B	28.43	31.48	27.91	27.77

AI USE STATEMENT

In this work, we used generative AI tools to assist with implementing methods. We did not use generative AI tools to propose or refine hypotheses, design or provide feedback on research methodology or experiments, support qualitative or thematic data analysis, or interpret results. Generating synthetic datasets, developing theoretical models or conceptual frameworks, formulating mathematical claims, providing critical ingredients for proving mathematical claims, assisting in the writing of proofs, assisting with translation, and cleaning or reformatting datasets were not applicable to our work. Additionally, we used generative AI tools to create or edit software code, search for and identify relevant literature and information, draft and edit parts of the manuscript, and improve readability. We reviewed all AI-assisted work. AI-generated or edited code was inspected and tested for correctness, and information and references suggested by AI tools were verified against the original sources. We take responsibility for the final content of this work, including text, claims, code, and artifacts produced with the aid of generative AI.

ETHICS STATEMENT

Our work studies text-to-image generative modeling, which can be used to generate realistic visual content and may therefore enable misuse. By providing a significantly lower-cost recipe for training strong T2I models without internet-scale language pretraining, our work may also lower the barrier to developing capable generative systems, including for malicious use. `Xanthus` and the baseline models are pretrained on GPIC and may inherit biases or limitations present in its images and captions. Our work does not involve human-subject studies or the collection of a new dataset. We do not identify additional ethical risks specific to our method beyond those associated with visual generative models.

REPRODUCIBILITY STATEMENT

We provide the model architecture, tokenization, training objective, data composition and scheduling, baseline modifications, evaluation protocol, and compute accounting in the main paper, with full architecture, optimization, sampling, and guidance details in the Appendix. GPIC and the evaluation protocol used in this work are publicly available. Our code and model checkpoints will be publicly released.

REFERENCES

Niket Agarwal, Arslan Ali, Maciej Bala, Yogesh Balaji, Erik Barker, Tiffany Cai, Prithvijit Chattopadhyay, Yongxin Chen, Yin Cui, Yifan Ding, et al. Cosmos world foundation model platform for physical ai. *arXiv preprint arXiv:2501.03575*, 2025.

Black Forest Labs. FLUX.2. `https://github.com/black-forest-labs/flux2`, 2025.

Keshigeyan Chandrasegaran, Kyle Sargent, Suchir Agarwal, Michael Jang, Michael Poli, Juan Carlos Niebles, Justin Johnson, Jiajun Wu, and Li Fei-Fei. Gpic: A giant permissive image corpus for visual generation, 2026. URL `https://arxiv.org/abs/2605.30341`.

Junsong Chen, Jincheng Yu, Chongjian Ge, Lewei Yao, Enze Xie, Zhongdao Wang, James Kwok, Ping Luo, Huchuan Lu, and Zhenguo Li. Pixart-α: Fast training of diffusion transformer for photorealistic text-to-image synthesis. In *International conference on learning representations*, volume 2024, pp. 57611–57640, 2024.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. Scaling instruction-finetuned language models. *JMLR*, 25(70):1–53, 2024.

Shizhe Diao, Yu Yang, Yonggan Fu, Xin Dong, Dan Su, Markus Kliegl, Zijia Chen, Peter Belcak, Yoshi Suhara, Hongxu Yin, et al. Nemotron-climb: Clustering-based iterative data mixture bootstrapping for language model pre-training. *Advances in Neural Information Processing Systems*, 38, 2026.

Ming Ding, Zhuoyi Yang, Wenyi Hong, Wendi Zheng, Chang Zhou, Da Yin, Junyang Lin, Xu Zou, Zhou Shao, Hongxia Yang, et al. Cogview: Mastering text-to-image generation via transformers. *Advances in neural information processing systems*, 34:19822–19835, 2021.

