

# SCALING AUTOREGRESSIVE TEXT-TO-IMAGE GENERATIVE MODELS WITH CONTINUOUS TOKENS

Anonymous authors

Paper under double-blind review



Figure 1: Samples from our Fluid 10.5B autoregressive model with continuous tokens. Please see the appendix for the text prompts to generate these images.

## ABSTRACT

Scaling up autoregressive models in vision has not proven as beneficial as in large language models. In this work, we investigate this scaling problem in the context of text-to-image generation, focusing on two critical factors: whether models use discrete or continuous tokens, and whether tokens are generated in a random or fixed raster order using BERT- or GPT-like transformer architectures. Our empirical results show that, while all models scale effectively in terms of validation loss, their evaluation performance—measured by FID, GenEval score, and visual quality—follows different trends. Models based on continuous tokens achieves significantly better visual quality than those using discrete tokens. Furthermore, the generation order and attention mechanisms significantly affect the GenEval score: random-order models achieve notably better GenEval scores compared to raster-order models. Inspired by these findings, we train Fluid, a random-order autoregressive model on continuous tokens. Fluid 10.5B model achieves a new state-of-the-art zero-shot FID of 6.16 on MS-COCO 30K, and 0.69 overall score on the GenEval benchmark. We hope our findings and results will encourage future efforts to further bridge the scaling gap between vision and language models.

## 1 INTRODUCTION

Scaling laws underpin the unprecedented success of large language models (LLMs). Empirically, increasing the number of parameters in autoregressive models consistently leads to significant performance improvements and the emergence of new capabilities in natural language processing (NLP) tasks (Devlin et al., 2018; Radford et al., 2018; Brown et al., 2020; Kaplan et al., 2020; Wei et al., 2022). This empirical relationship has inspired numerous efforts to scale up language models, resulting in the development of many highly capable models (Bubeck et al., 2023; Team et al., 2023; Achiam et al., 2023).

Encouraged by this success, many attempts have been made to adopt and scale up autoregressive models in computer vision, particularly for generative tasks like text-to-image generation (Yu et al., 2021; 2023; Bai et al., 2024; El-Nouby et al., 2024). However, the performance and visual quality of content generated by these models often fall short compared to other generative models, such as diffusion models (Ho et al., 2020; Saharia et al., 2022; Rombach et al., 2022a; Esser et al., 2024), leaving it unclear whether similar scaling laws apply to the vision domain.

We propose several hypotheses for the performance gap. First, the vector quantization (VQ) (van den Oord et al., 2017) step, which is required for most visual autoregressive models, may introduce significant information loss, ultimately limiting model performance. Second, unlike the inherently sequential nature of language, generating visual content might benefit more from a different autoregressive prediction order. Third, there is often a confusion between two levels of generalizability when evaluating scaling laws in vision models: (a) generalization to new data using the same metric as the training loss (commonly referred to as validation loss), and (b) generalization to a new metric or problem different from the training objective, such as FID (Heusel et al., 2017), the GenEval benchmark (Ghosh et al., 2024), or visual quality. We hypothesize that power-law scaling (Kaplan et al., 2020) applies to (a) for autoregressive models on vision data, but not necessarily to (b).

To investigate these hypotheses, we conduct a comprehensive empirical study on the scaling behavior of autoregressive models in the context of text-to-image generation. Specifically, we explore two key factors: whether the model operates on continuous or discrete tokens, and whether tokens are generated in a random or fixed raster order. To this end, we utilize the Diffusion Loss (Li et al., 2024) to make autoregressive models compatible with continuous tokens. We generalize BERT-like vision model MaskGIT (Chang et al., 2022) as random-order autoregression, as it conceptually predicts output tokens in a randomized order while retaining the autoregressive nature of “predicting next tokens based on known ones”. We analyze the behavior of four autoregressive variants, each employing different combinations of these two factors. We scale their parameters from 150M to 3B and evaluate their performance using three metrics: validation loss, FID (Heusel et al., 2017), and GenEval score (Ghosh et al., 2024). We also inspect the visual quality of the generated images.

Our experiments indicate that VQ-based models, regardless of whether they use a random or fixed raster order, exhibit a *slower* improvement in FID scores when scaling model size compared to models operating on continuous tokens. VQ models also produce images of lower visual quality, likely due to information loss introduced by vector quantization.

Furthermore, the token generation order and the associated attention mechanism primarily influence the global structure of the generated image. In the GenEval benchmark, random-order models with bidirectional attention significantly outperform raster-order models with causal attention, particularly when generating multiple objects. Random-order models can readjust the global structure at every prediction step, whereas raster-order models cannot. This suggests that the token generation order plays a crucial role in achieving better text-to-image alignment.

Our experiments also demonstrate that validation loss scales as a power-law with model size, no matter whether the model operates with continuous or discrete tokens. This implies scalable behavior at the level of (a) generalization—generalizing to new data using the same metric as the training loss—which aligns with observations in language models (Kaplan et al., 2020). However, as for generalizing to a different metric, such as FID or GenEval score, although performance consistently improves with better validation loss, the trend may not follow a strict power-law.

Building on these findings, we scale the Fluid model, *i.e.*, random-order model with continuous tokens, up to 10.5B parameters and train it using the WebLI dataset (Chen et al., 2022). The resulting Fluid 10.5B model achieves a zero-shot FID of 6.16 on MS-COCO and a GenEval overall score of

0.69, comparing favorably with leading text-to-image generative models such as DALL-E 3 (Betker et al., 2023) and Stable Diffusion 3 (Esser et al., 2024). We hope that our empirical findings and positive results could shed light on the scaling behavior for text-to-image generation models and further innovation in this frontier.

## 2 RELATED WORK

**Text-to-image diffusion models.** The dominant approaches are based on diffusion models. Dall-E 2 (OpenAI, 2023), Imagen (Ho et al., 2022), and Stable Diffusion (Rombach et al., 2022a) revolutionized text-to-image generation. Most recently, MidJourney (Midjourney, 2023), SD v3 (Esser et al., 2024), and Imagen3 (Baldridge et al., 2024) can generate realistic images that are hard to tell from real images. However, generating samples is usually computationally expensive due to the multiple forward passes.

**Autoregressive (AR) models.** While being the defacto model for language modeling, AR models are lagging behind diffusion models for text-to-image generation. AR models (Yu et al., 2022; Chang et al., 2023; Yu et al., 2023) are often used together with discrete tokenizers (van den Oord et al., 2017; Esser et al., 2021), which often limit the modeling capability. For example, Parti (Yu et al., 2022) scales up the model to 20B, which obtains a slightly lower/better FID score of 7.23 on MS-COCO than 7.27 by the 3.4B diffusion-based Imagen. Here we find that, with a continuous tokenizer, our small 369M model can already achieve the same FID score as Parti.

Recently Li et al. (2024) challenges the conventional wisdom and replaces the discrete tokenizer with a continuous tokenizer via a diffusion loss. They introduced masked autoregressive (MAR) model obtains strong results for class-conditioning generation on ImageNet. However, scaling MAR models for text-to-image generation is unexplored. Here we empirically study the scaling behavior for MAR and report several findings important for the research community.

**Scaling language models.** Kaplan et al. (2020) empirically observed that, for language model, the validation loss scales as a power-law with model size, dataset size, and the amount of compute used for training. Hoffmann et al. (2022) discovered that contemporary LLMs are under-trained, and that for compute-optimal training, the model size and the number of training tokens should be scaled equally. Under the same compute budget while using 4x more data, their 70B Chinchilla outperforms much larger models, such as the 530B Megatron-Turing NLG (Smith et al., 2022). Wei et al. (2022) found that larger models have emergent abilities that are not present in smaller models. These observations have inspired significant efforts to scale up language models to trillions of parameters (Achiam et al., 2023; Team et al., 2023; Dubey et al., 2024).

**Scaling vision models.** Similar scaling law has been obscure for computer vision. For recognition models (Tan, 2019; Zhai et al., 2022; He et al., 2022; Dehghani et al., 2023), scaling often comes with diminishing returns. For example, Dehghani et al. (2023) scale ViT up from 3B to 22B but only observed 0.25% accuracy increase in ImageNet linear probing. The scaling of generative models is more promising - DiT (Peebles & Xie, 2023) shows consistent improvement in generation quality when scaling up compute and model size (despite only up to 600M). The follow-up Sora (OpenAI, 2024) further shows the potential to scale up for video generation. In this paper, we perform a comprehensive empirical study of AR models for text-to-image generation and our findings shed light on closing the gap in scaling behavior between vision and language generative models.

