Scaling Autoregressive Multi-Modal Models: Pretraining and Instruction Tuning

Lili Yu[*] **Bowen Shi**[*] **Ramakanth Pasunuru**[*] **Benjamin Muller** **Olga Golovneva**

Tianlu Wang **Arun Babu** **Binh Tang** **Brian Karrer** **Shelly Sheynin**

Candace Ross **Adam Polyak** **Russell Howes** **Vasu Sharma** **Puxin Xu**

Hovhannes Tamoyan[1] **Oron Ashual** **Uriel Singer** **Shang-Wen Li** **Susan Zhang**

Gargi Ghosh **Yaniv Taigman** **Maryam Fazel-Zarandi** **Asli Celikyilmaz**

Luke Zettlemoyer **Armen Aghajanyan**[*]
FAIR, YerevaNN[1]
armenag@meta.com

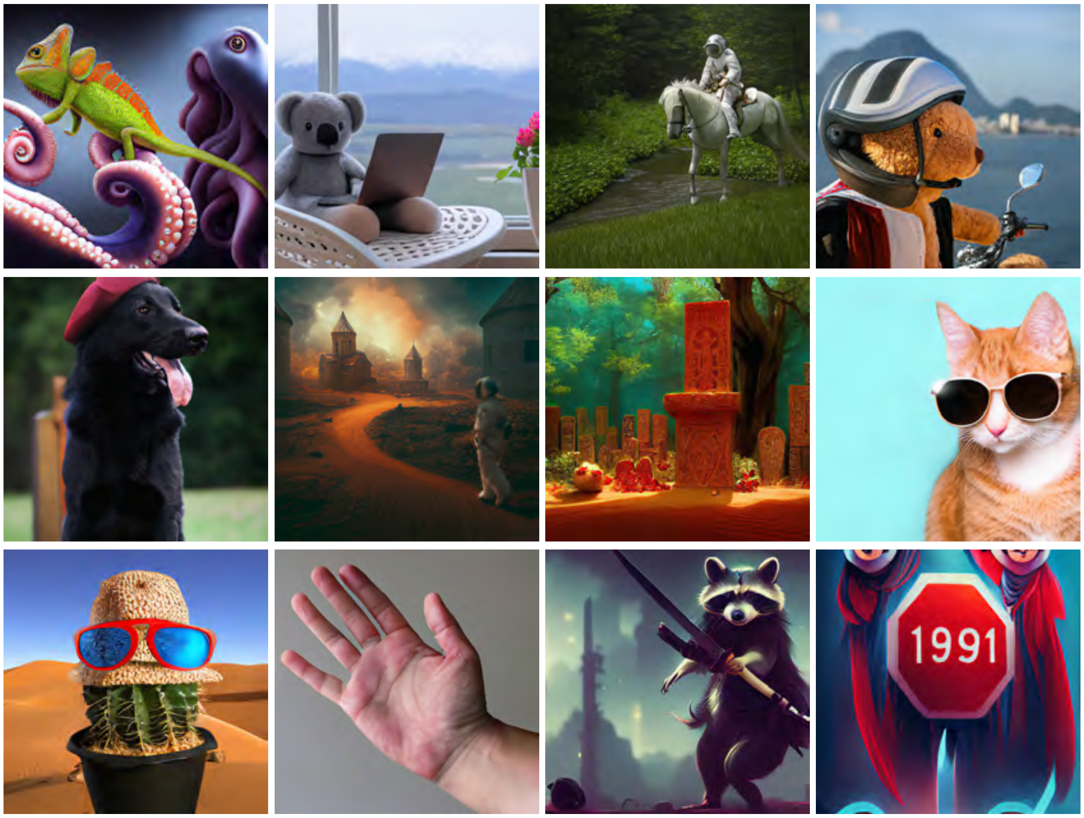


Figure 1: Showcase of CM3Leon zero-shot generations (no-retrieval augmentation). Refer to § A for a complete list of prompts. CM3Leon can generate complex compositional objects, tail entities (Khachkar–Armenian crosses carved from stone), and historically hard entities such as hands and text.

[*] First Author

Abstract

We present CM3Leon (pronounced "Chameleon"), a retrieval-augmented, token-based, decoder-only multi-modal language model capable of generating and infilling both text and images. CM3Leon uses the CM3 multi-modal architecture but additionally shows the extreme benefits of scaling up and tuning on more diverse instruction-style data. It is the first multi-modal model trained with a recipe adapted from text-only language models, including a large-scale retrieval-augmented pretraining stage and a second multi-task supervised fine-tuning (SFT) stage. It is also a general-purpose model that can do both text-to-image and image-to-text generation, allowing us to introduce self-contained contrastive decoding methods that produce high-quality outputs. Extensive experiments demonstrate that this recipe is highly effective for multi-modal models. CM3Leon achieves state-of-the-art performance in text-to-image generation with 5x less training compute than comparable methods (**zero-shot MS-COCO FID of 4.88**). After SFT, CM3Leon can also demonstrate unprecedented levels of controllability in tasks ranging from language-guided image editing to image-controlled generation and segmentation.

1 Introduction

Diffusion models have recently dominated image generation work due to their strong performance and relatively modest computational cost (Saharia et al., 2022; Chen et al., 2022; Rombach et al., 2022). In contrast, token-based autoregressive models (Ramesh et al., 2021; Yu et al., 2022) are known to also produce strong results, with even better global image coherence in particular, but are much more expensive to train and use for inference. In this paper, we show that it is possible to extend training and inference ideas originally developed for text-only models to flip this narrative; autoregressive models can be efficient and performant while also generalizing beyond the strict text-to-image format to be tuneable for a wide range of image and text generation tasks.

More specifically, we introduce CM3Leon (pronounced "Chameleon"), a retrieval-augmented, token-based, decoder-only multi-modal language model capable of generating and infilling both text and images. CM3Leon uses the CM3 multi-modal architecture (Aghajanyan et al., 2022), but additionally shows the extreme benefits of scaling up and training on more diverse data. It is the first multi-modal model trained with a recipe adapted from text-only language models, including a large-scale retrieval-augmented pretraining stage and a second multi-task supervised fine-tuning (SFT) stage. The pretraining is efficient because it follows the retrieval-augmented CM3 approach (Yasunaga et al., 2022) but uses a new large-scale Shutterstock dataset that includes only licensed image and text data. The SFT stage follows multi-task instruction tuning for text-only models Iyer et al. (2022), which allow arbitrary mixtures of image and text tokens in both the inputs and outputs. The generality of CM3Leon also supports the introduction of an improved, self-contained contrastive decoding method Li et al. (2022), which can provide self-guidance to improve both text and image generation.

CM3Leon achieves state-of-the-art performance in text-to-image generation with 5x less training compute than comparable methods (**zero-shot MS-COCO FID of 4.88**). It can also do non-trivial image-to-text generation, even though it was trained on only 3B Shutterstock text tokens. After SFT, CM3Leon demonstrates unprecedented levels of controllability in tasks ranging from language-guided image editing to image-controlled generation and segmentation. We also show that retrieval augmentation is key for efficient training, and our new contrastive decoding method enables much higher quality generation overall. These results strongly suggest that autoregressive models are worth significantly more study for any text and image task.

2 Pretraining

We explore the potential of token-based decoder-only models in the text-to-image domain by building upon the foundation laid by RA-CM3 Yasunaga et al. (2022). We simplify the original settings in RA-CM3 by streamlining the objective, modifying the dataset, and incorporating insights from multi-modal scaling laws presented by Aghajanyan et al. (2023).

2.1 Data

The ethical implications of image data sourcing in the domain of text-to-image generation have been a topic of considerable debate. In this study, we use only licensed images from Shutterstock. As a result, we can avoid concerns related to images ownership and attribution, without sacrificing performance.

Image Tokenization We use the image tokenizer from Gafni et al. (2022a), which encodes a 256×256 image into 1024 tokens from a vocabulary of 8192. For text, we train a custom tokenizer over the Zhang et al. (2022) data with a vocabulary size of 56320. Additionally, we introduce a novel special token, denoted as `<break>`, which serves to indicate a transition between modalities. A visualization of one caption-image pair after tokenization and formatting with our special tokens is available in § B.1(Figure 8).

Retrieval Augmentation Our retrieval approach aims to retrieve relevant and diverse multi-modal documents from a memory bank, given an input sequence (Yasunaga et al., 2022). It includes both a dense retriever and a retrieval strategy.