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

Shengding Hu, Yuge Tu, Xu Han, Ganqu Cui, Chaoqun He, Weilin Zhao, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Xinrong Zhang, Zhen Leng Thai, Chongyi Wang, Yuan Yao, Chenyang Zhao, Jie Zhou, Jie Cai, Zhongwu Zhai, Ning Ding, Chao Jia, Guoyang Zeng, dahai li, Zhiyuan Liu, and Maosong Sun. MiniCPM: Unveiling the potential of small language models with scalable training strategies. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=3X2L2TFr0f.

Tianhong Li and Kaiming He. Back to basics: Let denoising generative models denoise. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 36115–36125, 2026.

Zhiheng Liu, Weiming Ren, Xiaoke Huang, Shoufa Chen, Tianhong Li, Mengzhao Chen, Yatai Ji, Sen He, Jonas Schult, Belinda Zeng, et al. Tuna-2: Pixel embeddings beat vision encoders for multimodal understanding and generation. *arXiv preprint arXiv:2604.24763*, 2026.

Zehong Ma, Ruihan Xu, and Shiliang Zhang. Pixelgen: Improving pixel diffusion with perceptual supervision. *arXiv preprint arXiv:2602.02493*, 2026.

Fabian Mentzer, David Minnen, Eirikur Agustsson, and Michael Tschannen. Finite scalar quantization: Vq-vae made simple. In *International Conference on Learning Representations*, 2024.

Alexander Quinn Nichol, Prafulla Dhariwal, Aditya Ramesh, Pranav Shyam, Pamela Mishkin, Bob Mcgrew, Ilya Sutskever, and Mark Chen. GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models. In Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (eds.), *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 16784–16804. PMLR, 17–23 Jul 2022. URL https://proceedings.mlr.press/v162/nichol22a.html.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *International conference on machine learning*, pp. 8821–8831. Pmlr, 2021.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in neural information processing systems*, 35:36479–36494, 2022.

Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *ACL*, 2016.

Peize Sun, Yi Jiang, Shoufa Chen, Shilong Zhang, Bingyue Peng, Ping Luo, and Zehuan Yuan. Autoregressive model beats diffusion: Llama for scalable image generation. *arXiv preprint arXiv:2406.06525*, 2024.

Chameleon Team, Mingda Chen, Jacob Kahn, and Shang-Wen Li. Chameleon: Mixed-modal early-fusion foundation models. *ArXiv*, abs/2405.09818, 2024. URL https://api.semanticscholar.org/CorpusID:269791516.

Xinlong Wang, Xiaosong Zhang, Zhengxiong Luo, Quan Sun, Yufeng Cui, Jinsheng Wang, Fan Zhang, Yueze Wang, Zhen Li, Qiying Yu, et al. Emu3: Next-token prediction is all you need. *arXiv preprint arXiv:2409.18869*, 2024.

Enze Xie, Junsong Chen, Junyu Chen, Han Cai, Haotian Tang, Yujun Lin, Zhekai Zhang, Muyang Li, Ligeng Zhu, Yao Lu, and Song Han. Sana: Efficient high-resolution image synthesis with linear diffusion transformer, 2025. URL https://arxiv.org/abs/2410.10629.

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.

Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan, Alexander Ku, Yinfei Yang, Burcu Karagol Ayan, et al. Scaling autoregressive models for content-rich text-to-image generation. *TMLR*, 2022.

Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 586–595. IEEE, 2018.

Bing Zhao, Chenfei Wu, Deqing Li, Hao Meng, Jiahao Li, Jie Zhang, Jingren Zhou, Junyang Lin, Kaiyuan Gao, Kuan Cao, et al. Qwen-image-2.0 technical report. *arXiv preprint arXiv:2605.10730*, 2026.

Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and diffuse images with one multi-modal model. In *International Conference on Learning Representations*, volume 2025, pp. 6446–6469, 2025.

APPENDIX

A Training Details **13**

B Sampling Details **13**

C Additional Evaluation Metrics **14**

D Samples **15**

A TRAINING DETAILS

We provide model architecture and training details for all `Xanthus` models in Table 9.