## 3 PRELIMINARY: AUTOREGRESSIVE IMAGE GENERATION

Given a sequence of tokens  $\{x^1, x^2, \dots, x^n\}$  where the superscript  $1 \leq i \leq n$  specifies an order, autoregressive models (Gregor et al., 2014; van den Oord et al., 2016b;a; Parmar et al., 2018; Chen et al., 2018; 2020) formulate the generation problem as “next token prediction”:

$$p(x^1, \dots, x^n) = \prod_{i=1}^n p(x^i | x^1, \dots, x^{i-1}). \quad (1)$$

Following the chain rule, the network is trained to model  $p(x^i | x^1, \dots, x^{i-1})$  and generate tokens iteratively. While all autoregressive models share this fundamental approach, differences in their

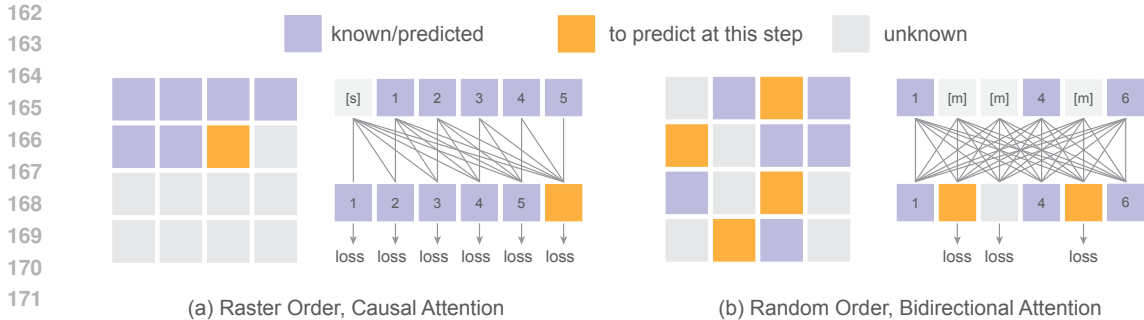


Figure 2: **Autoregressive models with different orders.** (a) A raster-order autoregressive model predicts one next token based on the known ones, implemented using a GPT-like transformer with causal attention. (b) A random-order autoregressive model predicts one or multiple tokens simultaneously given a random order, implemented using a BERT-like transformer with bidirectional attention.

design can affect the performance. Two key design choices are the representation of  $x$ , *i.e.*, discrete or continuous, and the generation order, which we elaborate below.

**Discrete vs. continuous tokens.** The goal of an autoregressive model is to estimate  $p(x^i | x^1, \dots, x^{i-1})$ . Traditionally, this is done by transforming the image into a set of discrete tokens with a finite vocabulary and then estimating a categorical distribution over the vocabulary. The training objective is to minimize the cross-entropy loss, and sampling can be efficiently performed using categorical sampling. Most autoregressive image generation models rely on this form of token discretization (Esser et al., 2021; Chang et al., 2022; Tian et al., 2024; Yu et al., 2022).

However, such discretization often leads to a significant loss of information from the image (Figure 3). Recent work (Li et al., 2024) has shown the possibility of applying a small diffusion model to approximate the distribution of each image token in a continuous fashion. This approach eliminates the need for vector quantization, and allows modeling images with continuous tokenizers with much better visual reconstruction quality. In this paper, we explore the scaling behavior of autoregressive image models on both discrete and continuous tokens.

**Raster Order + GPT vs. Random Order + BERT.** In autoregressive image generation, there are two primary generation orders: raster and random. As illustrated in Figure 2, raster order generates tokens sequentially from left to right, top to bottom. This fixed-order generation is well-suited for a GPT-like transformer architecture, which predicts the next token in a causal manner. In contrast, random order allows multiple tokens to be generated in each step. The selection of these tokens can either be completely random or based on a sampling mechanism that prioritizes tokens with higher predicted confidence scores (Chang et al. (2023); Li et al. (2024)).

Each generation order has its pros and cons. Raster order models with GPT-like transformer support fast inference via key-value (kv) caching. However, this causal structure can also introduce performance degradation. On the other hand, random order generation is usually achieved with a BERT-like bidirectional attention mechanism. While this approach prevents the usage of kv-caching, it enables the model to decode multiple tokens at each autoregressive step, allowing global editing. Despite their individual strengths, it remains unclear in the literature which generation order scales better for text-to-image generation tasks. In this work, we compare the performance and scaling behaviors of raster-order and random-order autoregressive models.

#### 4 IMPLEMENTATION

The overall framework of our text-to-image model training is straightforward. An image tokenizer first converts the original image into tokens. These tokens are then partially masked, and a transformer is trained to reconstruct the masked tokens conditioned on the text. Below, we provide a detailed description of each component of our framework.

216  
217  
218  
219  
220  
221  
222  
223  
224  
225  
226  
227  
228  
229  
230  
231  
232  
233  
234  
235  
236  
237  
238  
239  
240  
241  
242  
243  
244  
245  
246  
247  
248  
249  
250  
251  
252  
253  
254  
255  
256  
257  
258  
259  
260  
261  
262  
263  
264  
265  
266  
267  
268  
269

**Image Tokenizer.** We use a pre-trained image tokenizer to encode  $256 \times 256$  images into a token space. Such a tokenizer can be either discrete or continuous, facilitating different training objectives of the autoregressive model. In our experiments, the discrete tokenizer is an VQ-based model pre-trained on the WebLI dataset (Chen et al., 2022). We follow Muse (Chang et al., 2023) to encode each image into  $16 \times 16$  discrete tokens with a vocabulary size of 8192. For the continuous tokenizer, we adopt a widely-used one from Stable Diffusion (Rombach et al., 2022b), which encodes the image into  $32 \times 32$  continuous tokens, each containing 4 channels. To be consistent in sequence length with the discrete tokenizer, we further group each  $2 \times 2$  patch of continuous tokens into a single token, resulting in a final sequence length of 256, with each token containing 16 channels. As shown in Figure 3, the continuous tokenizer can achieve notably better reconstruction quality than the discrete one.

**Text Encoder.** The raw text (maximum length of 128) is tokenized by SentencePiece tokenization (SentencePiece, 2023), and embedded with a pre-trained T5-XXL encoder (Raffel et al., 2020), which has 4.7B parameters and is frozen during training. To further align the text embeddings for image generation, we add a small bidirectional text aligner consisting of six trainable transformer blocks on top of the T5 embeddings to extract the final text representation.

**Transformer.** After encoding the original image into a token sequence using an image tokenizer, we use a standard decoder-only transformer model (Vaswani et al., 2017) for autoregressive generation. Each block consists of three consecutive layers – self-attention, cross-attention, and MLP. The self-attention and MLP layers are only applied to visual tokens, while the cross attention layer takes visual and textual tokens as queries and keys, respectively. As shown in Figure 2, for raster-order models, the transformer predicts the next token based on previous tokens using causal attention for the self-attention block, similar to GPT. In random-order models, unknown tokens are masked by a learnable token, and the transformer predicts these masked tokens using bidirectional attention, similar to BERT.

**Output head.** For *discrete* tokens, we follow the common practice with autoregressive models. The outputs are transformed into categorical distributions by softmax following a linear layer, whose weights are reused from the input embedding layer. For *continuous* tokens, we apply a six layer light-weight MLP as the diffusion head (Li et al., 2024) to model the per-token distribution. The embedding dimension of this head is the same as the backbone transformer. The per-token diffusion process follows (Nichol & Dhariwal, 2021; Li et al., 2024). The noise schedule has a cosine shape, with 1000 steps at training time; at inference time, it is resampled to 100 steps.

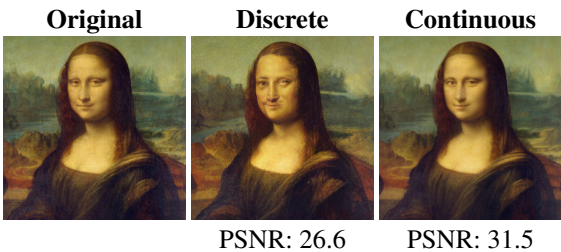


Figure 3: **Reconstruction quality of the tokenizers we used.** The discrete tokenizer is significantly worse than the continuous tokenizer.