The dense retriever takes a query q (e.g., the input sequence x) and a candidate document m from the memory bank $\mathcal{M}$ and returns a relevance score $r(q, m)$. We adopt the dense retrieval method from Karpukhin et al. (2020), which uses a bi-encoder architecture. The encoder is CLIP-based. We split the multi-modal document into a text part and an image part, encode them separately using off-the-shelf frozen CLIP text and image encoders, and then average the two as a vector representation of the document (Radford et al., 2021). We use the ViT-B-32 model and normalize the image/text embeddings. The final retrieval is done with Maximum Inner Product Search (MIPS) over the memory bank using the dense retriever to obtain a list of candidate documents sorted by relevance score (Tiwari et al., 2022).

To sample informative retrieved documents for the generator during training, we consider three key factors: relevance, modality, and diversity. First, the retrieved documents should be relevant to the input sequence, captured by the dense retriever score based on CLIP. Second, retrieving a multi-modal document consisting of images and text leads to better generator performance than retrieving either image or text. Third, diversity is essential to avoid redundancy in the retrieved documents. Simply taking the top K documents based on relevance score can result in duplicates or highly similar documents, hurting downstream pretraining. We skip a candidate document if it is too similar to the query or if the documents have already been retrieved. In practice, we only use retrieved documents with relevance score ≤ 0.9. Additionally, we use query dropout, which drops some tokens of the query used in retrieval (20% of tokens) to encourage diversity and serve as regularization for training.

Throughout our work, we retrieve two documents each, based on image and text, respectively. In training, we randomly select three retrieved samples for every caption-image pair in our dataset, effectively 4x the number of tokens available in the pretraining. A visualization of a single training example can be found in § B.1(Figure 9).

2.2 Objective Function

The CM3 objective accepts multi-modal inputs (e.g., $x_{\text{input}} =$ "Image of a chameleon: `[image]`") and transforms them into an infilling instance by masking specific spans and relocating them to the end (e.g., $x_{\text{input}} =$ "Image of `<mask>`: `[image] <infill>` a chameleon"). It uses a standard next token prediction loss, $-\log p(x_{\text{input}})$. This results in a versatile model capable of infilling and autoregressive generation tasks for both images and text. In the case of caption-to-image generation, CM3 creates a continuation from the prompt "Image of a chameleon:". For image-to-caption generation, CM3 utilizes the prompt "Image of `<mask>`: `[image] <infill>`".

Yasunaga et al. (2022) built upon the original CM3 by including retrieved multi-modal documents in the context for each training example and up weighting the query image-caption pair loss, as illustrated in the last image-caption pair in Figure 9. This approach encourages the model to concentrate more on using retrieved samples during the generation process. However, this method adversely affects the zero-shot scenario, where the goal is to generate an image without retrieval, such as predicting a continuation from `<eos> text <break>`. We remove this weighting in our setting and make a minor modification to the CM3 objective by preventing masking across `<break>` tokens. This adjustment is justified by the fact that allowing masking across <break> tokens may lead to the model generating image content from an arbitrary midpoint, which is not a desirable outcome.

2.3 Model

The CM3Leon models follow a decoder-only transformer architecture, similar to Zhang et al. (2022) and Brown et al. (2020). Compared to Zhang et al. (2022), we remove bias terms, dropout, and learnable parameters for layer norms and use a sequence length of 4096 instead of 2048. For weight initialization, we use a truncated normal distribution with a mean of 0 and a standard deviation of 0.006, truncated to 3 standard deviations. Output layers are initialized as 0, and the learned absolute positional embedding is initialized near zero with a standard deviation of 0.0002. The models were trained with Metaseq[2], with experiment tracking done with Aim Arakelyan et al. (2020).

2.4 Training

Our models are trained across three distinct sizes, with the corresponding parameters and training setup detailed in Table 3. The major hyperparameters, such as the learning rate and batch size, are adopted from prior work in multi-modal scaling laws, creating a stable and smooth training progression as illustrated in Figure 3 (Aghajanyan et al., 2023). The 350 Million (350M), 760 Million (760M), and 7 Billion (7B) models are trained to 1.4 Trillion (T), 1.9T, and 2.4T tokens, respectively. The losses for all three models decrease steadily throughout training, strongly suggesting they have not saturated.

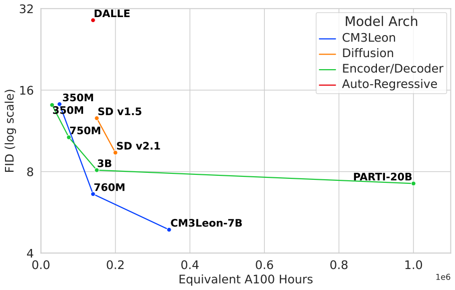


Figure 2: We plot FID score in log scale of various models against the equivalent A100 GPU hours during training. CM3Leon scales better than DALLE (Ramesh et al., 2021), stable diffusion (SD) (Rombach et al., 2022) and PARTI (Yu et al., 2022) models.

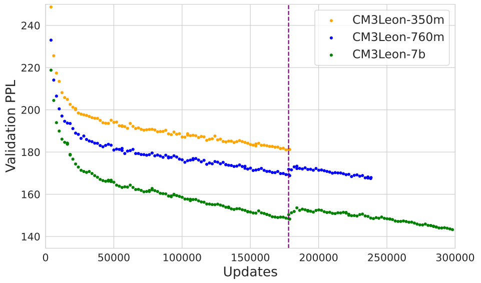


Figure 3: We plot validation perplexity (PPL) against with number of training updates for CM3Leon models in 350m, 760m and 7b size. We resume the training of 760m and 7b models after a full epoch (the purple dashed line), and the small rise in the PPL is due to the sudden increase of the learning rate.

3 Text-To-Image Results

3.1 Importance of Decoding Strategies

There has been significant work on developing decoding algorithms for autoregressive text-to-image models, such as DALL-E Ramesh et al. (2021), which can have a large effect on the quality of the final outputs. DALL-E employs temperature sampling and a re-ranking stage via CLIP over 512 prompt candidates. Models like PARTI and Make-A-Scene user token-based classifier-free guidance, significantly reducing the number of candidates required for re-ranking to just 16 samples (Yu et al., 2022; Gafni et al., 2022a). Our experiments show that different approaches offer complementary benefits, as decribed in this section. We compare the following options.

Temperatured Sampling is a probabilistic technique used in autoregressive models, such as Ramesh et al. (2021). The method involves modifying the softmax temperature during the sampling stage to control the randomness of predictions. We pair this with Classifier Free Guidance in all of our experiments.

[2] https://github.com/facebookresearch/metaseq

TopP Sampling also known as nucleus sampling, involves sampling from the smallest set of top-ranked tokens with a cumulative probability exceeding a predefined threshold (Holtzman et al., 2020). We pair this with Classifier Free Guidance in all of our experiments.

Classifier Free Guidance (CFG) Classifier-free guidance refers to directing an unconditional sample towards a conditional sample (Gafni et al., 2022a). We replace the text with the mask token from the CM3 objective to facilitate unconditional sampling. This is one of the core benefits of training with the CM3 objective, allowing us to do classifier-free guidance without the need for finetuning. During the inference stage, two concurrent token streams are generated: a conditional token stream, which is contingent on the input text, and an unconditional token stream, which is conditioned on a mask token. Borrowing the notation from Gafni et al. (2022a):

$$\text{logits}_{\text{cond}} = T(t_y|t_x), \text{logits}_{\text{uncond}} = T(t_y|\texttt{<mask>}), \tag{1}$$

$$\text{logits}_{\text{cf}} = \text{logits}_{\text{uncond}} + \alpha_c \cdot (\text{logits}_{\text{cond}} - \text{logits}_{\text{uncond}}) \tag{2}$$

where T denotes the transformer, t_y is the output tokens and t_x is the conditional input text, `<mask>` represents the absence of input text (and replacement with a mask token), and α_c is a scaling factor. The classifier-free guidance effectively blends the unconditional and conditional logits, influencing the model's output towards a more desired conditional output.

Contrastive Decoding TopK (CD-K) A key insight is that the logit subtraction in Equation 2 resembles the log probability subtraction in contrastive decoding methods in text (Li et al., 2022). This leads us to propose a variant of the contrastive decoding (CD) algorithm, originally proposed by Li et al. (2022), as an alternative to CFG.

