SynthCLIP: Are We Ready for a Fully Synthetic CLIP Training?

Hasan Abed Al Kader Hammoud[1]* Hani Itani[1]* Fabio Pizzati[2]
Philip H.S. Torr[2] Adel Bibi[2] Bernard Ghanem[1]
[1]KAUST [2]University of Oxford

Abstract

We present SynthCLIP, a novel framework for training CLIP models with entirely synthetic text-image pairs, significantly departing from previous methods relying on real data. Leveraging recent text-to-image (TTI) generative networks and large language models (LLM), we are able to generate synthetic datasets of images and corresponding captions at any scale, with no human intervention. With training at scale, SynthCLIP achieves performance comparable to CLIP models trained on real datasets. We also introduce SynthCI-30M, a purely synthetic dataset comprising 30 million captioned images. Our code, trained models, and generated data will be released as open source on https://github.com/hammoudhasan/SynthCLIP.

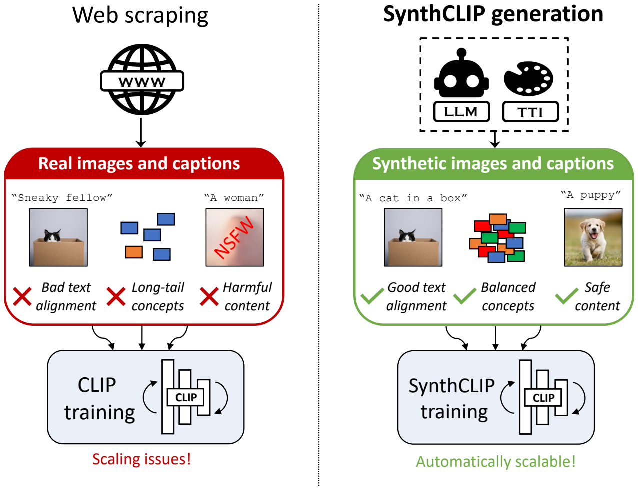


Figure 1. **Advantages of SynthCLIP.** Collecting text-image pairs from the internet often presents challenges: captions may not accurately match the images, specific classes may have limited representation due to scarcity, and there is a risk of encountering harmful content. We propose SynthCLIP, an approach for generating text-image pairs, effectively to overcome these issues. It ensures that the generated images have corresponding descriptive captions, and it enforces a balanced representation of classes. Moreover, we can benefit from safety checks in state-of-the-art LLM and TTI. Our approach is automatically scalable, allowing to match performance of real data with no human intervention in the data generation process.

1. Introduction

Self-supervised training techniques [3, 5, 17] are fundamental for all recently released foundation models, since they make use of vast amount of data without incurring a large annotation cost. In particular, contrastive representation learning [55] has been successfully employed to extract joint embeddings for heterogeneous data modalities. By harnessing multi-modal training data, CLIP [46] provides a common representation that effectively links visual and linguistic information. Today, CLIP encoders are included in a wide range of applications, spanning from zero-shot image understanding [37, 47], to style transfer [27], and robotics control [60], among others.

However, training CLIP requires large-scale text-image datasets, that are often collected from the web. Unfortunately, retrieving captioned images from the internet presents notable challenges. Firstly, web data is often noisy; a mismatch between images and their textual descriptions may impact the quality of the learned representations [29]. Secondly, the frequency of certain visual and textual elements varies naturally, leading to the emergence of long-tail distributions. Lastly, despite safety measures, gathering images from the web in large numbers poses difficulties in filtering out inappropriate or copyrighted content, which raises safety concerns[1]. All these, together, make scaling web-crawled text-image datasets surprisingly difficult, due to the required control on the collected data [24, 31, 44]. On the other hand, synthetic data can resolve these issues natively. While there have been attempts to train CLIP models with either synthetic images [64] or captions [11, 29], they always relied on at least one real data modality, limiting the

*Equal Contribution

[1] We report a recent article in mainstream news on the topic.

scalability of the training dataset to the number of either real images or captions.