Table 9: **`Xanthus` configurations.**

	0.1B	0.25B	0.52B	1.6B	4B
architecture					
backbone	Qwen3 (Yang et al., 2025), decoder-only				
layers	14	21	28	28	36
hidden dim	512	768	1024	2048	2560
FFN dim	1536	2304	3072	6144	9728
query / KV heads	8 / 4	12 / 6	16 / 8	16 / 8	32 / 8
head dim	128				
non-embedding params	55.1M	185.8M	440.4M	1409.3M	3634M
embedding params	41.3M	61.9M	82.6M	165.2M	206.4M
total params	96.4M	247.7M	523.0M	1574.5M	3840M
vocabulary	80,640 (16,384 text + 64,000 image + 256 special)				
tied embeddings	yes				
image size	256				
image tokens	1,024				
max caption tokens	254				
sequence length	1,280				
training					
training steps	97,632 (1 epoch on GPIC)				
global batch size	1,024				
warmup steps	1,000				
peak learning rate	1e-3	7.5e-4	7.5e-4	7.5e-4	7.5e-4
lr schedule	linear warmup, constant to step 49,595, cooldown to peak/10				
optimizer	AdamW, $\beta_1, \beta_2 = 0.9, 0.95$				
weight decay	0.1 (0 on embeddings)				
gradient clipping	1.0				
init std	0.02				
image loss weight λ_{img}	0.5				
caption dropout p_{drop}	0.1				
precision	bf16				

B SAMPLING DETAILS

All models generate one 256×256 image per caption for the 50K GPIC Test-50K captions. **Xanthus.** We sample the 1,024 image tokens autoregressively after the caption prefix and `[SOV]`, using temperature 1.0 with top-k and top-p disabled and a fixed random seed. CFG is applied only to image-token logits, $\ell_{\text{cfg}} = \ell_u + \gamma(\ell_c - \ell_u)$, where the unconditional branch uses the `[SOV]` prefix without the caption. For each model size, we select γ to minimize gFD on the GPIC validation set (Figure 6) and use the same scale for all configurations of that size (Tables 5–7 and 12). **Baselines.** For LlamaGen and JiT-T2I, we use the official sampling code and settings and use guidance scales of 6.0 and 6.25, respectively.

Table 10: **Tokenizer training configurations.**

Image tokenizer	
architecture	Cosmos (Agarwal et al., 2025) encoder–decoder, FSQ (Mentzer et al., 2024)
decoder width	192 (128 in Cosmos)
spatial downsampling	$8\times$
codebook size	64,000 (FSQ levels $[8, 8, 8, 5, 5, 5]$)
tokens per image	1,024 (32×32, raster order)
losses (stage 1)	reconstruction (1.0) + LPIPS (0.1)
losses (stage 2)	+ PatchGAN adversarial (5e-3)
training data	GPIC train images, 1.5 epochs (1.0 stage 1 / 0.5 stage 2)
iterations	242,172
batch size	640
learning rate	1e-4, constant
optimizer	Adam, $\beta_1, \beta_2 = 0.9, 0.999$
weight decay	0.001
gradient clipping	1.0
EMA	yes
hardware	$1\times$ 8×H100 node
Text tokenizer	
algorithm	byte-level BPE (Sennrich et al., 2016)
vocabulary size	16,384
training data	GPIC train captions (21.3 GB) + OpenWebText (11.9 GB)
caption length	mean 44.5, p50 36, p90 64, p99 182 tokens

Table 11: **Total pretraining compute of T2I systems.** We report total pretraining FLOPs, decomposed into language, image-generator, and image-tokenizer compute, for all configurations shown in Fig. 1. Vision compute includes both image-generator and image-tokenizer pretraining. For late-fusion methods, language FLOPs include pretraining of the language model. For `Xanthus`, language FLOPs correspond to the text-token portion of joint multimodal pretraining, without external language pretraining. All image generators are pretrained on GPIC for one epoch at 256×256 resolution. Generation Fréchet distance (gFD) is evaluated on GPIC Test-50K; lower is better.