Recall that CD defines a score per token:

$$CD(t_{y_i}; t_{y_{<i}}) = \begin{cases} \log \frac{p_{\text{EXP}}(t_{y_i}|t_{y_{<i}})}{p_{\text{AMA}}(t_{y_i}|t_{y_{<i}})}, & \text{if } t_{y_i} \in \mathcal{V}(t_{y_{<i}}), \\ -\inf, & \text{otherwise.} \end{cases}$$

Here, $\mathcal{V}(t_{y_{<i}})$ represents the set of potential subsequent tokens whose probabilities are at least α times the maximum probability value:

$$\mathcal{V}(t_{y_{<i}}) = \{t_{y_i} \in \mathcal{V} : p_{\text{EXP}}(t_{y_i} \mid t_{y_{<i}}) \geq \alpha \max_w p_{\text{EXP}}(w|t_{y_{<i}})\}$$

Traditionally p_{EXP} and p_{AMA} in the CD decoding algorithm represent a strong and weak model where the strong model was trained with more compute (or larger model size) compared to the weak model. Instead we select p_{EXP} having text conditioning and p_{AMA} has no text conditioning. Additionally we saw that the $\mathcal{V}(t_{y_{<i}})$ constraint was too strict, and would consistently become greedy decoding. Therefore we propose a slight modification of CD we call CD-K that alters $\mathcal{V}(t_{y_{<i}})$ to:

$$\mathcal{V}(t_{y_{<i}}) = \{t_{y_i} \in \mathcal{V} : p_{\text{EXP}}(t_{y_i} \mid t_{y_{<i}}) \geq \alpha * \operatorname*{kmax}_{k,w} \left(p_{\text{EXP}}(w|t_{y_{<i}})\right)\} \tag{3}$$

where instead of taking the largest probability we take the k-th largest probability.

Ablation In Figure 4 we show that CD-K is competitive with standard CFG based sampling while providing a complementary set of generations to CFG allowing us to continue minimizing FID as we increase number of generations (while both CD-K and CFG independently stagnate).

3.2 Quantitative Evaluations

Table 1 and Figure 2 provide a comparative overview of CM3Leon and state-of-the-art text-to-image models, evaluated based on the zero-shot MS-COCO (30K) task using the Fréchet Inception Distance (FID) metric (Seitzer, 2020). CM3Leon-7B model set's a new state-of-the-art FID score of 4.88, while only using a fraction of the training data and compute of other models such as PARTI.

This observation underlines the effectiveness of retrieval-augmented decoder-only models like CM3Leon. In particular, the CM3Leon-7B model, when operated with one or two retrieved examples during inference, records superior FID scores. This result demonstrates the crucial role retrieval plays in expanding the world knowledge provided to the model and its capacity to generate high-quality images. CM3Leon surpasses all other retrieval-augmented models, including KNN-diffusion and RE-IMAGEN.

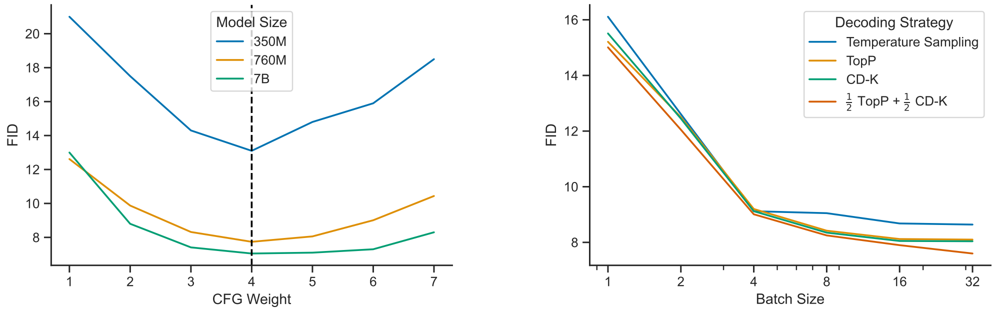


Figure 4: **(Left)** Comparison of Classifier-Free Guidance (CFG) weight and FID on 8k held-out MS-COCO data across our series of models. The optimal CFG remains consistent across all model sizes. **(Right)** Comparison of the number of generated samples per prompt before CLIP-based re-ranking and their respective FID. The data shows that TopP and CD-K are similar across sample counts but exhibit complementary behavior when combined.

	Retrieval in Training	Responsible	# of Retrieved Documents	Dataset Size	Model Size	Zero-shot FID-30K
RA-CM3	✓	✗	2	150M	2.7B	15.70
StableDiffusion	✗	✗	-	400M	800M	12.60
KNN-Diffusion	✓	✗	10	70M	400M	12.50
MUSE	✗	✗	-	500M	3B	7.88
PARTI	✗	✗	-	5B	20B	7.23
RE-IMAGEN	✓	✗	2	450M	3.6B	5.25
CM3Leon-7B	✓	✓	0	340M	7B	10.82
CM3Leon-7B	✓	✓	1	340M	7B	5.78
CM3Leon-350M	✓	✓	2	340M	350M	14.20
CM3Leon-760M	✓	✓	2	340M	760M	6.61
CM3Leon-7B	✓	✓	2	340M	7B	**4.88**

Table 1: Summary of various text-to-image models on the zero-shot MS-COCO task as measured by FID. For all of our models, we generate 8 samples for each input query, and use a CLIP model to select the best generation.

4 Supervised Fine-Tuning

Supervised fine-tuning (SFT) is critical in training large language models (LLMs) like ChatGPT. Despite this, its application in multi-modal settings remains largely unexplored. SFT trains a model to better understand of future instructions or prompts, enhancing its performance in novel and even zero-shot tasks. We have found that instruction tuning notably amplifies multi-modal model performance across various tasks such as image caption generation, visual question answering, text-based editing, and conditional image generation.

We fine-tune CM3Leon on a wide array of mixed image and text tasks. We organized each task as a series of interleaved text and image examples, as shown in Figure 5. The fine-tuning process follows the pretraining stage, employing the same CM3 objective by combining the task instruction with the output. Further details about the hyperparameters and scale of the SFT can be found in Section E.1.

4.1 Instructable Image Generation

Text-Guided Image Editing allows the modification of an initial image based on text instructions, with changes such as seasonal and weather adjustments, background changes, and material alterations. We used InstructPix2Pix methodology and proprietary face-filtering techniques on their data, yielding around 600,000 examples (Brooks et al., 2023).

Image-to-Image Grounded Generation involves producing grounding images with various features and text prompts. Features like edge maps, segmentation maps, key points, and human poses