In this paper, we investigate whether it is possible to train CLIP models on fully generated text-image data, in the form of captioned images, and match the performances of CLIP trained on real data. To achieve this goal, we introduce SynthCLIP, a novel approach for training CLIP models using exclusively large-scale synthetic data. We propose a pipeline that jointly leverages existing text-to-image models (TTI) and large language models (LLM) to produce text-image pairs. The captioned images are generated in an end-to-end fashion, starting from a large list of concepts necessary to guarantee variability of the synthesized data. We use the LLM to produce captions starting from sampled concepts, and then synthesize their corresponding images using TTI models. This brings a significant novel advantage, unprecedented in literature: we can generate data at any scale, arbitrarily increasing the size of training data depending only on computational power, *with no human intervention*. Moreover, compared with training on real data, our pipeline ensures that captions are well associated with the corresponding images, allowing for singificant performance gains on vision-language tasks, such as image or text retrieval. Furthermore, sampling from a large pool of concepts enables us to avoid long-tail distributions in the synthesized dataset. Finally, we benefit from the included security checks in state-of-the-art LLM and TTI to filter out potentially harmful content from the generated training data. A visual comparison between CLIP and SynthCLIP is shown in Figure 1. Our contributions in this paper are threefold:

1. We propose SynthCLIP, a novel approach for end-to-end generation of synthetic language and vision data for CLIP training, automatically scalable to any desired dataset size.
2. We show that when running our data generation at scale, we are able to match the performance of CLIP pre-trained on real text-image pair datasets.
3. We release SynCI-30M, an entirely synthetic dataset produced using our generation pipeline, composed of 30 million pairs of images and corresponding captions. We also release models trained on different synthetic dataset scales, and the code to generate the dataset.

2. Related Work

Representation Learning. Early works in self supervised representation learning on images used pre-text tasks such as inpainting, jigsaw puzzle solving, and image rotation prediction [15, 41, 43]. More recent works such as masked autoencoder (MAE) [17] uses a masked image patch prediction task to learn visual representations. Instead, SimCLR [5] leverages contrastive learning to maximize the similarity between two augmented views of the same image. On the other hand, CLIP [45] and other similar works [39, 75] use contrastive learning to learn joint visual and textual representations. Language-image pre-training necessitates high quality text-image pairs. Its core idea is to maximize the similarity between encoded textual and image representation. In this work, we study the possibility of generating end-to-end synthetic text-image pairs for training CLIP like models starting from simple concepts only.

Synthetic Data Synthetic data has been used in many machine learning fields ranging from audio [51] to language [32, 70] and vision [21, 66, 76]. In computer vision, synthetic data have been used to improve models' performance on several downstream tasks such as semantic segmentation [6, 48, 50], object detection [23], and image classification [59, 74]. Recent works have explored the use of synthetic data from from text-to-image models, to augment training on real data [1, 18, 53]. Yu et al. [73] uses a framework to generate synthetic images, increasing the diversity of existing datasets. All these assume knowledge about object classes in the downstream task, and work with images only. Most recently, StableRep [64] showed that synthetic images generated from Stable Diffusion can be used to train self supervised methods and match the performance of training on real images. This uses real captions of common datasets used to train language-vision models as prompts for Stable Diffusion, which limits the scalability of the generated dataset.

Synthetic Captions. Recent works emphasize the importance of high quality and aligned text-image pairs when training CLIP models, and propose synthetic caption generation pipeline for improving it. VeCLIP [29] and CapsFusion [72] propose pipelines to produce better aligned captions. Both start with a captioning model such as BLIP [33] or LLaVA [36], to produce a semantically and visually enriched synthetic caption. However, captioning models suffer from over-simplification and lack world knowledge, hence they can be effectively compensated by the usage of an LLM [29, 72]. LaCLIP [11] propose to improve the text branch of CLIP by leveraging an LLM to provide multiple rewrites of the same caption to use in contrastive learning. While this improves CLIP performance on downstream tasks, it may not reflect the content of the image due to hallucinations [11]. All the reported works assume availability of real data, instead we introduces a fully synthetic pipeline for data generation, allowing arbitrary scalability.