		Language		Vision			
Method	Total FLOPs	FLOPs	(%)	Generator FLOPs	Tokenizer FLOPs	(%)	gFD ↓
LlamaGen-XL	9.32×10^{21}	9.06×10^{21}	97.17	1.19×10^{20}	1.45×10^{20}	2.83	147.17
LlamaGen-XL/8	9.68×10^{21}	9.06×10^{21}	93.58	4.76×10^{20}	1.45×10^{20}	6.42	63.52
LlamaGen-XL-T5-Base/8	1.29×10^{21}	6.69×10^{20}	51.87	4.76×10^{20}	1.45×10^{20}	48.13	66.24
JiT-T2I-XXL/16	3.67×10^{23}	3.67×10^{23}	99.95	1.69×10^{20}	0	0.05	68.67
JiT-T2I-XXL/16++	3.67×10^{23}	3.67×10^{23}	99.95	1.69×10^{20}	0	0.05	55.70
JiT-T2I-XXL/8++	3.68×10^{23}	3.67×10^{23}	99.82	6.76×10^{20}	0	0.18	43.22
`Xanthus`-0.1B	2.05×10^{20}	1.47×10^{19}	7.15	5.93×10^{19}	1.31×10^{20}	92.85	144.16
`Xanthus`-0.25B	3.21×10^{20}	3.78×10^{19}	11.74	1.53×10^{20}	1.31×10^{20}	88.26	72.55
`Xanthus`-0.52B	5.33×10^{20}	7.97×10^{19}	14.96	3.22×10^{20}	1.31×10^{20}	85.04	48.02
`Xanthus`-1.6B	1.34×10^{21}	2.40×10^{20}	17.90	9.69×10^{20}	1.31×10^{20}	82.10	35.28
`Xanthus`-4B	3.08×10^{21}	5.85×10^{20}	19.00	2.36×10^{21}	1.31×10^{20}	81.00	28.43

C Additional Evaluation Metrics

Text–image alignment. We measure prompt adherence with Qwen3-VL-Reranker-8B, a multimodal reranker trained to judge whether a document satisfies a query. For each of the 50K GPIC test captions, we generate one image, then score the (caption, image) pair with the caption as the query and the generated image as the document. We report the average score over all 50K pairs (higher is better). Each model is sampled at its gFD-optimal guidance scale with top-k disabled.

Table 12: **Text–image alignment across model sizes.** We compare the same two `Xanthus` configurations as in Table 7: **Baseline** trains exclusively on T2I examples with caption dropout ($p_{\text{drop}} = 0.1$); **Mixing** replaces caption-dropped T2I examples with I2T supervision ($\eta = 0.1$, $p_{\text{drop}} = 0$). We report the mean Qwen3-VL-Reranker-8B alignment score over 50K GPIC test captions; higher is better.

Size	Baseline	Mixing: T2I + I2T
0.1B	0.801	**0.806**
0.25B	0.840	**0.849**
0.52B	0.870	**0.872**
1.6B	**0.886**	**0.886**
4B	**0.893**	0.890

D SAMPLES

We compare generated samples across the four GPIC caption types: tag (Figure 8), short (Figure 9), medium (Figure 10), and long (Figure 11). Each row shows JiT-T2I-XXL/8++, LlamaGen-XL/8, and Xanthus from 0.1B to 4B for the same caption, sampled with the settings in Appendix B. Sample quality and prompt adherence improve as `Xanthus` scales, with larger models capturing finer details such as clothing, object attributes, and scene layout. Rendering legible text remains challenging for all models at 256×256 resolution.