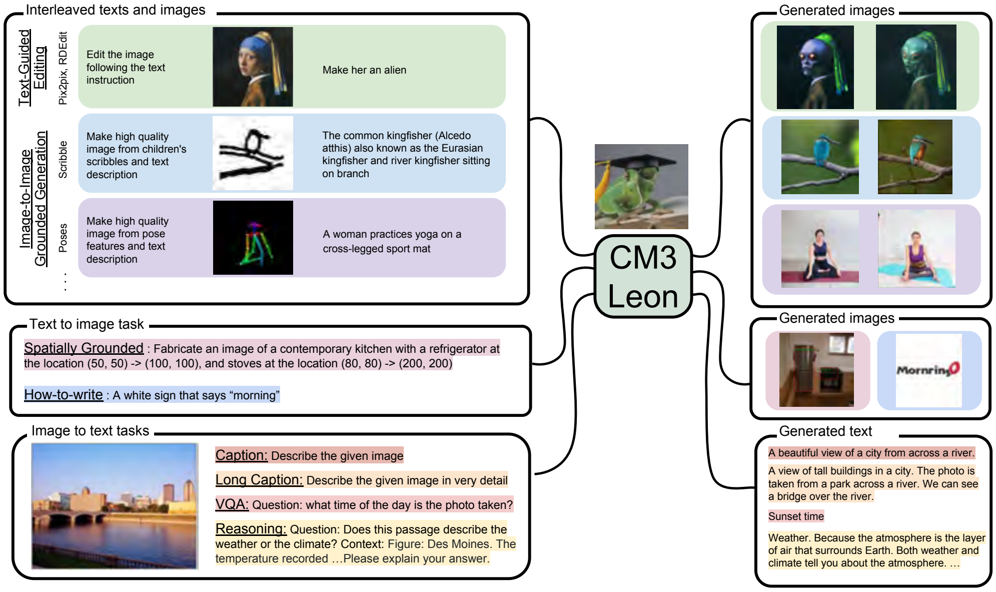

Interleaved texts and images
Text-Guided Editing
Pix2pix, RDEdit
Edit the image following the text instruction
Make her an alien
Image-to-Image Grounded Generation
Scribble
Make high quality image from children's scribbles and text description
The common kingfisher (Alcedo atthis) also known as the Eurasian kingfisher and river kingfisher sitting on branch
Poses
Make high quality image from pose features and text description
A woman practices yoga on a cross-legged sport mat
. . .
Text to image task
Spatially Grounded : Fabricate an image of a contemporary kitchen with a refrigerator at the location (50, 50) -> (100, 100), and stoves at the location (80, 80) -> (200, 200)
How-to-write : A white sign that says “morning”
Image to text tasks
Caption: Describe the given image
Long Caption: Describe the given image in very detail
VQA: Question: what time of the day is the photo taken?
Reasoning: Question: Does this passage describe the weather or the climate? Context: Figure: Des Moines. The temperature recorded …Please explain your answer.
CM3 Leon
Generated images
Generated images
Mornring
Generated text
A beautiful view of a city from across a river.
A view of tall buildings in a city. The photo is taken from a park across a river. We can see a bridge over the river.
Sunset time
Weather. Because the atmosphere is the layer of air that surrounds Earth. Both weather and climate tell you about the atmosphere. …

Figure 5: We perform fine-tuning on the CM3Leon model using a vast assortment of combined image and text tasks. Our retrieval augmented pretraining allows us to fine-tune the model effectively on a mixture of interleaved texts and images, as well as text-to-image and image-to-text tasks. We present some common model inputs for various tasks on the left, with the corresponding model outputs displayed on the right. Throughout the training process, we concatenate the model input and output and train them using the same objective that was utilized during the pretraining stage.

can be derived from user-uploaded images or sketches. We used ControlNet processing code on Shutterstock datasets to curate 7 million examples with features like canny edge, hed boundary, user sketching, human pose, and more (Zhang & Agrawala, 2023).

Spatially Grounded Image Generation allows the user to integrate spatial information into text prompts for image generation, with each object represented by discrete tokens. We used object detection datasets like MS-COCO, Openimage, and Object365 to compile 3 million training examples(Lin et al., 2014; Kuznetsova et al., 2020; Shao et al., 2019).

How-to-write task enables users to request the model to create signs or logos based on text prompts. We used an OCR detector to find suitable examples from Shutterstock datasets, resulting in 200,000 examples.

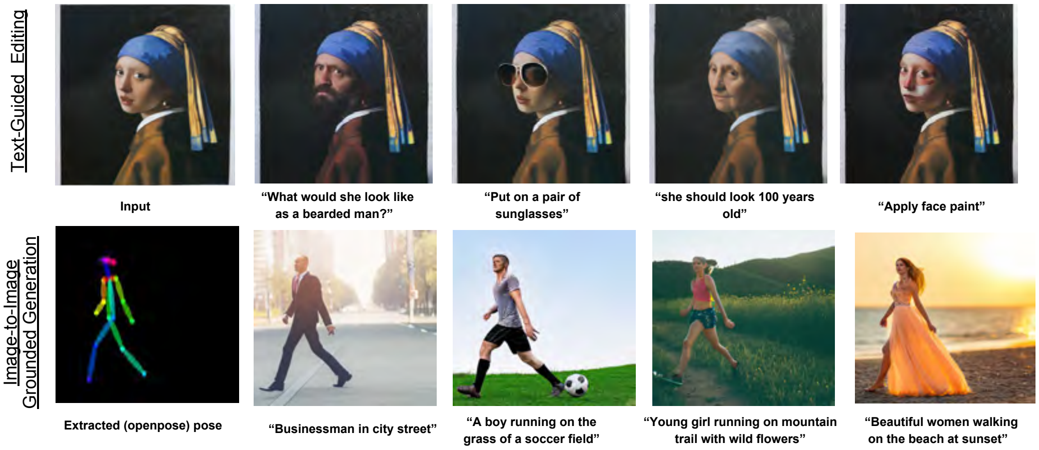

Text-Guided Editing
Input
“What would she look like as a bearded man?”
“Put on a pair of sunglasses”
“she should look 100 years old”
“Apply face paint”
Image-to-Image Grounded Generation
Extracted (openpose) pose
“Businessman in city street”
“A boy running on the grass of a soccer field”
“Young girl running on mountain trail with wild flowers”
“Beautiful women walking on the beach at sunset”

Figure 6: Qualitative examples of finetuned CM3Leon-7B model.

Results: We showcase qualitative examples of images produced by a fine-tuned CM3Leon-7B model, as depicted in Figure 6. All instances in text-guided editing and image-image-grounded generation utilize a task prefix. For instance, we precede every text-guided editing example with the phrase, "Edit the image following the text instruction," and every scribble generation example with "Create a high-quality image from children's scribble and text description," amongst others. The top row of Figure 6 presents text-guided image examples. We employ separate image CFG (1.5) and text CFG (7.5) values during decoding. This approach is crucial for producing edited images that mirror the original image and closely align with the text editing instruction. The second row in Figure 6 show Structure-Guided Image Editing examples. For decoding, we utilized a single CFG value of 3. Given identical input open pose features, our model can generate markedly distinct images that follow different text prompts while maintaining the same pose as in the input image. More examples in 15

4.2 Conditional Text Generation

We also include several vision-language tasks to teach CM3Leon to respond in text to various kinds of textual prompts conditioned on an image, such as visual question answering, long-form captioning, etc. We use the following 8 vision-language tasks: MS-COCO (Chen et al., 2015), Flickr30k (Young et al., 2014), Image Paragraph (Krause et al., 2017), Localized Narratives (Pont-Tuset et al., 2020), VQA2 Goyal et al. (2017), VizWiz (Gurari et al., 2018), OKVQA (Marino et al., 2019), and ScienceQA (Lu et al., 2022). We use multiple prompt templates for each task to make the model robust to prompt variations (more details on the templates in Table 5 of the Appendix).

Results: Table 2 presents the performance comparison of our SFT-CM3Leon model w.r.t. previous state-of-the-art (SoTA) such as Flamingo (Alayrac et al., 2022) and OpenFlamingo[3]. We show that our SFT-CM3Leon model achieves strong zero-shot performance on several vision-language tasks even though they saw significantly fewer text data ($\approx$ 3B tokens) compared to Flamingo (100B tokens) and OpenFlamingo (40B tokens). Notably, SFT-CM3Leon even beats Flamingo on the VizWiz task. Figure 16 presents our SFT-CM3Leon-7B model generations, given an image context and an instruction. The model is quite flexible with the instruction and can generate captions or answer a variety of questions. Further, the ability of to follow instructions is more evident in Figure 7 where the model can generate very long captions or reason over an image based on the given instruction.

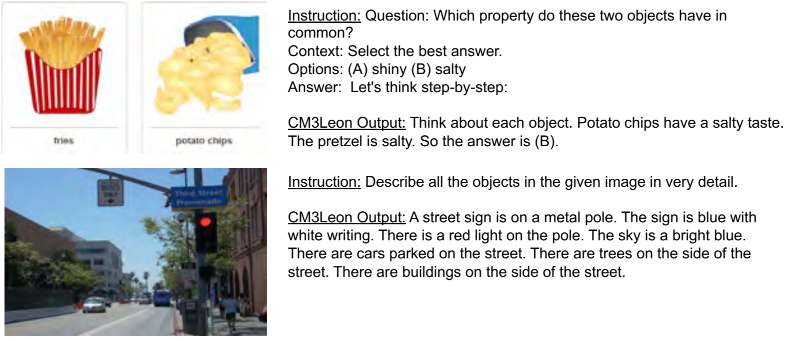
Model	MS-COCO CIDEr (test)	VQA2 Acc. (test-dev)	VizWiz Acc. (test-dev)	OKVQA Acc. (val)	Image Paragraph CIDEr (test)	VisDial NDCG (val)
OpenFlamingo-9B† (0-shot)	65.5	43.5	-	-	-	-
Flamingo-9B (0-shot)	79.4	51.8	28.8	44.7	-	48.4
SFT-CM3Leon-7B (0-shot)	61.6	47.6	37.6	23.8	10.5	22.6

Table 2: Comparison of our supervised fine-tuning (SFT) CM3Leon with state-of-the-art models in zero-shot and few-shot settings. † Reported numbers are all based on validation set.