3. Methodology

In this section, we present SynthCLIP, the first approach for CLIP training where both textual and image modalities are generated synthetically. In Figure 2, we summarize SynthCLIP synthetic data generation and training pipeline. First, we start with a concept bank that contains many raw visual concepts, *i.e.* words that can be associated to their

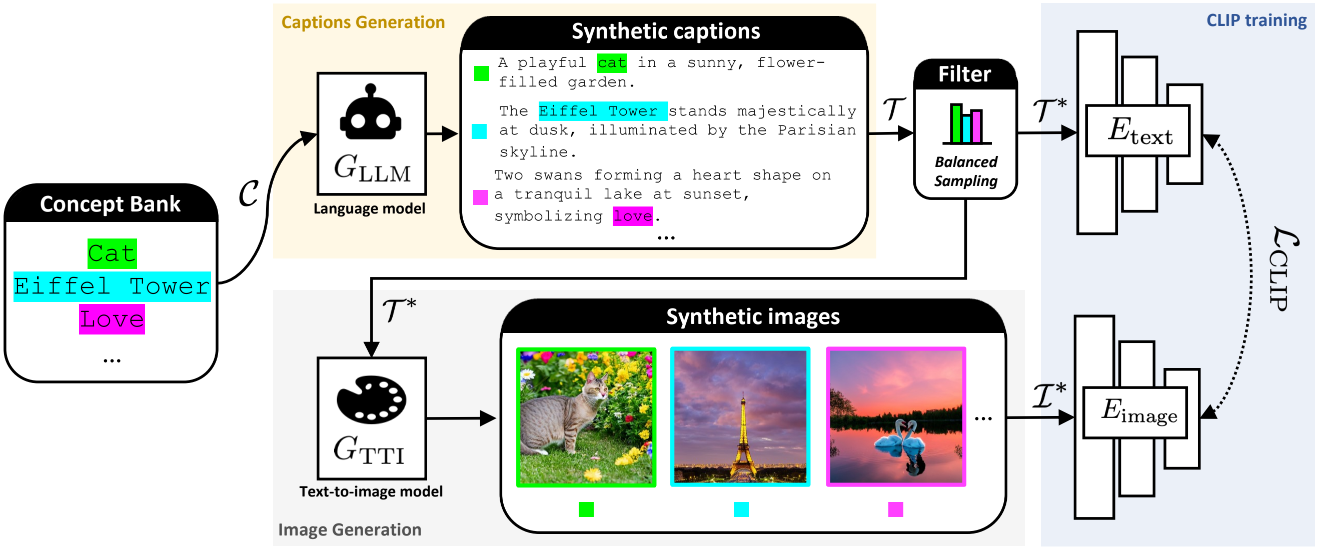


Figure 2. **Pipeline Overview.** From a set of concepts $\mathcal{C}$ (left), we obtain a set of synthetic captions $\mathcal{T}$ with an LLM, further refined to $\mathcal{T}^*$ by a filtering operation which subsamples $\mathcal{T}$ using balanced sampling (top). The generated captions are then used to prompt a text-to-image model, obtaining synthetic images aligned with the prompt (bottom). Finally, we train CLIP encoders on the generated synthetic text-image pairs. (right)

corresponding representations in images. This broad definition covers either common objects, items, and animals (*e.g.* "cat"), proper nouns and specific elements (*e.g.* "Eiffel Tower") and intangible items associated to specific visual characteristics (*e.g.* "love", that is often represented with stylized representations of hearts). A large language model is then prompted to generate captions for all the concepts in the concept bank, leading to a set of synthetically generating captions describing a variety of concepts (Section 3.1). The generated captions are then filtered to a smaller corpus of captions for improved performance (Section 3.2). The filtered captions are then passed to a text-to-image model to generate corresponding images (Section 3.3). After obtaining our synthetic $\{\text{caption}, \text{image}\}$ pairs, a standard CLIP training is carried on the generated data, obtaining the language and text encoders that can be used for downstream tasks (Section 3.4). We next describe each step in details.