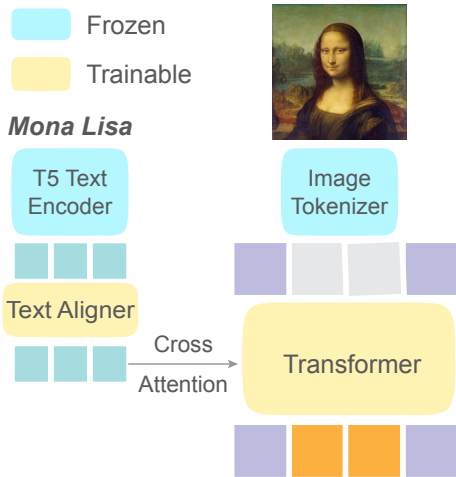


Figure 4: **Our text-to-image generation framework.** We use a pre-trained image tokenizer to convert the image into either discrete or continuous tokens. The corresponding text is embedded using a pre-trained T5 encoder, followed by a trainable text aligner. The transformer then takes cross-attention from the text embeddings to predict the missing tokens.

## 270 5 EXPERIMENTS

271  
272 **Dataset.** We use a subset of the WebLI (Web Language Image) dataset (Chen et al., 2022) as our  
273 training set, which consists of image-text pairs from the web with high scores for both image quality  
274 and alt-text relevance. By default, the images are center-cropped and resized to  $256 \times 256$ , and the  
275 text is the corresponding alt-text of each image.

276  
277 **Training.** Unless otherwise specified, we use the AdamW optimizer ( $\beta_1 = 0.9, \beta_2 = 0.95$ )  
278 (Loshchilov & Hutter, 2019) with a weight decay of 0.02 to train each model for 1M steps with  
279 a batch size of 2048. This is equivalent to approximately 3 epochs on our dataset. We employ  
280 a constant learning rate schedule with a 65K-step linear warmup and a maximum learning rate of  
281  $1 \times 10^{-4}$ . For training the random-order models, we randomly sample the masking ratio from  $[0, 1]$   
282 following a cosine schedule, similar to MaskGIT (Chang et al., 2022), to mask each image. For all  
283 models, exponential moving average of the weights are gathered by a decay rate of 0.9999 and then  
284 used for evaluation.

285 **Inference.** We follow the practices established by VQGAN (Esser et al., 2021), Muse (Chang  
286 et al., 2023), and MAR (Li et al., 2024) to generate images from text prompts. For random-order  
287 models, we use 64 steps for image generation with a cosine schedule (Chang et al., 2022). To further  
288 enhance generation performance, we apply temperature and classifier-free guidance, as is commonly  
289 practiced.

290  
291 **Evaluation.** We evaluate the scaling behavior of different autoregressive model variants both  
292 quantitatively and qualitatively. Quantitatively, we evaluate the validation loss on 30K images  
293 from the MS-COCO 2014 training set, as well as two widely-adopted metrics: zero-shot Frechet  
294 Inception Distance (FID) on MS-COCO, and the GenEval score (Ghosh et al., 2024). Inference  
295 hyper-parameters, such as temperature and classifier-free guidance, are optimized for each evalu-  
296 ation metric. FID is computed over 30K randomly selected image-text pairs from the MS-COCO  
297 2014 training set, providing a metric that evaluates both the fidelity and diversity of generated im-  
298 ages. The GenEval benchmark, on the other hand, measures the model’s ability to generate images  
299 that accurately reflect the given prompt. For qualitative evaluation, we generate images from several  
300 prompts using each model and compare the visual quality of the generated images.

### 301 5.1 SCALING BEHAVIORS

302  
303 In this section, we explore how two key design choices in autoregressive image generative models—  
304 token representation and generation order—affect performance and scaling behavior. We construct  
305 models with different combinations of these two design choices, resulting in four distinct variants  
306 of autoregressive image generation models. We also explore the generalizability of these models  
307 across different data and evaluation metrics. Our experiments reveal several intriguing properties.

308 **Validation losses consistently scale with model size.** In Figure 5, we examine the scaling behav-  
309 ior of the four autoregressive variants in terms of validation loss. We observe consistent reductions  
310 in validation loss as model size increases from 150 million to 3 billion parameters, which aligns  
311 with the findings in Henighan et al. (2020). This demonstrates that the improvements in training  
312 loss resulting from increased model size generalize well to validation loss on data different from the  
313 training data.

314  
315 **Continuous tokens and large models are crucial for visual quality.** In Figure 7, we compare the  
316 visual quality of images generated by the four autoregressive variants. The visual quality of models  
317 using discrete tokens is significantly worse than that of models using continuous tokens, *e.g.*, the  
318 eyes of the corgi is asymmetric for discrete token based models and scaling up can not solve this  
319 problem. This limitation is largely because of the discrete tokenizer, which introduces substantial  
320 information loss. For instance, even with 3B parameters, the discrete token models cannot generate  
321 an accurate Mona Lisa due to poor reconstruction quality of the tokenizer (Figure 3). In contrast,  
322 models with continuous tokens produce much higher-quality images.

323 Additionally, larger models show consistent improvements in both visual quality and image-text  
alignment. For example, a random-order model with 0.2B parameters struggles to generate “an

324  
325  
326  
327  
328  
329  
330  
331  
332  
333  
334  
335  
336  
337  
338  
339  
340  
341  
342  
343  
344  
345  
346  
347  
348  
349  
350  
351  
352  
353  
354  
355  
356  
357  
358  
359  
360  
361  
362  
363  
364  
365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377

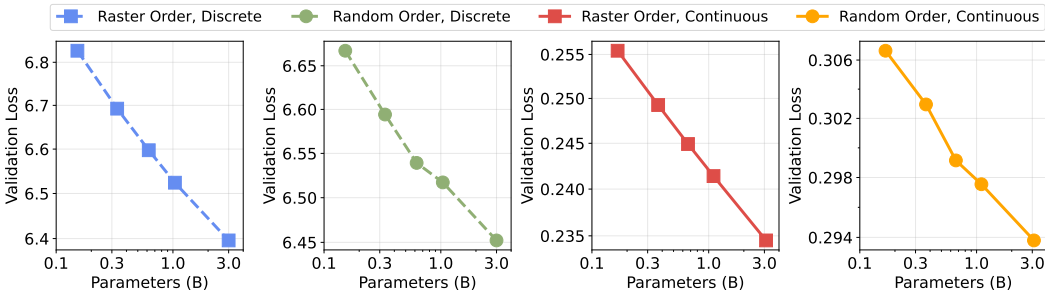


Figure 5: **Validation loss scales as a power-law with model size.** The validation loss is evaluated on 30K images randomly sampled from the MS-COCO 2014 training set. The x and y axes are in log-scale. The change in y is relatively small for each plot, making the log-scale alike linear-scale.