5 Related Work

Diffusion Models Significant progress in the domain of text-to-image generation has been achieved through the use of diffusion models (Rombach et al., 2022; Nichol et al., 2021; Ramesh et al., 2022). The underlying mechanism involves sequentially adding noise to an image and then learning to reverse the noise based on provided text inputs or features (Luo, 2022). Diffusion models generally incorporate pretrained text or language representations such as the text encoder of the CLIP (Radford et al., 2021) image-text model or text encoders like T5 (Raffel et al., 2020). The recursive application of multi-resolution diffusion model (by employing multiple steps of super-resolution) has further enhanced their capability to generate high-quality images from text prompts, leading to state-of-the-art zero-shot non-retrieval based MS-COCO FID scores

[3] `https://laion.ai/blog/open-flamingo/`



Instruction: Question: Which property do these two objects have in common?
Context: Select the best answer.
Options: (A) shiny (B) salty
Answer: Let's think step-by-step:

CM3Leon Output: Think about each object. Potato chips have a salty taste. The pretzel is salty. So the answer is (B).

Instruction: Describe all the objects in the given image in very detail.

CM3Leon Output: A street sign is on a metal pole. The sign is blue with white writing. There is a red light on the pole. The sky is a bright blue. There are cars parked on the street. There are trees on the side of the street. There are buildings on the side of the street.

Figure 7: Qualitative examples showing our SFT-CM3Leon-7B model's generations for various long form generation tasks.

Retrieval Augmented Diffusion Models Conventional text-to-image models often struggle to capture the vast array of concepts and entities in the image domain. Methods like enabling retrieval during inference time can help address the complexity of these tail entities by delegating it to a retrieval step. Building on the work of Saharia et al. (2022), Chen et al. (2022) incorporates retrieval to enhance zero-shot MS-COCO FID scores, demonstrating further improvement in this area.

Autoregressive Token Models Significant advancements have been made in the field by utilizing LLMs over tokenized image representations (Esser et al., 2020; Ramesh et al., 2021). A widely-used approach in the field (Van Den Oord et al., 2017; Razavi et al., 2019; Esser et al., 2021) involves an initial stage of converting images into discrete latent variables through tokenization, which transforms a text-to-image generation problem into a sequence-to-sequence problem, thereby enabling subsequent application of LLM techniques (Ramesh et al., 2021; Gafni et al., 2022b).

Non-Autoregressive Token Models Although autoregressive models have benefited from extensive research in NLP, autoregressive decoding can be quite computationally expensive. Non-autoregressive models, such as Ghazvininejad et al. (2019), have been proposed in NLP and extended to text-to-image models, exemplified by Chang et al. (2023) which achieves state-of-the-art image generation performance and higher efficiency than diffusion or autoregressive models by employing masked modeling in discrete token space (non-autoregressively with iterative decoding).

Retrieval Augmented Autoregressive Token Models Token-based models face challenges akin to those encountered by non-retrieval augmented diffusion models. To address these issues, Yasunaga et al. (2022) suggested prefixing decoder-only text-to-image models, such as Ramesh et al. (2021); Aghajanyan et al. (2022), with statically retrieved instances during training, resulting in significant efficiency gains during the training process.

Our paper primarily concentrated on scaling this strategy.

6 Conclusion

We presented CM3Leon, a retrieval-augmented, token-based, decoder-only multi-modal language model that efficiently and flexibly generates and infills text and images. Our approach extends the scope of autoregressive models, demonstrating their potential to compete with and exceed diffusion models in terms of cost-effectiveness and performance. By integrating a retrieval-augmented pretraining stage with a diverse, large-scale Shutterstock dataset and a second multi-task supervised fine-tuning stage, CM3Leon demonstrates the benefits of a comprehensive training approach. Further enhanced by an innovative, self-contained contrastive decoding method, our model offers improved text and image generation quality. Our results support the value of autoregressive models for a broad range of text and image tasks, encouraging further exploration for this approach.

References

Armen Aghajanyan, Bernie Huang, Candace Ross, Vladimir Karpukhin, Hu Xu, Naman Goyal, Dmytro Okhonko, Mandar Joshi, Gargi Ghosh, Mike Lewis, et al. Cm3: A causal masked multimodal model of the internet. *arXiv preprint arXiv:2201.07520*, 2022.

Armen Aghajanyan, Lili Yu, Alexis Conneau, Wei-Ning Hsu, Karen Hambardzumyan, Susan Zhang, Stephen Roller, Naman Goyal, Omer Levy, and Luke Zettlemoyer. Scaling laws for generative mixed-modal language models. *arXiv preprint arXiv:2301.03728*, 2023.

Jean-Baptiste Alayrac, Jeff Donahue, Pauline Luc, Antoine Miech, Iain Barr, Yana Hasson, Karel Lenc, Arthur Mensch, Katherine Millican, Malcolm Reynolds, et al. Flamingo: a visual language model for few-shot learning. *Advances in Neural Information Processing Systems*, 35:23716–23736, 2022.

Gor Arakelyan, Gevorg Soghomonyan, and The Aim team. Aim, 6 2020. URL `https://github.com/aimhubio/aim`.

Tim Brooks, Aleksander Holynski, and Alexei A. Efros. Instructpix2pix: Learning to follow image editing instructions, 2023.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. In *NeurIPS*, 2020.

Huiwen Chang, Han Zhang, Jarred Barber, AJ Maschinot, Jose Lezama, Lu Jiang, Ming-Hsuan Yang, Kevin Murphy, William T Freeman, Michael Rubinstein, et al. Muse: Text-to-image generation via masked generative transformers. *arXiv preprint arXiv:2301.00704*, 2023.

Wenhu Chen, Hexiang Hu, Chitwan Saharia, and William W Cohen. Re-imagen: Retrieval-augmented text-to-image generator. *arXiv preprint arXiv:2209.14491*, 2022.

Xinlei Chen, Hao Fang, Tsung-Yi Lin, Ramakrishna Vedantam, Saurabh Gupta, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco captions: Data collection and evaluation server. *arXiv preprint arXiv:1504.00325*, 2015.

Patrick Esser, Robin Rombach, and Björn Ommer. Taming transformers for high-resolution image synthesis, 2020.

Patrick Esser, Robin Rombach, and Bjorn Ommer. Taming transformers for high-resolution image synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 12873–12883, 2021.

Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv preprint arXiv:2203.13131*, 2022a.

Oran Gafni, Adam Polyak, Oron Ashual, Shelly Sheynin, Devi Parikh, and Yaniv Taigman. Make-a-scene: Scene-based text-to-image generation with human priors. *arXiv preprint arXiv:2203.13131*, 2022b.

Marjan Ghazvininejad, Omer Levy, Yinhan Liu, and Luke Zettlemoyer. Mask-Predict: Parallel decoding of conditional masked language models. In *EMNLP*, 2019.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the V in VQA matter: Elevating the role of image understanding in visual question answering. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 6904–6913, 2017.

Danna Gurari, Qing Li, Abigale J Stangl, Anhong Guo, Chi Lin, Kristen Grauman, Jiebo Luo, and Jeffrey P Bigham. VizWiz grand challenge: Answering visual questions from blind people. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3608–3617, 2018.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. The curious case of neural text degeneration. In *ICLR*, 2020.

Srinivasan Iyer, Xi Victoria Lin, Ramakanth Pasunuru, Todor Mihaylov, Dániel Simig, Ping Yu, Kurt Shuster, Tianlu Wang, Qing Liu, Punit Singh Koura, et al. Opt-iml: Scaling language model instruction meta learning through the lens of generalization. *arXiv preprint arXiv:2212.12017*, 2022.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*, 2020.

Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 317–325, 2017.