3.1. Step 1: Concept-based Captions Generation

The first stage of our pipeline involves the generation of synthetic image captions, that we later aim to use as prompt for text-to-image generators. To achieve this, we utilize an LLM conditioned on our concept bank. The model is prompted to generate captions that describe a scene related to a chosen concept. In our process of generating these captions, we experimented with various prompting techniques, discovering that conditioning the LLM to focus on a particular concept leads to more diverse captions. Indeed, concept conditioning ensures that the LLM does not just repeatedly produce captions about a limited set of concepts that are over represented in the training dataset. In other words, this approach helps prevent the model from becoming biased towards certain concepts and encourages a broader spectrum of caption generation. Limited concept diversity would hinder the CLIP training, since contrastive learning highly benefit from variability and more concept coverage [69]. Hence, diversity is a requirement for the scalability of SynthCLIP.

We start by introducing our concept bank $\mathcal{C}$ composed by $N_\mathcal{C}$ concepts. We observe that N_C deeply influences CLIP performance, and we investigate this effect in Section 4.3. Unless otherwise stated, we use the MetaCLIP concept bank [69], that contains over 500,000 concepts drawn from WordNet Synsets and Wikipedia common unigrams, bigrams, and titles. We then focus on prompt engineering, a critical aspect for generating effective captions for text-to-image generation. Image generators are known to be sensitive to the quality of the input prompt [16], which is often a brief text description capturing the characteristics of the desired image. So, we set specific requirements to ensure that the prompts generated by the LLM are well-suited for the subsequent image generation:

(1) Focus on a Single Concept: Each generated caption should center around a single concept, presented in a clear and coherent context.

(2) Brevity and Clarity: The prompts need to be concise yet grammatically correct. The goal is to avoid overly complex or vague inputs that could lead to ambiguous or incorrect images.

(3) Prompt-Only Generation: Our aim is to have the LLM

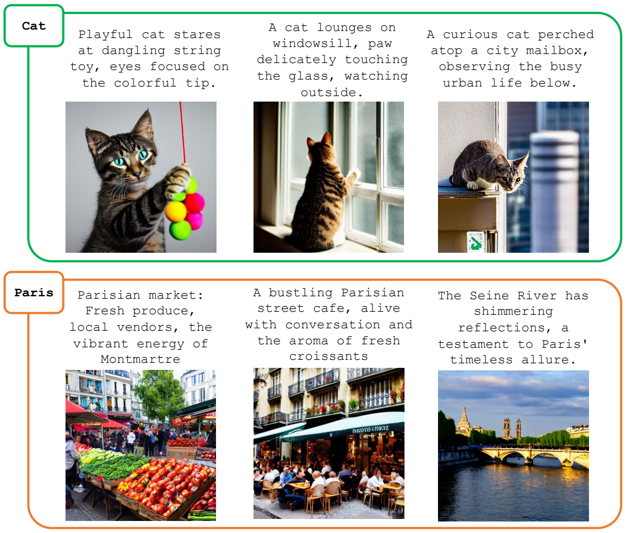

Cat
Playful cat stares at dangling string toy, eyes focused on the colorful tip.
A cat lounges on windowsill, paw delicately touching the glass, watching outside.
A curious cat perched atop a city mailbox, observing the busy urban life below.
Paris
Parisian market: Fresh produce, local vendors, the vibrant energy of Montmartre
A bustling Parisian street cafe, alive with conversation and the aroma of fresh croissants
The Seine River has shimmering reflections, a testament to Paris' timeless allure.

Figure 3. **Generation samples.** We show generated captions and images pairs for the concepts “`cat`” and “`Paris`”. Our generation pipeline provides both high variability and realistic contextual placement of input concepts.

generate prompts without engaging in further reasoning or elaboration. This approach not only saves computational resources but also simplifies the parsing process.

Assuming $c \in \mathcal{C}$, our designed prompt is:

> Your task is to write me an image caption that includes and visually describes a scene around a concept. Your concept is c. Output one single grammatically correct caption that is no longer than 15 words. Do not output any notes, word counts, facts, etc. Output one single sentence only.