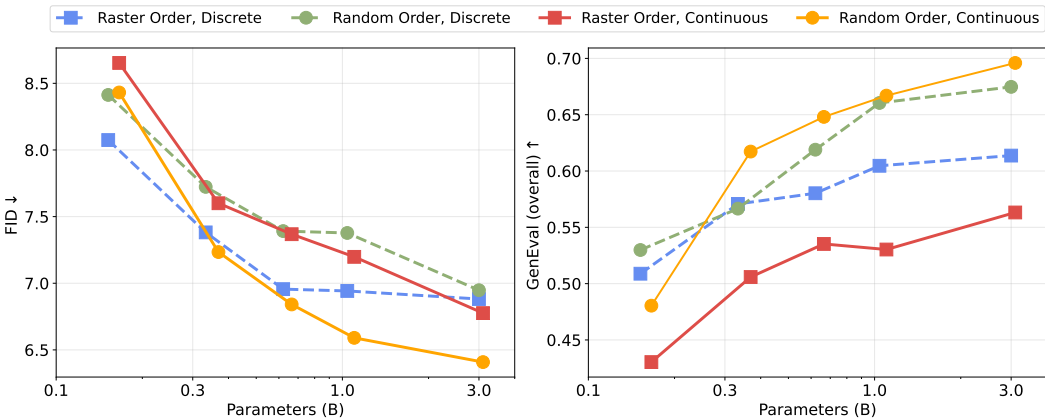
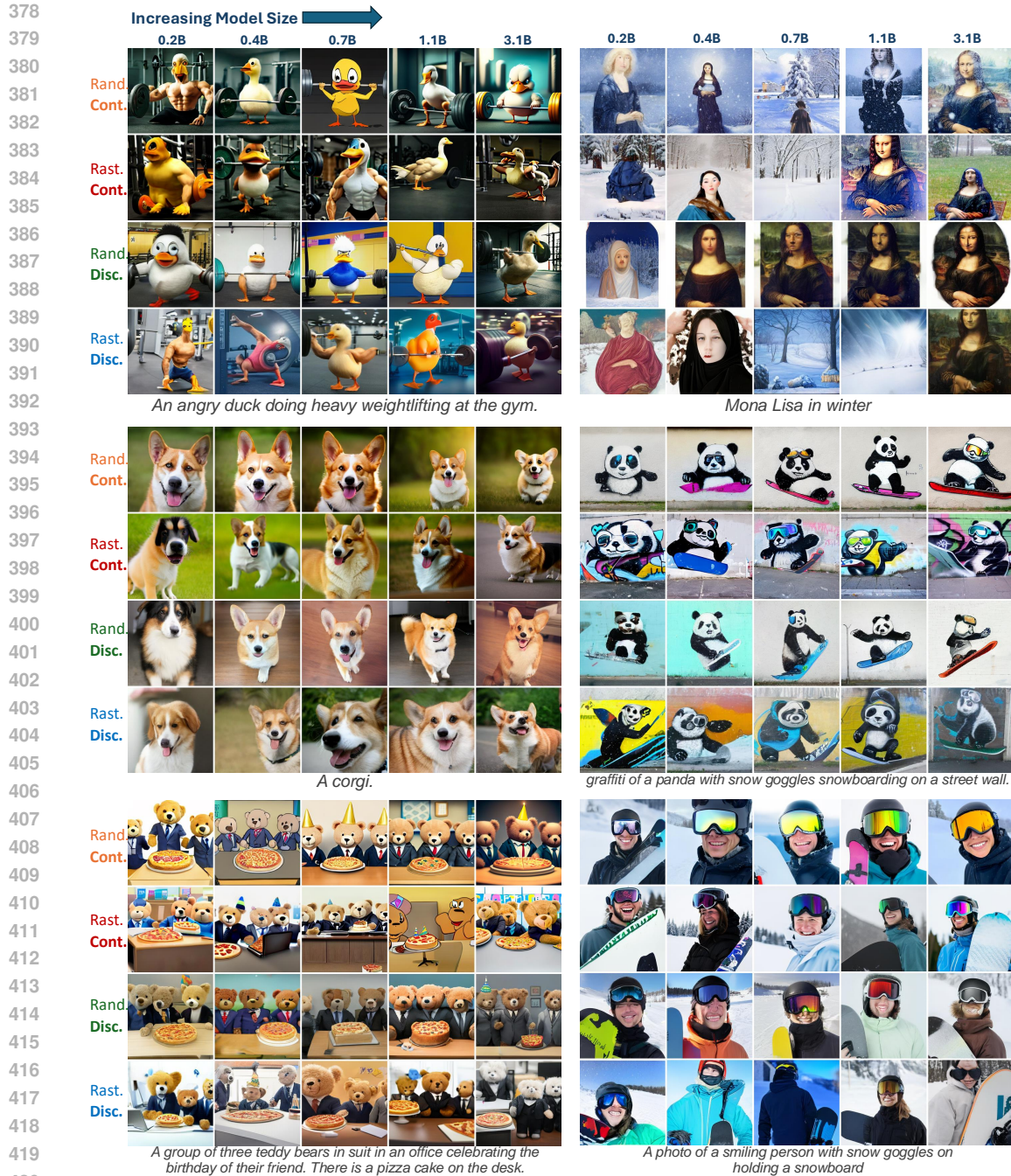


Figure 6: **Random-order models using continuous tokens (orange) achieve the best performance on evaluation metrics.** FID (lower is better) is evaluated on 30K images randomly sampled from the MS-COCO 2014 training set, while the GenEval overall score (higher is better) is assessed using the 553 prompts provided by the official benchmark, with four images generated for each prompt. Among all models, random-order models on continuous tokens consistently show an improvement in evaluation metrics as model size increases and achieve the best FID and GenEval scores.

angry duck doing heavy weightlifting at the gym”, while the same model with 3B parameters can generate the corresponding images successfully. This demonstrates that modeling continuous tokens and increasing model size are crucial for achieving high visual quality in autoregressive image generation models.

**Random-order models with continuous tokens scale best in evaluation scores.** In Figure 6, we analyze the scaling behavior of the four autoregressive variants in terms of FID and GenEval overall scores. We find that the improvements observed in validation loss do not always translate to better evaluation metrics. For example, raster-order models with discrete tokens (blue line) reach a plateau in both FID and GenEval scores around 1B parameters. Among the four variants, random-order models with continuous tokens (*i.e.*, Fluid) show consistent improvements in evaluation metrics up to 3B parameters, achieving the best overall performance. Therefore, we focus on further investigating the scaling behavior of this model in the remaining section.

**Random-order models with continuous tokens scale with training computes.** In Figure 8, we plot validation loss, FID, and GenEval scores as functions of total training steps and compute for different model sizes of Fluid. We observe consistent improvements in both validation loss and evaluation performance with increased training steps and compute. However, the benefits from additional training steps saturate around 1M steps, indicating that training smaller models for more



421 **Figure 7: Visual quality and image-text alignment improves with increasing model size.**  
 422 Random-order (Rand.) models with continuous tokens (Cont.) achieve the highest visual quality  
 423 and best image-text alignment (top row). Best viewed with zoom-in for details.

424

425 steps is less compute-efficient compared to training larger models for fewer steps. This behavior  
 426 aligns with observations in language models, highlighting the potential for scaling up model sizes  
 427 with sufficient training.

428

429

430 **Strong correlation between validation loss and evaluation metrics.** In **Figure 9**, we plot FID  
 431 and GenEval scores against validation loss for different model sizes of Fluid and observe a strong  
 correlation. To quantify this, we fit the data points using linear regression. The Pearson correlation



432  
433  
434  
435  
436  
437  
438  
439  
440  
441  
442  
443  
444  
445  
446  
447  
448  
449  
450  
451  
452  
453  
454  
455  
456  
457  
458  
459  
460  
461  
462  
463  
464  
465  
466  
467  
468  
469  
470  
471  
472  
473  
474  
475  
476  
477  
478  
479  
480  
481  
482  
483  
484  
485

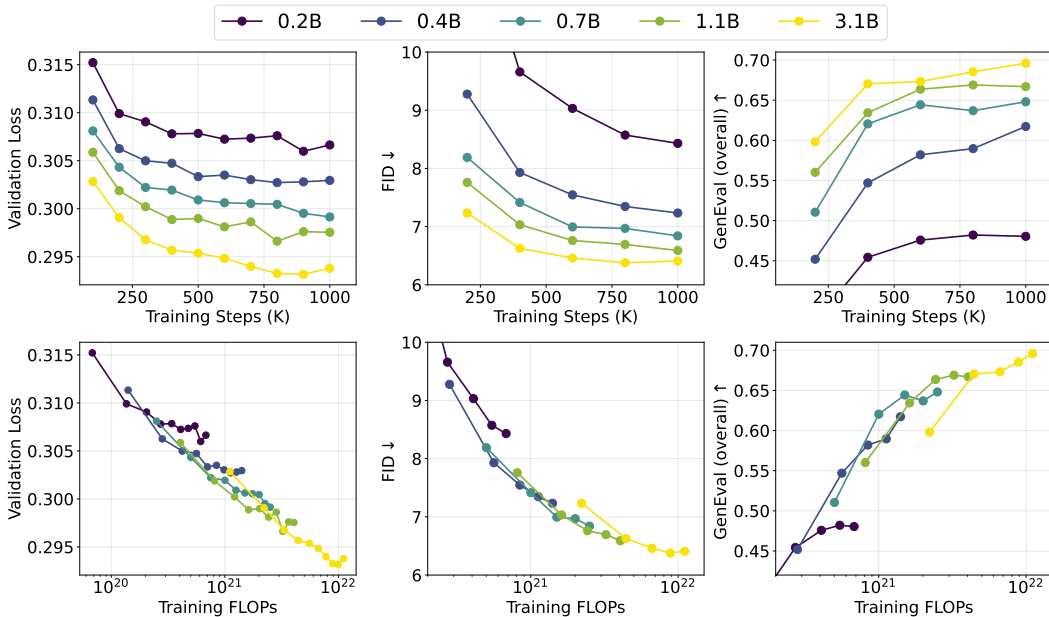


Figure 8: **Validation losses and evaluation performance scale with increasing training steps and computes.** We use random-order models with continuous tokens. Results for other autoregressive variants are included in the appendix. The training compute is computed as model GFLOPs×batch size×training steps×3, where the factor of 3 accounts for the backward pass being approximately twice as compute-intensive as the forward pass.