Alina Kuznetsova, Hassan Rom, Neil Alldrin, Jasper Uijlings, Ivan Krasin, Jordi Pont-Tuset, Shahab Kamali, Stefan Popov, Matteo Malloci, Alexander Kolesnikov, Tom Duerig, and Vittorio Ferrari. The open images dataset v4. *International Journal of Computer Vision*, 128(7):1956–1981, mar 2020. doi: 10.1007/s11263-020-01316-z.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. Contrastive decoding: Open-ended text generation as optimization. *arXiv preprint arXiv:2210.15097*, 2022.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Pan Lu, Swaroop Mishra, Tanglin Xia, Liang Qiu, Kai-Wei Chang, Song-Chun Zhu, Oyvind Tafjord, Peter Clark, and Ashwin Kalyan. Learn to explain: Multimodal reasoning via thought chains for science question answering. *Advances in Neural Information Processing Systems*, 35:2507–2521, 2022.

Calvin Luo. Understanding diffusion models: A unified perspective. *arXiv preprint arXiv:2208.11970*, 2022.

Kenneth Marino, Mohammad Rastegari, Ali Farhadi, and Roozbeh Mottaghi. OK-VQA: A visual question answering benchmark requiring external knowledge. In *Proceedings of the IEEE/cvf conference on computer vision and pattern recognition*, pp. 3195–3204, 2019.

Alex 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. *arXiv preprint arXiv:2112.10741*, 2021.

Jordi Pont-Tuset, Jasper Uijlings, Soravit Changpinyo, Radu Soricut, and Vittorio Ferrari. Connecting vision and language with localized narratives. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part V 16*, pp. 647–664. Springer, 2020.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. *arXiv preprint arXiv:2103.00020*, 2021.

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

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. *arXiv preprint arXiv:2102.12092*, 2021.

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.

Ali Razavi, Aaron Van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. *Advances in neural information processing systems*, 32, 2019.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10684–10695, 2022.

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.

Maximilian Seitzer. pytorch-fid: FID Score for PyTorch. `https://github.com/mseitzer/pytorch-fid`, August 2020. Version 0.2.1.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

Mo Tiwari, Ryan Kang, Je-Yong Lee, Luke Lee, Chris Piech, Sebastian Thrun, Ilan Shomorony, and Martin Jinye Zhang. Faster maximum inner product search in high dimensions. *arXiv preprint arXiv:2212.07551*, 2022.

Aaron Van Den Oord, Oriol Vinyals, et al. Neural discrete representation learning. *Advances in neural information processing systems*, 30, 2017.

Michihiro Yasunaga, Armen Aghajanyan, Weijia Shi, Rich James, Jure Leskovec, Percy Liang, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. Retrieval-augmented multimodal language modeling. *arXiv preprint arXiv:2211.12561*, 2022.

Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014.

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. *arXiv preprint arXiv:2206.10789*, 2022.

Lvmin Zhang and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models, 2023.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

A Showcase Prompts

1. Chameleon and octopus, side by side, high quality render, drawing, professional.
2. A plush toy koala bear relaxing on a lounge chair and working on a laptop. The chair is beside a rose flower pot. There is a window on the wall beside the flower pot with a view of snowy mountains.
3. A photo of an astronaut riding a horse in the forest. There is a river in front of them with water lilies.
4. A teddy bear wearing a motorcycle helmet and cape is riding a motorcycle in Rio de Janeiro with Dois Irmãos in the background. dslr photo.
5. A black german shepherd wearing a red beret
6. An Armenian church on the surface of Mars, with Astronaut walking into the church, in Focus. Photo. Fantasy. Dramatic.
7. Armenian khachkars surrounded by pomegranates in a bright green forest.
8. A cat wearing sunglasses
9. A small cactus wearing a straw hat and neon sunglasses in the Sahara desert.
10. A close up photo of a human hand, hand model. High quality
11. A raccoon main character in an Anime preparing for an epic battle with a samurai sword. Battle stance. Fantasy, Illustration
12. A stop sign in a Fantasy style with the text "1991"

B Pre-Training

B.1 Data Visualizations

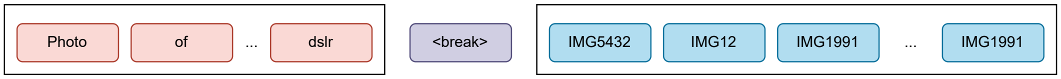


Figure 8: Visualization of the tokenization of one caption-image pair.

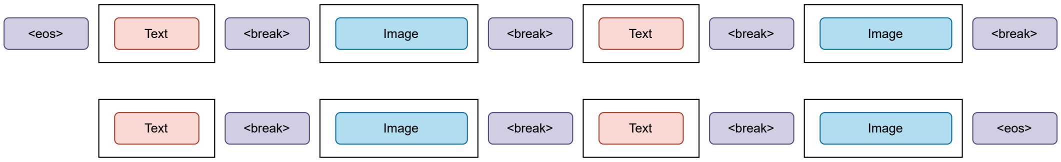


Figure 9: Visualization of the tokenization of a full training sample consisting of retrieved sampled and query caption-image pair.

B.2 Model Hyper-Parameters

Model size	# L	d_{model}	Seq Length	Batch	LR	Warmup Steps	# GPUs	# Tokens
350M	24	1024	4096	8M	6e-04	1500	256	1.4T
760M	24	1536	4096	8M	5e-04	1500	256	1.9T
7B	32	4096	4096	8M	1.2e-04	1500	512	2.4T

Table 3: **Model architecture details.** We report the number of layers (# L), embedding size (d_{model}), sequence length, batch size, peak learning rate (LR), learning rate warmup steps, number of GPUs used, and number of tokens consumed by each model.

Model	Resolution	Time
Imagen	256 × 256	9.1s
Imagen	1024 × 1024	13.1s
LDM (50 steps)	512 × 512	3.7s
LDM (250 steps)	512 × 512	18.5s
Parti (3B)	256 × 256	6.4s
MUSE (3B)	256 × 256	0.5s
CM3Leon (7B, BF16)	256 × 256	11.8s
CM3Leon (7B, INT8)	256 × 256	9.1s

Figure 10: Inference latency for several models.

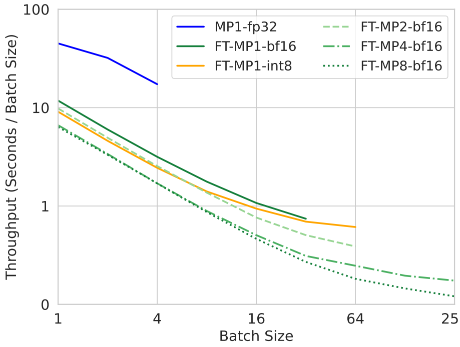


Figure 11: Inference throughput of CM3Leon-7B for generating images, without retrieval, across different model parallelism (MP), FasterTransformer (FT) implementation, data type (DType) and batch sizes