Formally, we define our LLM generator as G_{LLM} and the prompt as p. Hence, the set of generated captions is $\mathcal{T} = \{t_{c,n} \sim G_{\text{LLM}}(p, c)\}, \forall c \in \mathcal{C}, \forall n \in \{1, 2, ..., N\}$ where N is the number of desired captions for each concept. By looking at the captions in Figure 3, we show how this mechanism results in highly variable contextual placement of each concept.

3.2. Step 2: Captions filtering

When generating captions conditioned on a specific concept c, it is typical for other concepts $c' \neq c, c' \in \mathcal{C}$ to appear within the same caption. This is expected, since even when a sentence is focused on a single concept, other related concepts often emerge within the context of the described scene. For example, if $c =$ “`bird`”, a generated caption might be “`a bird is resting on a tree`”, introducing an additional concept $c' =$ “`tree`”. This LLM-specific behavior may create imbalances in the generated data for CLIP training, which instead benefits from the usage of a balanced amount of concepts [69].

To address this, we propose creating a balanced ensemble of captions, $\mathcal{T}^*$, applying the balancing method proposed in MetaCLIP [69] to our setting. It consists of two stages, substring matching and balanced sampling. Substring matching determines which concepts from $\mathcal{C}$ appear in each caption within $\mathcal{T}$. This enables us to measure the real frequency of each described concept across the synthesized captions. Balanced sampling is then employed to subsample captions $\mathcal{T}^*$ from $\mathcal{T}$. It increases the probability of selecting captions with long tail concepts, and thresholds that of sampling captions with frequently occurring concepts. This yields a subset of captions where both frequent and long tail concepts are adequately represented. Therefore, this approach ensures a diverse and task-agnostic captions set suitable for foundation model pre-training. By sizing the parameters of balanced sampling, we are able to choose the size of the subset $\mathcal{T}^*$. For more details, we refer to [69].

3.3. Step 3: Image Generation

Having successfully created a balanced set of synthetic captions $\mathcal{T}^*$, our next step is to generate the corresponding images. For this, we utilize a text-to-image generator G_{TTI}. We choose Stable Diffusion [49] for this purpose, due to its open-source availability and relatively lower computational demands. For each caption in our set $\mathcal{T}^*$, we generate a corresponding image. This process results in a collection of images, $\mathcal{I}^* = \{x_k \sim G_{\text{TTI}}(t_k)\}$, where each x_k is an image synthesized from the caption $t_k \in \mathcal{T}^*$. In Figure 3, we show how we generate highly aligned images which correctly capture the described scene and complement it with related realistic information. This proves the efficacy of our caption generation pipeline, leading to appropriate image generation.

3.4. Step 4: CLIP Training

Finally, we use the synthetic text-image pairs to train a CLIP model, exploring how effectively a model can learn from entirely synthetic data. We train two encoders, each one dedicated to either the image or text modality, defined as E_{image} and E_{text}, respectively. We follow the standard CLIP training pipeline [45], by applying a contrastive loss on the image and text representations through the encoders. Formally, we extract representations $h = E_{\text{image}}(x_k), x_k \in \mathcal{I}^*$ and $z = E_{\text{text}}(t_k), t_k \in \mathcal{T}^*$, and train by minimizing the CLIP loss $\mathcal{L}_{\text{CLIP}}(h, z)$.

Safety considerations for CLIP training. SynthCLIP is trained exclusively on synthetic data, which will increase the safety standard of vision-language encoders. Indeed, data collection from the web is exposed to unsafe or offending concepts [56], which are difficult to filter. Contrarily, our generation pipeline natively exploits an aligned LLM for safe captions generation [58]. Moreover, text-to-image generators often include unsafe content detectors [61], that are triggered in presence of unwanted sexual or violent generated images.

4. Experiments

In this section, we evaluate the performance of SynthCLIP. We start by introducing the experimental setup in Section 4.1, including details about datasets, generation models, and downstream tasks. Section 4.2 benchmarks SynthCLIP against baselines trained on real data on multiple tasks. Finally, Section 4.3 encompasses complementary experiments on the impact of the size of the concept bank, $\mathcal{C}$, as well as several ablations that test various components of SynthCLIP.