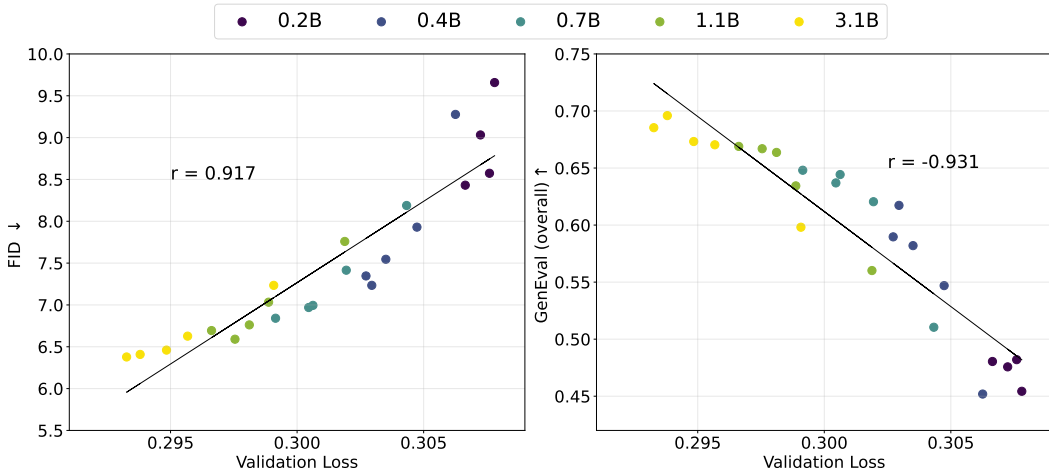


Figure 9: **Validation loss and evaluation metrics are highly correlated.** We use random-order models with continuous tokens. The Pearson correlation coefficients for FID and GenEval scores are 0.917 and -0.931, respectively. We also observe that the linear correlation slightly weakens and becomes less pronounced for the 3.1B model.

coefficients for FID and GenEval scores are 0.917 and -0.931, respectively, indicating a nearly linear relationship between validation loss and these evaluation metrics across model sizes ranging from 150M to 3B<sup>1</sup>. Encouraged by this positive trend, we trained a model with 10.5B parameters and a batch size of 4096 for 1M steps, achieving state-of-the-art text-to-image generation performance, as discussed in the next section.

<sup>1</sup>Since both FID and GenEval scores have lower/upper bounds, this linear relationship cannot hold indefinitely. Future work should explore the limits of this correlation.

Table 1: **System-level comparison.** Fluid achieves leading results on both MS-COCO zero-shot FID-30K and GenEval benchmark (Ghosh et al., 2024). †: CM3Leon result is reported without retrieval.

	#params	MS-COCO FID-30K↓	Single Obj.	Two Obj.	Counting	Colors	Position	Color Attri.	Overall
<i>diffusion model</i>									
LDM	1.4B	12.64	0.92	0.29	0.23	0.70	0.02	0.05	0.37
DALL-E 2	4.2B	10.39	0.94	0.66	0.49	0.77	0.10	0.19	0.52
DALL-E 3	-	-	0.96	0.87	0.47	0.83	0.43	0.45	0.67
Imagen	3B	7.27	-	-	-	-	-	-	-
SD3	8B	-	0.98	0.84	0.66	0.74	0.40	0.43	0.68
Transfusion	7.3B	6.78	-	-	-	-	-	-	0.63
RAPHAEL	3B	6.61	-	-	-	-	-	-	-
<i>autoregressive model</i>									
CM3Leon†	7B	10.82	-	-	-	-	-	-	-
Show-o	1.3B	9.24	0.95	0.52	0.49	0.82	0.11	0.28	0.53
Muse	3B	7.88	-	-	-	-	-	-	-
Parti	20B	7.23	-	-	-	-	-	-	-
<b>Fluid (our work)</b>	369M	7.23	0.96	0.64	0.53	0.78	0.33	0.46	0.62
	665M	6.84	0.96	0.73	0.51	0.77	0.42	0.51	0.65
	1.1B	6.59	0.96	0.77	0.61	0.78	0.34	0.53	0.67
	3.1B	6.41	0.98	0.83	0.60	0.82	0.41	0.53	<b>0.70</b>
	10.5B	<b>6.16</b>	0.96	0.83	0.63	0.80	0.39	0.51	0.69

## 5.2 BENCHMARKING WITH PREVIOUS SYSTEMS

In this section, we compare our Fluid, *i.e.*, continuous random-order autoregressive model, with leading text-to-image generation systems in Table 1 (Rombach et al., 2022b; Ramesh et al., 2022; Betker et al., 2023; Saharia et al., 2022; Esser et al., 2024; Zhou et al., 2024; Xue et al., 2024; Yu et al., 2023; Xie et al., 2024; Chang et al., 2023; Yu et al., 2022). The Fluid smallest model, with 369M parameters, achieves a zero-shot FID of 7.27 on MS-COCO and a GenEval overall score of 0.62, matching the performance of many state-of-the-art models with several billion parameters. The Fluid largest model, with 10.5B parameters, further improves the zero-shot FID on MS-COCO to 6.16 and increases the GenEval overall score to 0.69<sup>2</sup>, with a speed of 1.571 seconds per image per TPU (evaluated on 32 TPU v5 with batch size 2048). Detailed model configurations and generation speed results are included in the appendix. We hope these strong results and promising scaling behavior provide valuable insights and support for the scalability of autoregressive models in visual generative modeling.

## 6 DISCUSSION

In this paper, we present an empirical study on the scaling behavior of autoregressive models for text-to-image generation. We investigate two critical design factors: random order versus raster order, and discrete tokens versus continuous tokens. Our results show that random-order models with continuous tokens achieve the best performance and scaling behavior across various evaluation metrics and in terms of visual quality. Building on these findings, we scale up the random-order model with continuous tokens, namely Fluid, to 10.5B parameters, and achieves state-of-the-art text-to-image generation performance. We hope that our findings and promising results could provide valuable insights into the scaling behavior of autoregressive models for image generation and help bridge the gap between the scaling performance of vision models and language models.

**Reproducibility Statement.** To aid reproducibility, we have provided the implementation details of our framework in Section 4, training hyper-parameters in Section 5, and model configurations in the Appendix. For the diffusion loss used for continuous tokens, we have strictly followed the open-sourced code of Li et al. (2024).

<sup>2</sup>We observe that the GenEval scores plateau for the 10.5B Fluid compared to the 3.1B Fluid; however, it continues to show consistent improvements in visual quality and FID.

## REFERENCES

- 540  
541  
542 Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Ale-  
543 man, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical  
544 report. *arXiv preprint arXiv:2303.08774*, 2023.
- 545 Yutong Bai, Xinyang Geng, Karttikeya Mangalam, Amir Bar, Alan L Yuille, Trevor Darrell, Jitendra  
546 Malik, and Alexei A Efros. Sequential modeling enables scalable learning for large vision models.  
547 In *CVPR*, 2024.
- 548 Jason Baldridge, Jakob Bauer, Mukul Bhutani, Nicole Brichtova, Andrew Bunner, Kelvin Chan,  
549 Yichang Chen, Sander Dieleman, Yuqing Du, Zach Eaton-Rosen, et al. Imagen 3. *arXiv preprint*  
550 *arXiv:2408.07009*, 2024.
- 551 James Betker, Gabriel Goh, Li Jing, Tim Brooks, Jianfeng Wang, Linjie Li, Long Ouyang, Juntang  
552 Zhuang, Joyce Lee, Yufei Guo, et al. Improving image generation with better captions. *OpenAI*,  
553 2023.
- 554 Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal,  
555 Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel  
556 Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler,  
557 Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray,  
558 Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever,  
559 and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020.
- 560 Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Ka-  
561 mar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. Sparks of artificial general  
562 intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*, 2023.
- 563 Huiwen Chang, Han Zhang, Lu Jiang, Ce Liu, and William T Freeman. MaskGIT: Masked genera-  
564 tive image Transformer. In *CVPR*, 2022.
- 565 Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan  
566 Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, Yuanzhen Li, and Dilip Krishnan.  
567 Muse: Text-to-image generation via masked generative Transformers. In *ICML*, 2023.
- 568 Mark Chen, Alec Radford, Rewon Child, Jeffrey Wu, Heewoo Jun, David Luan, and Ilya Sutskever.  
569 Generative pretraining from pixels. In *ICML*, 2020.
- 570 Xi Chen, Nikhil Mishra, Mostafa Rohaninejad, and Pieter Abbeel. PixelSNAIL: An improved au-  
571 toregressive generative model. In *ICML*, 2018.
- 572 Xi Chen, Xiao Wang, Soravit Changpinyo, AJ Piergiovanni, Piotr Padlewski, Daniel Salz, Sebastian  
573 Goodman, Adam Grycner, Basil Mustafa, Lucas Beyer, et al. Pali: A jointly-scaled multilingual  
574 language-image model. *arXiv preprint arXiv:2209.06794*, 2022.
- 575 Mostafa Dehghani, Josip Djolonga, Basil Mustafa, Piotr Padlewski, Jonathan Heek, Justin Gilmer,  
576 Andreas Peter Steiner, Mathilde Caron, Robert Geirhos, Ibrahim Alabdulmohsin, et al. Scaling  
577 vision transformers to 22 billion parameters. In *ICML*, 2023.
- 578 Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep  
579 bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.
- 580 Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha  
581 Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, et al. The llama 3 herd of models.  
582 *arXiv preprint arXiv:2407.21783*, 2024.
- 583 Alaaeldin El-Nouby, Michal Klein, Shuangfei Zhai, Miguel Angel Bautista, Alexander Toshev,  
584 Vaishaal Shankar, Joshua M Susskind, and Armand Joulin. Scalable pre-training of large au-  
585 toregressive image models. *arXiv preprint arXiv:2401.08541*, 2024.
- 586 Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming Transformers for high-resolution image  
587 synthesis. In *CVPR*, 2021.
- 588  
589  
590  
591  
592  
593