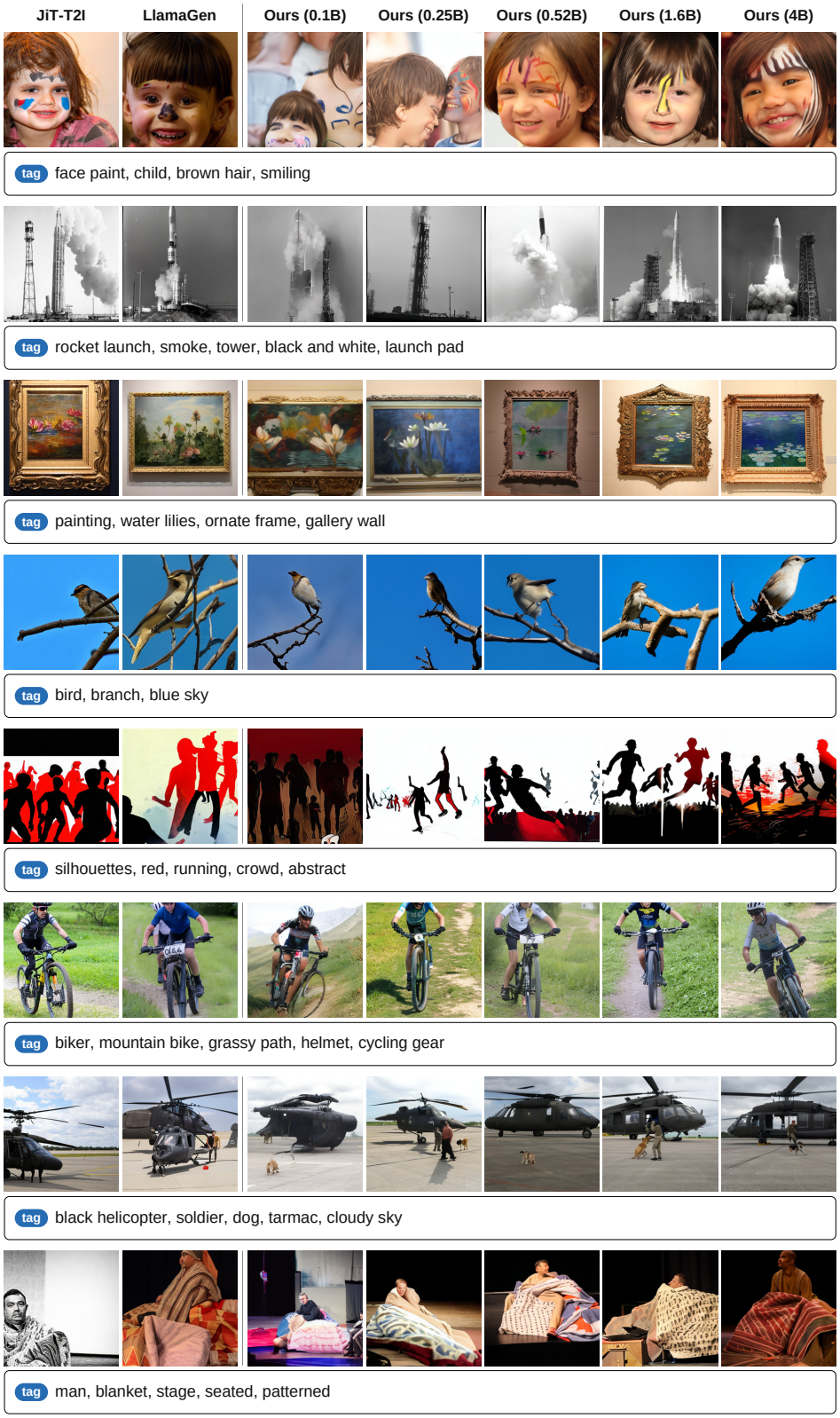

Figure 8: **Comparison of generated images with "tag" captions.** From left to right: JiT-T2I-XXL/8++, LlamaGen-XL/8, and ours from 0.1B to 4B.