4.1. Experimental Setup

Downstream Tasks We use five different downstream tasks to assess performance. For ease of evaluation, we categorize the downstream tasks into two categories; **(1) Vision Tasks** and **(2) Vision-Language Tasks**. The former focuses on evaluating the capabilities of the frozen vision encoder E_{image} only, *i.e.*, linear probing and few-shot classification. The latter evaluates the synergy between the image encoder E_{image} and text E_{text} together. The tasks used for evaluation vary from *image retrieval* , *text retrieval*, and *vision-language zero-shot classification tasks* following the original CLIP evaluation [45]. Since our evaluation pipeline consists of several tasks whose metrics can behave differenty with scaling, we aggregate performance across all tasks using the Δ_{MTL} metric [65], where a model with positive Δ_{MTL} indicates an overall better performance compared to a reference baseline.

Datasets We use the real datasets CC3M [57] (3×10^6 samples) and CC12M [4] (8.8×10^6 samples[2]). Real images come at different resolutions, so we resize the shorter edge of the images to 256px. For SynthCLIP, we generate an entirely synthetic dataset, that we call SynthCI (**Synth**etic **C**aptions-**I**mages) at different scales (number of samples). We refer to SynthCI-3M for a version of SynthCI where $\mathcal{T}^*$ and $\mathcal{I}^*$ include 3×10^6 captions and images, respectively. For zero-shot evaluation we use ImageNet [52], for linear probing and few shot we use CIFAR10 [25], CIFAR100 [26], Aircraft [38], DTD [8], Flowers [40], Pets [42], SUN397 [68], Caltech-101 [12] and Food-101 [2], and for image and text retrieval we use MSCoco [35], Flickr8K [20] and Flickr30K [71].

Caption & Image Generation Models For caption generation, we use Mistral-7B-Instruct V0.2 [22] with temperature 0.7 and top-p set to 0.95. We also set the presence and frequency penalties at 1. For image synthesis, we use Stable Diffusion v1.5 [49] with classifier-free guidance set to 2 and 50 Denoising Diffusion Implicit Models (DDIM) steps following Tian et al. [64]. The images are generated at 512×512px and then stored to disk at 256×256px. It takes 0.9 seconds to generate and save one image on NVIDIA A100 GPU. Image generation was performed on a 48 A100-80GB GPUs cluster.

Model Architecture & Training Parameters All trained CLIP models use a ViT-B/16 [9] as E_{image} and the default text encoder from CLIP [45] as E_{text}. E_{image} and E_{text} are trained for 40 epochs with a global batch size of 4096, a learning rate of 5×10^{-4}, weight decay of 0.5, cosine scheduler, and 1 warmup epoch. We use random resized crop with scale $0.5 - 1.0$ as data augmentation. We use the codebase of SLIP [39] as is to train all the CLIP models on 16 NVIDIA-V100-32GB GPUs.

4.2. Benchmark Evaluation

Performance on the same data scale We evaluate the effectiveness of our entirely synthetic data generation pipeline for training CLIP models compared to training on real data. We use CLIP [45] trained on CC3M and CC12M as baselines. We first train SynthCLIP on two versions of SynthCI each matching the data scale of CC3M and CC12M, which we call SynthCI-3M and SynthCI-8.8M, respectively. We report the performance on vision tasks in Table 1a and vision-language tasks in Table 1b, aggregating all metrics with Δ_{MTL} [65] in Table 1c. As visible in Table 1c, we obtain lower performance when both datasets are composed by 3×10^6 samples (**-5.60%**) and 8.8×10^6 samples (**-15.0%**), compared to the corresponding real data training with the same dataset size (CC3M and CC12M, respectively). This is expected: considering that real and synthetic data differ in distribution, while training on synthetic samples and testing on real ones, we incur in a distribution shift, which ultimately harms performance [10, 76].