- 594 Patrick Esser, Sumith Kulal, Andreas Blattmann, Rahim Entezari, Jonas Müller, Harry Saini, Yam  
595 Levi, Dominik Lorenz, Axel Sauer, Frederic Boesel, et al. Scaling rectified flow transformers for  
596 high-resolution image synthesis. In *ICML*, 2024.
- 597 Dhruva Ghosh, Hannaneh Hajishirzi, and Ludwig Schmidt. Geneval: An object-focused framework  
598 for evaluating text-to-image alignment. *NeurIPS*, 2024.
- 600 Karol Gregor, Ivo Danihelka, Andriy Mnih, Charles Blundell, and Daan Wierstra. Deep autoregres-  
601 sive networks. In *ICML*, 2014.
- 602 Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked  
603 autoencoders are scalable vision learners. In *CVPR*, 2022.
- 605 Tom Henighan, Jared Kaplan, Mor Katz, Mark Chen, Christopher Hesse, Jacob Jackson, Heewoo  
606 Jun, Tom B Brown, Prafulla Dhariwal, Scott Gray, et al. Scaling laws for autoregressive generative  
607 modeling. *arXiv preprint arXiv:2010.14701*, 2020.
- 608 Martin Heusel, Hubert Ramsauer, Thomas Unterthiner, Bernhard Nessler, and Sepp Hochreiter.  
609 GANs trained by a two time-scale update rule converge to a local nash equilibrium. In *NeurIPS*,  
610 2017.
- 612 Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*,  
613 2020.
- 614 Jonathan Ho, William Chan, Chitwan Saharia, Jay Whang, Ruiqi Gao, Alexey Gritsenko, Diederik P  
615 Kingma, Ben Poole, Mohammad Norouzi, David J Fleet, et al. Imagen video: High definition  
616 video generation with diffusion models. *arXiv preprint arXiv:2210.02303*, 2022.
- 618 Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza  
619 Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, et al. Train-  
620 ing compute-optimal large language models. *arXiv preprint arXiv:2203.15556*, 2022.
- 621 Jared Kaplan, Sam McCandlish, Tom Henighan, Tom B Brown, Benjamin Chess, Rewon Child,  
622 Scott Gray, Alec Radford, Jeffrey Wu, and Dario Amodei. Scaling laws for neural language  
623 models. *arXiv preprint arXiv:2001.08361*, 2020.
- 624 Tianhong Li, Yonglong Tian, He Li, Mingyang Deng, and Kaiming He. Autoregressive image  
625 generation without vector quantization. *arXiv preprint arXiv:2406.11838*, 2024.
- 626  
627 Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.
- 628  
629 Midjourney. <https://www.midjourney.com/>, 2023.
- 630  
631 Alexander Quinn Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models.  
632 In *ICML*, 2021.
- 633  
634 OpenAI. Dall-e-2, <https://openai.com/product/dall-e-2>, 2023.
- 635  
636 OpenAI. Sora: Creating video from text. <https://openai.com/sora>, 2024.
- 637  
638 Niki Parmar, Ashish Vaswani, Jakob Uszkoreit, Lukasz Kaiser, Noam Shazeer, Alexander Ku, and  
639 Dustin Tran. Image Transformer. In *ICML*, 2018.
- 640  
641 William Peebles and Saining Xie. Scalable diffusion models with Transformers. In *ICCV*, 2023.
- 642  
643 Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. Improving language under-  
644 standing by generative pre-training. 2018.
- 645  
646 Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi  
647 Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text  
transformer. *JMLR*, 2020.
- Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-  
conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*, 2022.

- 648 Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-  
649 resolution image synthesis with latent diffusion models. In *CVPR, 2022a*.
- 650  
651 Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-  
652 resolution image synthesis with latent diffusion models. In *CVPR, 2022b*.
- 653 Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar  
654 Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. Photorealistic  
655 text-to-image diffusion models with deep language understanding. *NeurIPS, 2022*.
- 656 SentencePiece. Unsupervised text tokenizer for neural network-based text generation. <https://github.com/google/sentencepiece>, 2023.  
657  
658
- 659 Shaden Smith, Mostofa Patwary, Brandon Norick, Patrick LeGresley, Samyam Rajbhandari, Jared  
660 Casper, Zhun Liu, Shrimai Prabhumoye, George Zerveas, Vijay Korthikanti, et al. Using deep-  
661 speed and megatron to train megatron-turing nlg 530b, a large-scale generative language model.  
662 *arXiv preprint arXiv:2201.11990, 2022*.
- 663 Mingxing Tan. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv*  
664 *preprint arXiv:1905.11946, 2019*.
- 665  
666 Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu,  
667 Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. Gemini: a family of highly  
668 capable multimodal models. *arXiv preprint arXiv:2312.11805, 2023*.
- 669 Keyu Tian, Yi Jiang, Zehuan Yuan, Bingyue Peng, and Liwei Wang. Visual autoregressive modeling:  
670 Scalable image generation via next-scale prediction. *arXiv preprint arXiv:2404.02905, 2024*.
- 671  
672 Aaron van den Oord, Nal Kalchbrenner, Lasse Espeholt, Oriol Vinyals, Alex Graves, and Koray  
673 Kavukcuoglu. Conditional image generation with PixelCNN decoders. In *NeurIPS, 2016a*.
- 674 Aaron van den Oord, Nal Kalchbrenner, and Koray Kavukcuoglu. Pixel recurrent neural networks.  
675 In *ICML, 2016b*.
- 676  
677 Aaron van den Oord, Oriol Vinyals, and Koray Kavukcuoglu. Neural discrete representation learn-  
678 ing. *arXiv preprint arXiv:1711.00937, 2017*.
- 679 Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez,  
680 Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS, 2017*.
- 681  
682 Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yo-  
683 gatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. Emergent abilities of large language  
684 models. *arXiv preprint arXiv:2206.07682, 2022*.
- 685 Jinheng Xie, Weijia Mao, Zechen Bai, David Junhao Zhang, Weihao Wang, Kevin Qinghong Lin,  
686 Yuchao Gu, Zhijie Chen, Zhenheng Yang, and Mike Zheng Shou. Show-o: One single transformer  
687 to unify multimodal understanding and generation. *arXiv preprint arXiv:2408.12528, 2024*.
- 688 Zeyue Xue, Guanglu Song, Qiushan Guo, Boxiao Liu, Zhuofan Zong, Yu Liu, and Ping Luo.  
689 Raphael: Text-to-image generation via large mixture of diffusion paths. *NeurIPS, 2024*.
- 690  
691 Jiahui Yu, Xin Li, Jing Yu Koh, Han Zhang, Ruoming Pang, James Qin, Alexander Ku, Yuanzhong  
692 Xu, Jason Baldridge, and Yonghui Wu. Vector-quantized image modeling with improved vqgan.  
693 *arXiv preprint arXiv:2110.04627, 2021*.
- 694 Jiahui Yu, Yuanzhong Xu, Jing Yu Koh, Thang Luong, Gunjan Baid, Zirui Wang, Vijay Vasudevan,  
695 Alexander Ku, et al. Scaling autoregressive models for content-rich text-to-image generation.  
696 *arXiv preprint arXiv:2206.10789, 2022*.
- 697  
698 Lili Yu, Bowen Shi, Ramakanth Pasunuru, Benjamin Muller, Olga Golovneva, Tianlu Wang, Arun  
699 Babu, Binh Tang, Brian Karrer, Shelly Sheynin, et al. Scaling autoregressive multi-modal models:  
700 Pretraining and instruction tuning. *arXiv preprint arXiv:2309.02591, 2023*.
- 701 Xiaohua Zhai, Alexander Kolesnikov, Neil Houlsby, and Lucas Beyer. Scaling vision transformers.  
In *CVPR, 2022*.