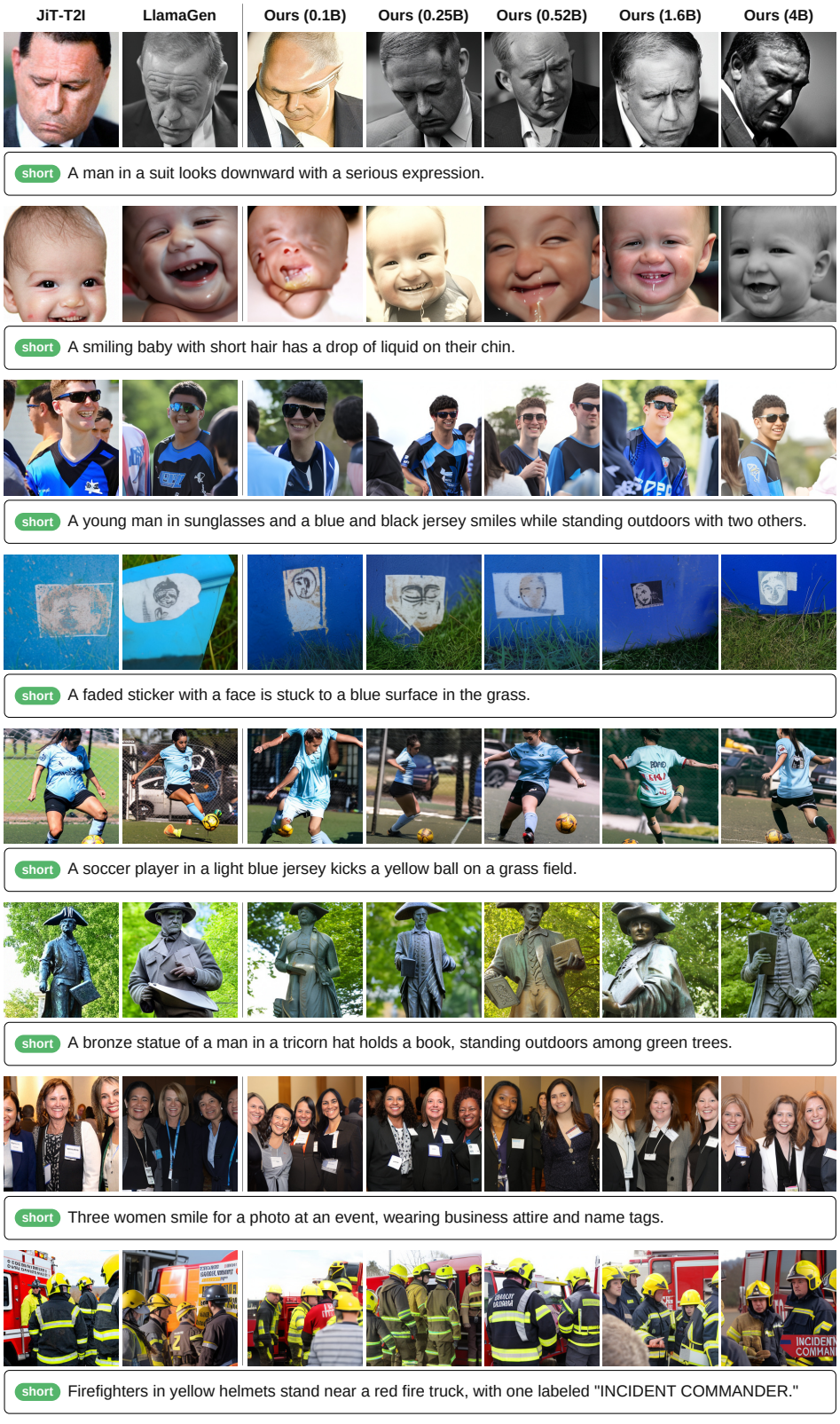

Figure 9: **Comparison of generated images with "short" captions.** From left to right: JiT-T2I-XXL/8++, LlamaGen-XL/8, and ours from 0.1B to 4B.



Figure 10: **Comparison of generated images with "medium" captions.** From left to right: JiT-T2I-XXL/8++, LlamaGen-XL/8, and ours from 0.1B to 4B.