Scaling SynthCLIP Our objective now is to compensate the effects of the distribution shift, to match performance obtainable by training CLIP on real data. We plan to do so by scaling the size of SynthCI, since it is well known that bigger training datasets help to increase performance [46]. However, while scaling real datasets necessitates custom collection pipelines from different sources and data curation, we exploit the great advantage of our data synthesis pipeline, *i.e.* the capability to scale the size of the training data with no human intervention. In practice, we simply let our generation script run for longer, and re-train SynthCLIP on the larger SynthCI version obtained doing so. In particular, we report performance for SynthCLIP trained on $\{10 \times 10^6, 20 \times 10^6, 30 \times 10^6\}$ SynthCI samples, finally matching with 30 million samples the performance of the biggest model we trained on real data (CLIP on CC12M),

[2] The original CC12M is composed of 12M samples. In December 2023, only 8.8M images were available at the linked URLs.

	Method	**Data**	**Samples** ($\times 10^6$)	Synth. data	CIFAR10	CIFAR100	Aircraft	DTD	Flowers	Pets	SUN397	Caltech-101	Food-101	Avg
Linear Probing	CLIP	CC3M	3	✗	81.8	62.7	34.7	57.3	84.1	60.5	54.3	75.6	58.7	63.3
		CC12M	8.8	✗	**91.3**	**73.0**	**48.5**	69.6	92.2	**81.3**	68.9	**88.2**	**77.7**	**76.7**
	SynthCLIP	SynthCI-3M	3	✓	80.9	60.7	36.3	60.6	85.9	59.3	55.4	73.8	60.7	63.7
		SynthCI-8.8M	8.8	✓	85.9	65.9	44.0	68.7	90.0	71.8	64.2	83.0	71.6	71.7
		SynthCI-10M	10	✓	86.4	67.8	44.9	68.8	90.4	71.9	64.8	85.2	72.2	72.5
		SynthCI-20M	20	✓	87.7	68.5	47.0	70.7	92.1	75.9	68.3	86.3	75.3	74.6
		SynthCI-30M	30	✓	88.0	69.6	45.3	**71.0**	**92.4**	77.6	**69.0**	86.2	76.0	75.0
Few-shot	CLIP	CC3M	3	✗	61.4	70.9	45.2	73.2	93.0	71.0	93.3	91.6	68.2	74.2
		CC12M	8.8	✗	**80.3**	**83.5**	55.7	82.0	96.8	85.5	**96.9**	**97.4**	**86.3**	**84.9**
	SynthCLIP	SynthCI-3M	3	✓	57.6	68.8	47.2	74.3	93.5	70.8	93.5	89.9	68.3	73.8
		SynthCI-8.8M	8.8	✓	62.4	73.3	56.9	79.6	95.7	80.9	95.8	95.1	78.4	79.8
		SynthCI-10M	10	✓	67.0	75.1	59.3	80.4	95.9	82.8	95.9	95.4	79.4	81.2
		SynthCI-20M	20	✓	70.6	77.4	64.4	81.4	96.7	84.7	96.6	96.1	82.8	83.4
		SynthCI-30M	30	✓	74.0	80.8	**66.1**	**82.5**	**97.2**	**86.2**	96.8	96.5	83.6	**84.9**

(a) Vision Tasks

				Image Retrieval				*Text Retrieval*				*0-shot*
Method	**Data**	**Samples** ($\times 10^6$)	Synth. data	MS Coco	Flickr 8K	Flickr 30K	Avg	MS Coco	Flickr 8K	Flickr 30K	Avg	Imagenet
CLIP	CC3M	3	✗	23.6	39.9	37.7	33.7	29.7	50.8	48.1	42.9	14.9
	CC12M	8.8	✗	43.8	66.2	66.8	58.9	57.4	80.3	77.3	71.7	**33.6**
SynthCLIP	SynthCI-3M	3	✓	21.5	39.1	41.1	33.9	28.9	53.7	55.4	46.0	9.5
	SynthCI-8.8M	8.8	✓	34.9	58.0	61.5	51.5	48.6	76.0	79.3	68.0	18.5
	SynthCI-10M	10	✓	36.7	58.0	64.0	52.9	50.0	75.1	81.8	69.0	20.9
	SynthCI-20M	20	✓	42.5	65.4	69.2	59.0	57.8	81.7	87.5	75.7	28.0
	SynthCI-30M	30	✓	**44.0**	**68.3**	**72.9**	**61.7**	**58.0**	**84.4**	**88.8**	**77.1**	30.5