C Inference Latency and Throughput

D Image Generation

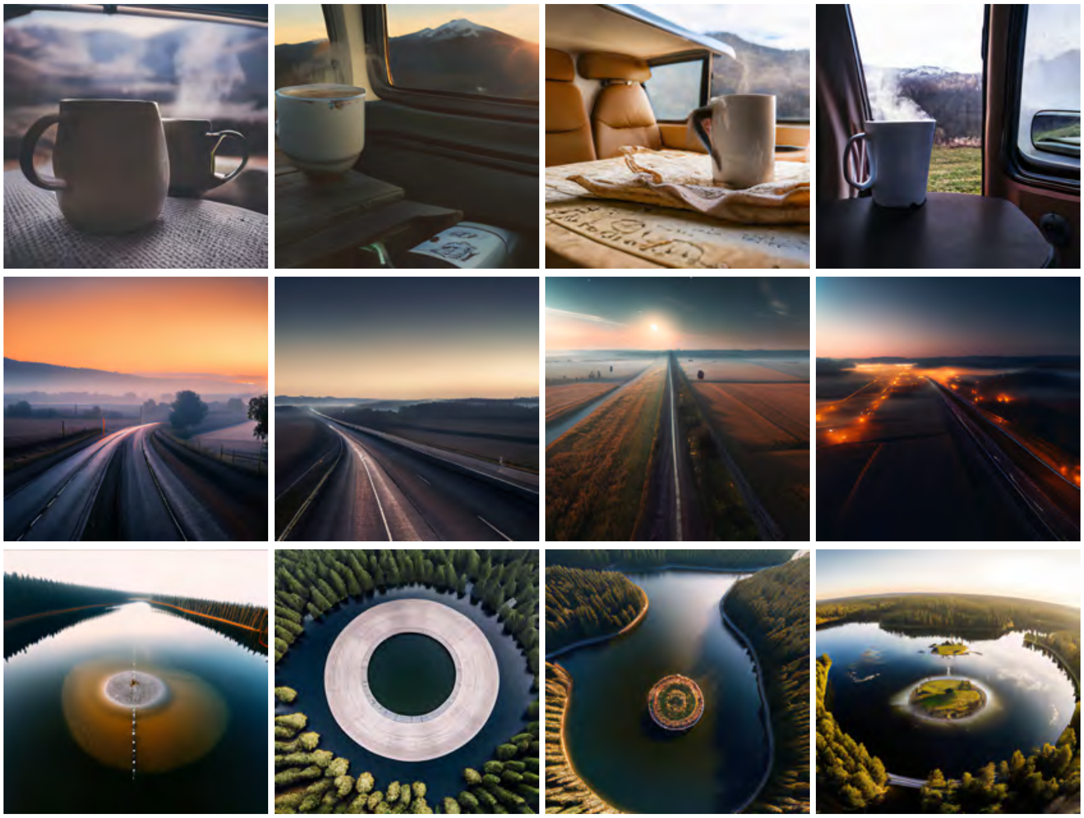


Figure 12: Top to bottom prompts: A steaming cup of coffee with mountains in the background. Resting during road trip., beautiful, majestic road during sunset. Aesthetic., *small circular island in the middle of a lake. Forests surrounding the lake. High Contrast.*

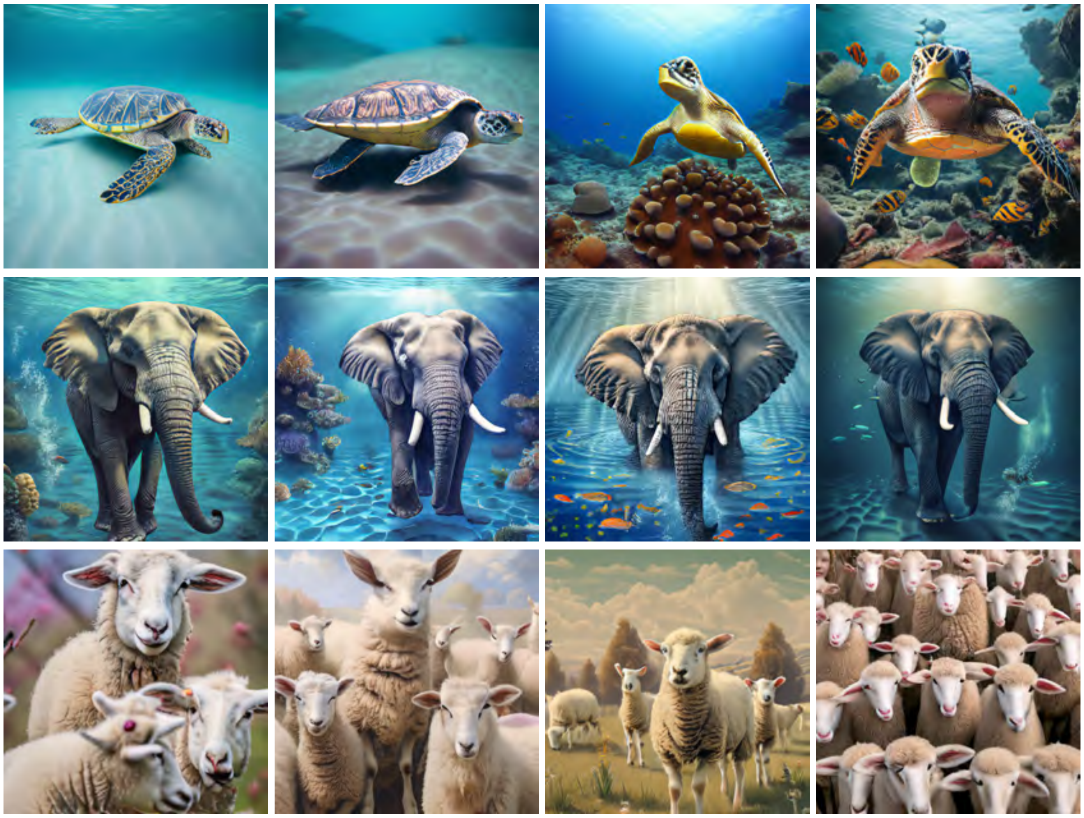


Figure 13: `turtle swimming underwater. aesthetic. Fantasy.`, elephant swimming underwater. aesthetic. Fantasy.
`, flock of sheep. aesthetic. Fantasy.`

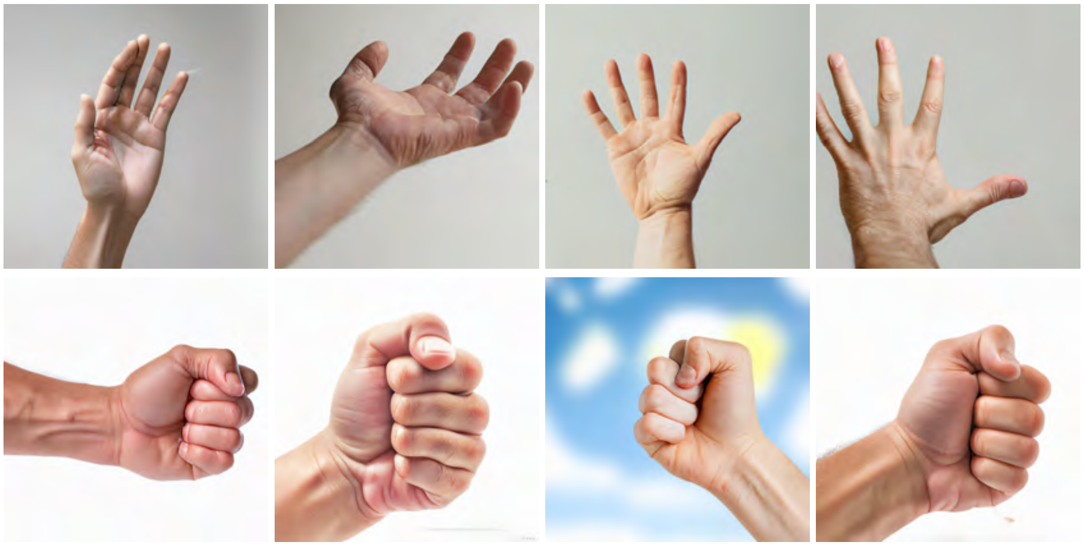


Figure 14: `open hand, hand model. 4k. white background`, `fist, hand model. 4k. white background`