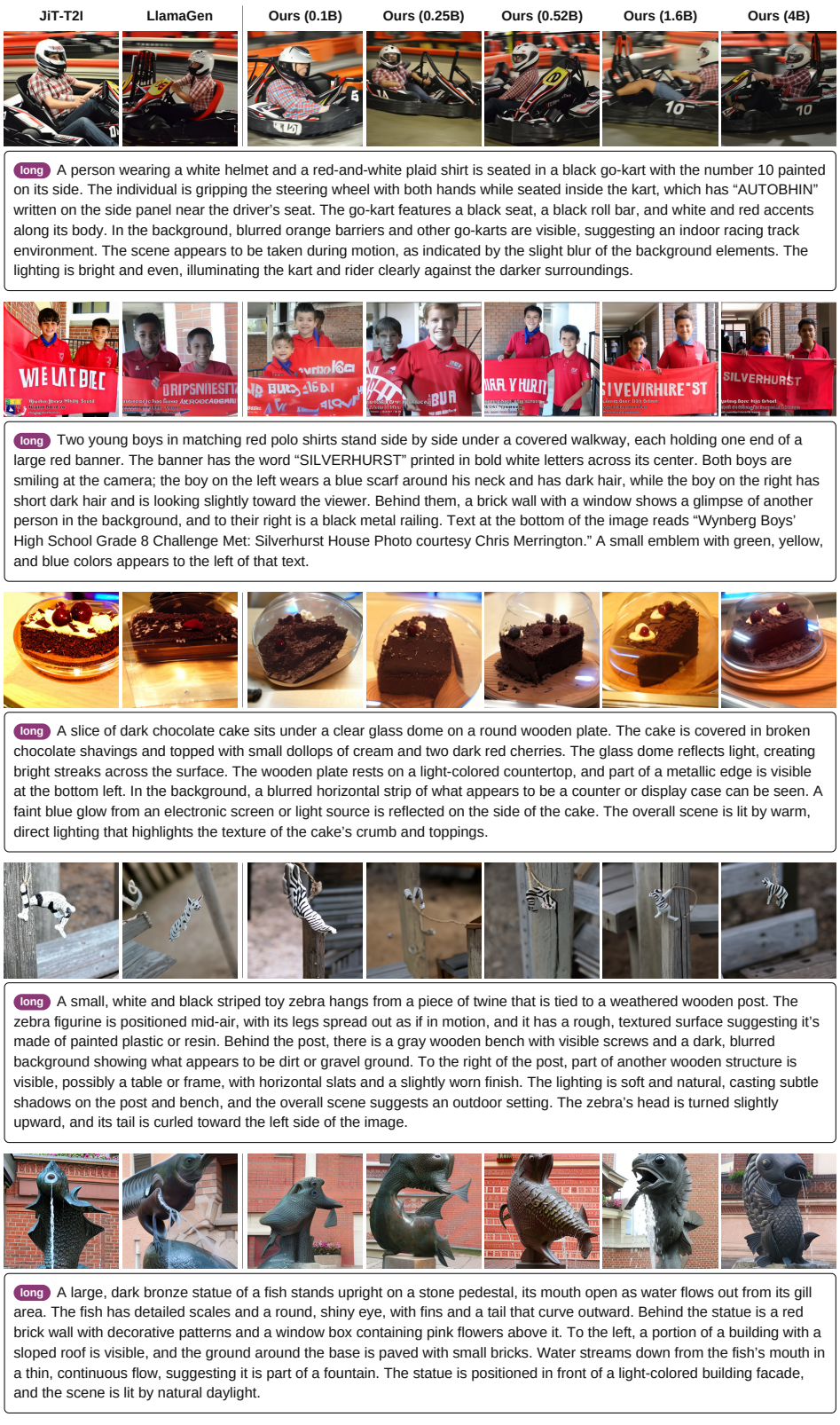

Figure 11: **Comparison of generated images with “long” captions.** From left to right: JiT-T2I-XXL/8++, LlamaGen-XL/8, and ours from 0.1B to 4B.