702 Chunting Zhou, Lili Yu, Arun Babu, Kushal Tirumala, Michihiro Yasunaga, Leonid Shamis, Jacob  
703 Kahn, Xuezhe Ma, Luke Zettlemoyer, and Omer Levy. Transfusion: Predict the next token and  
704 diffuse images with one multi-modal model. *arXiv preprint arXiv:2408.11039*, 2024.  
705  
706  
707  
708  
709  
710  
711  
712  
713  
714  
715  
716  
717  
718  
719  
720  
721  
722  
723  
724  
725  
726  
727  
728  
729  
730  
731  
732  
733  
734  
735  
736  
737  
738  
739  
740  
741  
742  
743  
744  
745  
746  
747  
748  
749  
750  
751  
752  
753  
754  
755

756  
757  
758  
759  
760  
761  
762  
763  
764  
765  
766  
767  
768

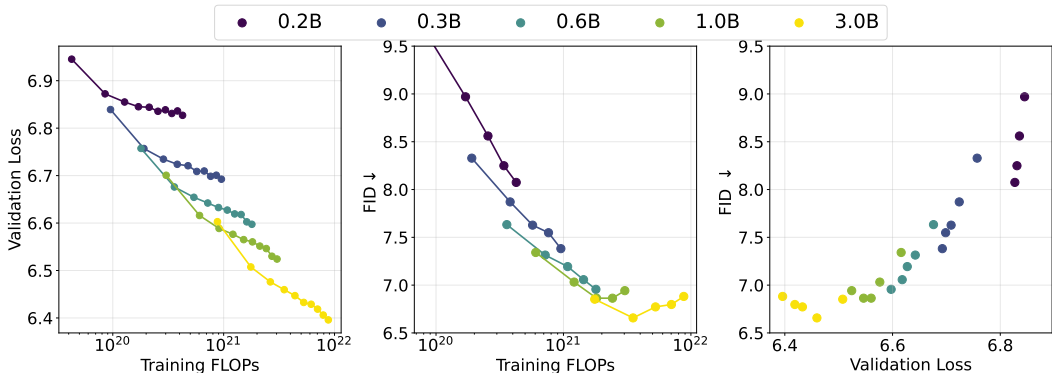


Figure 10: Validation loss and FID w.r.t. training FLOPs for raster-order models with discrete tokens.

771  
772  
773  
774  
775  
776  
777  
778  
779  
780  
781  
782

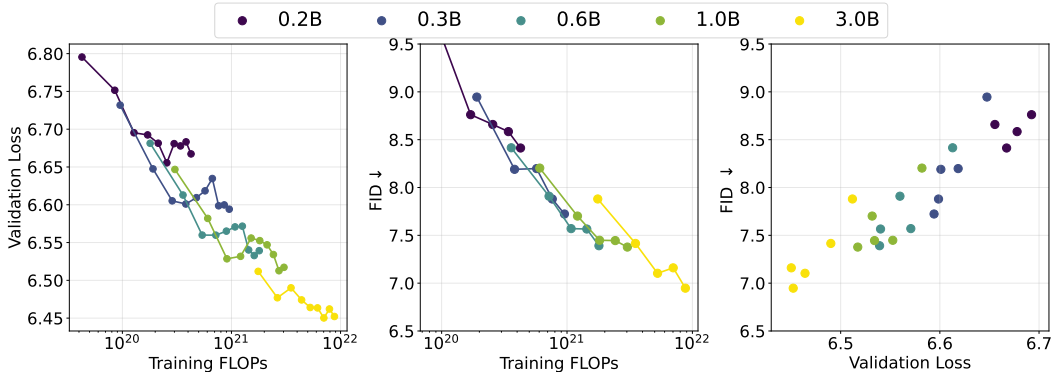


Figure 11: Validation loss and FID w.r.t. training FLOPs for random-order models with discrete tokens.

785  
786  
787  
788  
789  
790  
791  
792  
793  
794  
795  
796

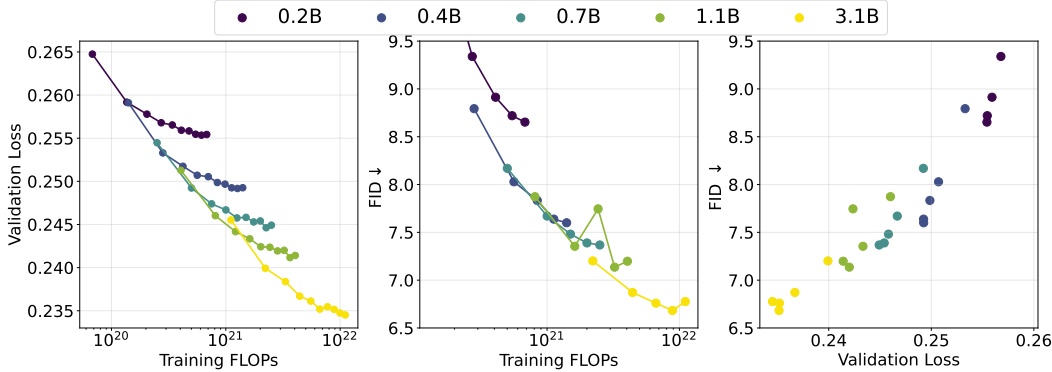


Figure 12: Validation loss and FID w.r.t. training FLOPs for raster-order models with continuous tokens.

800  
801  
802  
803  
804  
805  
806  
807  
808  
809

**Scaling behavior w.r.t training FLOPs.** In Figure 10, Figure 11 and Figure 12, we present the relationship between validation loss and FID with respect to training FLOPs for the other three autoregressive variants. As shown, all three variants demonstrate consistent scaling behavior in validation loss, while improvements in FID begin to plateau for the 3B raster-order model using discrete tokens. This suggests that directly applying a GPT-like language model to images in a brute-force manner may not scale effectively. Further adaptations, such as the use of continuous tokens and random-order generation, are necessary to enhance the scaling performance of language models on visual data.

**Fine-grained GenEval scores.** In Figure 13, we present the performance of the four autoregressive variants across all metrics in the GenEval benchmark. As shown, all models perform well in