E Supervised Fine Tuning

E.1 Hyper-Parameters

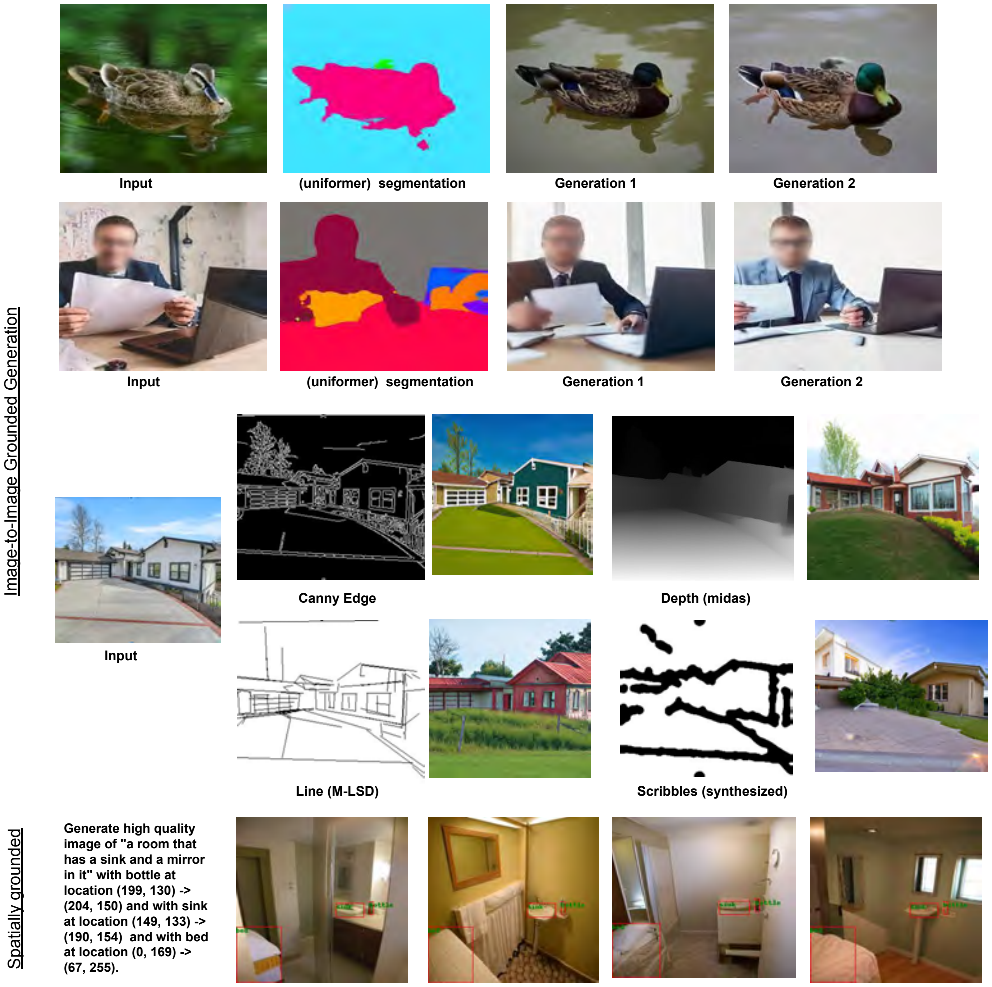
To maintain a balanced dataset during training, we implemented an up/down sampling strategy with a threshold of 3/0.3. This process was executed on the 760M and 7B models using 64 and 128 80GB A100s, respectively. We assembled our training examples into sequences of length 4096. Preliminary experiments were conducted to identify optimal learning rates from a range of $1e{-}5, 3e{-}5, 5e{-}5, 1e{-}4$ and per-GPU batch sizes from $4, 8, 16$ using our validation split. The selected hyperparameters are cataloged in Table 4. Throughout the fine-tuning phase, our models processed approximately 30 billion tokens.

Model	# GPUS	Seq Length	Batch Size	LR	Warm-up Steps	# Tokens
CM3Leon-760m	64	4096	2M	5e-05	150	30B
CM3Leon-7b	128	4096	2M	5e-05	150	30B

Table 4: Fine-tuning parameters for CM3Leon models

E.2 Training Data Breakdown

E.3 More Qualitative Samples

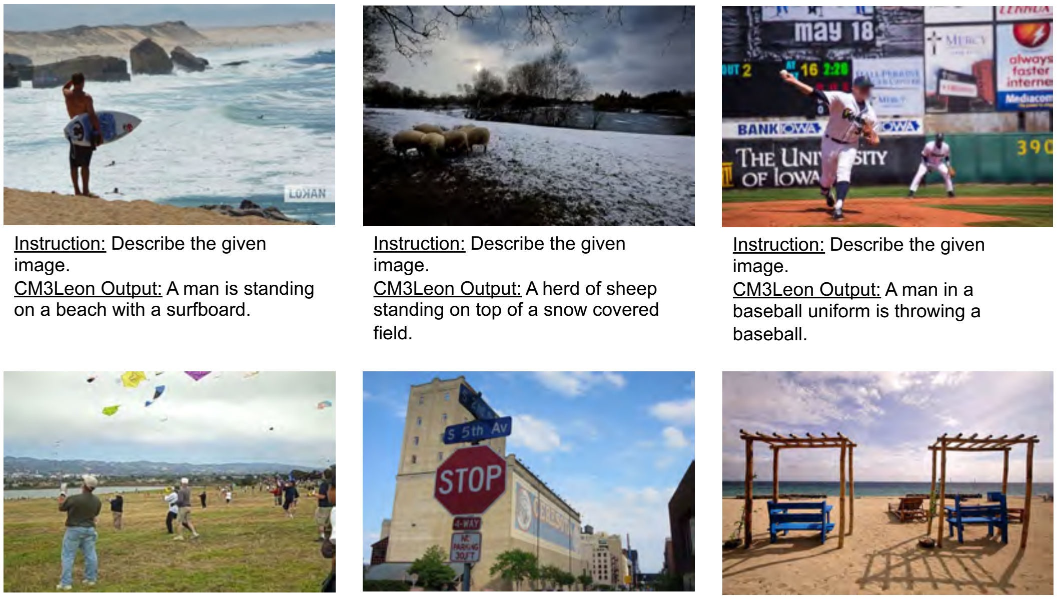


Generate high quality image of "a room that has a sink and a mirror in it" with bottle at location (199, 130) -> (204, 150) and with sink at location (149, 133) -> (190, 154) and with bed at location (0, 169) -> (67, 255).

Figure 15: Qualitative examples of finetuned CM3Leon-7b model. Human faces are blurred to remove PII information.

Dataset	Template	# Examples
	Image Focused Datasets	
InstructPix2Pix	Edit first image following the instruction <break> {image1} <break> edit instruction <break> {image 2}	127k
OCR	draw "{ocr_content}" <break> {image}	300k
Object Detection	Generate high quality image of {caption} with segmentations {obj1} at {loc1}, {obj2} at {loc2} ... <break> {image}	3M
Edge-to-Image	Make high quality image from canny edge features <break> {edge image} <break> {caption} <break> {image}	1M
Seg-to-Image	Make high quality image from a segmentation map <break> {seg image} <break> {caption} <break> {image}	1M
Hed-to-Image	Make high quality image from hed features <break> {seg image} <break> {caption} <break> {image}	1M
Pose-to-Image	Make high quality image from openpose features <break> {seg image} <break> {caption} <break> {image}	142k
Depth-to-Image	Make high quality image from depth features <break> {depth image} <break> {caption} <break> {image}	1M
Norm-to-Image	Make high quality image from 3D norm features <break> {depth image} <break> {caption} <break> {image}	1M
Scribbe-to-Image	Make high quality image from children's scribbles <break> {scribble image} <break> {caption} <break> {image}	500k
	Text Focused Datasets	
COCO Captioning (Chen et al., 2015)	{caption} <break> {image} Describe the given picture. {caption} <break> {image}	591k
Flickr30k (Young et al., 2014)	{caption} <break> {image} Describe the given picture. {caption} <break> {image}	144k
Image Paragraph (Krause et al., 2017)	Describe the given picture in very detail. {caption} <break> {image} Describe all the objects in the given image in very detail. {caption} <break> {image} Generate a long caption for the given image. {caption} <break> {image}	14k
Localized Narratives (Pont-Tuset et al., 2020)	Describe the given picture in very detail. {caption} <break> {image} Generate a long narration of what is happening in the given image. {caption} <break> {image} Generate a long caption for the given image. {caption} <break> {image}	164k
VQA2 (Goyal et al., 2017)	Question: {question} Answer: {answer}. <break> image Question: {question} [newline] {answer} <break> image Question: {question} The answer is {answer}. <break> image	1.3M
VizWiz (Gurari et al., 2018)	Question: {question} Answer: {answer}. <break> image Question: {question} [newline] {answer} <break> image Question: {question} The answer is {answer}. <break> image	92k
OKVQA (Marino et al., 2019)	Question: {question} Answer: {answer}. <break> image Question: {question} [newline] {answer} <break> image Question: {question} The answer is {answer}. <break> image	26k
ScienceQA (Lu et al., 2022)	Question: {question} [newline] Context: {context} [newline] Options: {choices_text} [newline] Answer: {answer}. <break> {image} Question: {question} [newline] Context: {context} [newline] Options: {choices_text} [newline] Answer: Let's think step-by-step: { explanation} So the answer is {answer}. <break> {image}	6k

Table 5: Details of the datasets and their prompt templates used in our supervised fine-tuning of CM3Leon models.



Instruction: Describe the given image.
CM3Leon Output: A man is standing on a beach with a surfboard.

Instruction: Describe the given image.
CM3Leon Output: A herd of sheep standing on top of a snow covered field.

Instruction: Describe the given image.
CM3Leon Output: A man in a baseball uniform is throwing a baseball.

Instruction: Question: What are people flying?
CM3Leon Output: Kites

Instruction: Question: What sign is on the street?
CM3Leon Output: stop

Instruction: Question: What color is the bench?
CM3Leon Output: blue

Figure 16: Qualitative examples showing our SFT-CM3Leon-7B model's generations for image captioning and visual question answering tasks. Human faces are blurred to remove PII information.