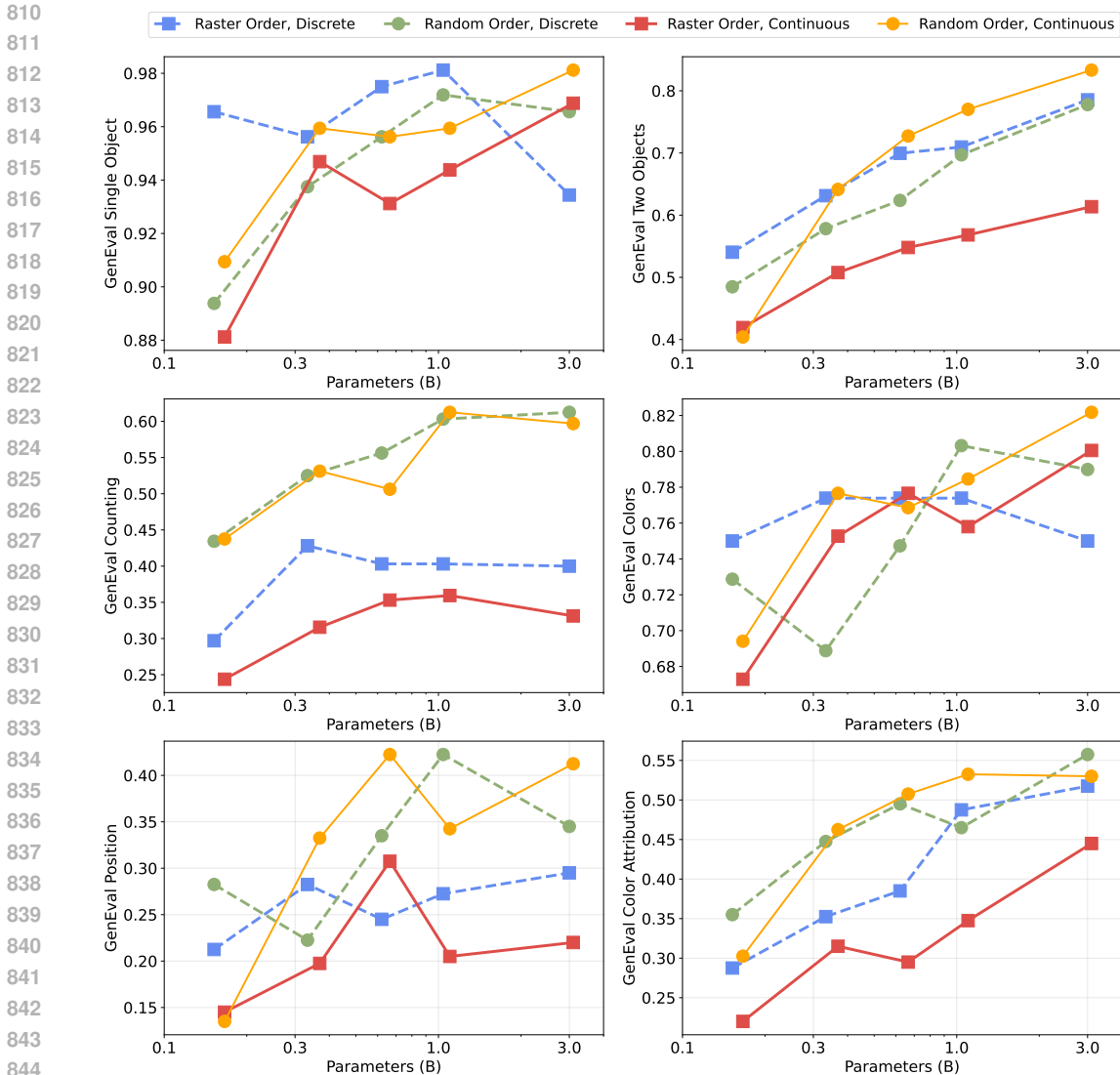


Figure 13: Validation loss and FID w.r.t. training FLOPs for raster-order models with discrete tokens.

single-object scenarios. The performance on other metrics also consistently improves as model size increases. Additionally, we observe that random-order models significantly outperform raster-order models in metrics related to counting and position, both of which require a better global generation structure—an area where random-order models have an advantage.

**Model Configurations.** In Table 2, we provide the detailed configurations of our models across different sizes. The MLP ratio is fixed at 4 for all models. The text aligner consistently consists of 6 transformer blocks, with the same channel size as the image transformer. The DiffLoss MLP also contains 6 MLP layers, with channels matching those of the image transformer. The generation speed is evaluated on 32 TPU v5 with a batch size of 2048, and we report the time needed to generate one image per TPU.

Table 2: Model configurations of our random-order models on continuous tokens.

#Params	#Blocks	#Channels	#Heads	Speed (sec/img)
166M	12	768	12	0.047
369M	16	1024	16	0.078
665M	20	1280	16	0.110
1.1B	24	1536	16	0.180
3.1B	32	2304	24	0.483
10.5B	34	4096	64	1.571



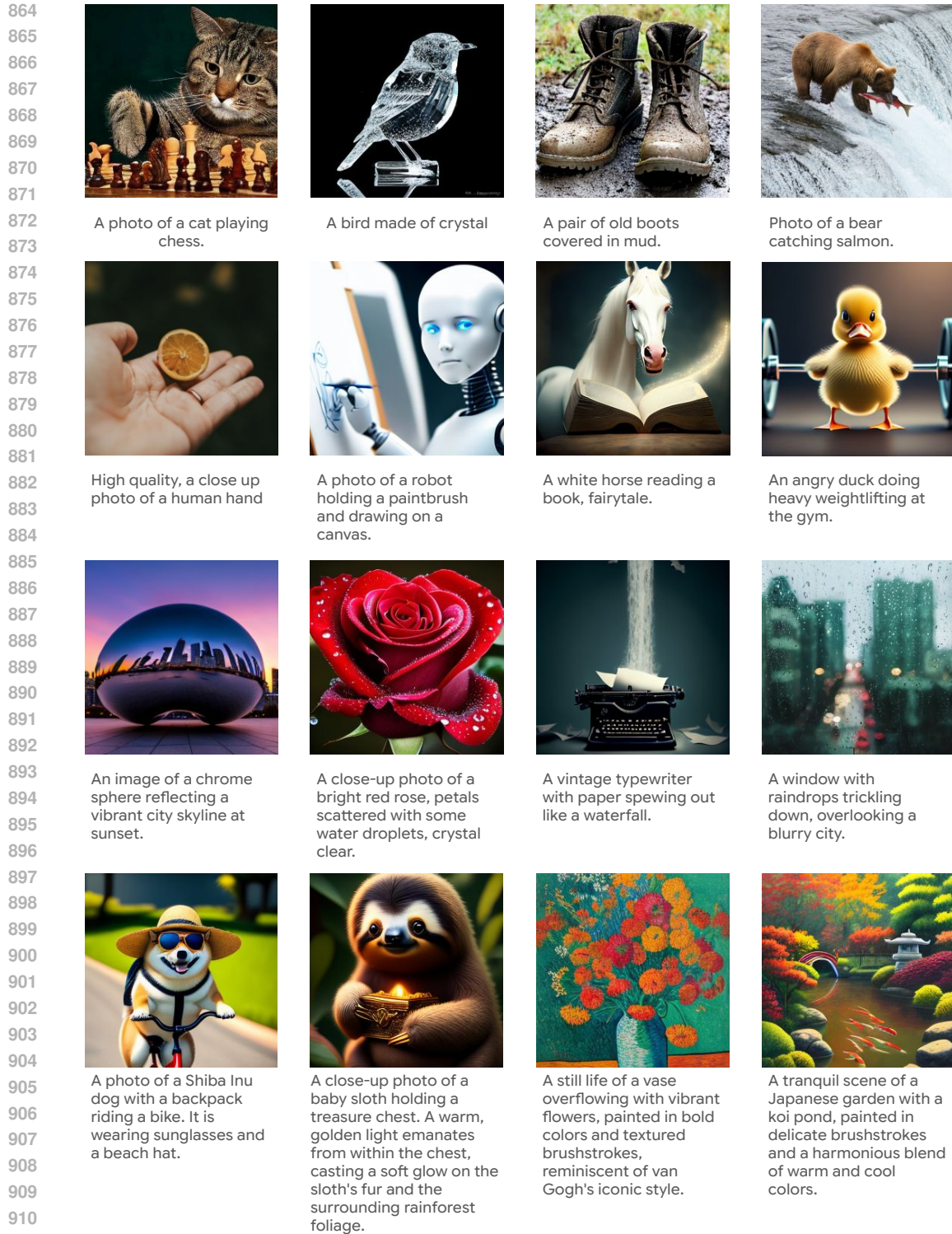


Figure 14: Additional images generated from our 10.5B autoregressive model with continuous tokens.

**Visualization Details of Figure 1.** To generate Figure 1, we first used our 10.5B random-order model with continuous tokens to generate  $256 \times 256$  images conditioned on the text prompts. We then applied an in-house super-resolution model to upscale the images to  $1024 \times 1024$  for improved

918 visual quality (only for Figure 1). The prompts used, from left to right and top to bottom, are:  
919 “Close up photo of a knight”, “A baseball bat”, “An origami bird made of paper is perched on a  
920 branch of an evergreen tree”, “A wise old mushroom wearing spectacles and reading a book under  
921 a tree”, “photo of an eagle with a golden crown resting upon its head”, “A beautiful castle beside a  
922 waterfall in the woods by Josef Thoma, matte painting, trending on artstation HQ”, “A photorealistic  
923 image of a beautiful mountain range reflected perfectly on the still surface of a crystal-clear lake,  
924 surrounded by lush green meadows and vibrant wildflowers”, “Oil painting of a vibrant landscape of  
925 rolling hills covered in wildflowers, with a quaint farmhouse nestled in the distance under a bright  
926 blue sky”, “Mona Lisa on a cake”, “A grumpy-looking lemon wearing a raincoat and holding an  
927 umbrella, standing in the rain”, “A hyperrealistic close-up of an eye, with the iris reflecting a vast  
928 and detailed landscape, complete with mountains, rivers, and forests”, “A section of the Great Wall  
929 in the mountains, detailed charcoal sketch”, “A present with a blue ribbon under a Christmas tree”.

930 **Additional Qualitative Results.** In [Figure 14](#) we show additional images generated from our  
931 10.5B model.  
932  
933  
934  
935  
936  
937  
938  
939  
940  
941  
942  
943  
944  
945  
946  
947  
948  
949  
950  
951  
952  
953  
954  
955  
956  
957  
958  
959  
960  
961  
962  
963  
964  
965  
966  
967  
968  
969  
970  
971