(b) Vision-Language Tasks

SynthCLIP setup		*Baseline CLIP*	
Data	**Samples** ($\times 10^6$)	CC3M	CC12M
SynthCI-3M	3	-5.60%	-36.0%
SynthCI-8.8M	8.8	+31.3%	-15.0%
SynthCI-10M	10	+36.4%	-12.3%
SynthCI-20M	20	+53.9%	-3.10%
SynthCI-30M	30	**+60.1%**	**+0.20%**

(c) Δ_{MTL} evaluation

Table 1. **Benchmark.** We compare against CLIP models trained on real datasets (CC3M and CC12M). We train SynthCLIP on our synthetic datasets, SynthCI, at various scales. We observe a consistent improvement in performance in both vision ((a)) and vision-language ((b)) tasks, as the scale of SynthCI dataset increase. This demonstrates the scalability advantage of SynthCLIP. In (c) we aggregate multi-task performance with Δ_{MTL} across all trained networks.

against which we achieve $\Delta_{\text{MTL}} =$ **+0.20%**. This is suprising, since it shows that with multiple synthetic examples it is possible to fill the distribution gap between real and synthetic data, paving new ways for fully synthetic trainings. The generation script ran for a total of 6.45 days. We also report a significant increase with respect to CLIP trained on CC3M ($\Delta_{\text{MTL}} =$ **+60.1%**). From a single task perspective, we outperform CLIP trained on CC12 on image and text retrieval (**+2.8%** and **+5.4%**, respectively), while performing competitively with linear probing (**-1.7%**) and few-shot (**+0.00%**). While we still underperform in zero-shot evaluation (**-3.1%**), we attribute this also to additional bias effects that we study in Section 4.3.

Scaling trends To ease understanding to which extent scaling training data influences each task, we plot percentage improvements for each task in Figure 4, assuming as reference the performance achieved with SynthCLIP trained on SynthCI-3M. As visible from the plot, vision-language tasks (green, red, purple curves) tend to achieve more significant performance increase with respect to vision (blue, orange). We attribute this to the good quality of our captions, that thanks to our two-step generation pipeline are always fairly aligned with the corresponding image. This further helps to mitigate the distribution shift at scale.

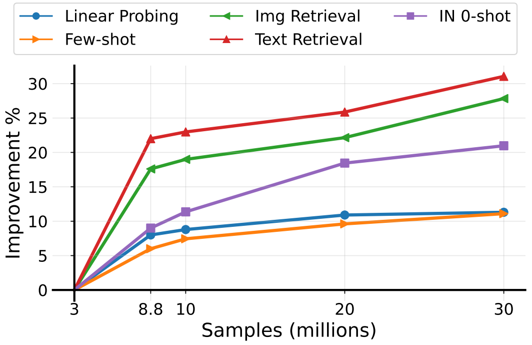
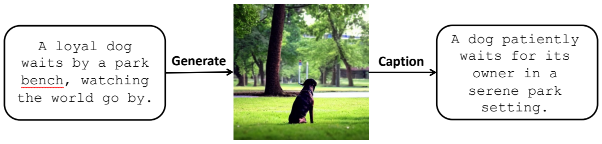


Figure 4. **Performance improvement for different SynthCI scales.** We show the improvements for all metrics with respect to SynthCLIP trained on SynthCI-3M. Vision-language tasks exhibit better absolute improvements and less saturation with respect to vision ones.

Method	Synth. Images	Synth. Captions	*Linear Probing*	*Few-shot*	*Img retrieval*	*Text retrieval*	*IN 0-shot*
CLIP	✗	✗	63.6	74.2	33.7	42.9	14.9
CLIP + Text-to-image	✓	✗	65.1	74.2	41.2	51.7	**15.4**
CLIP + Captioning	✗	✓	**70.1**	**78.1**	**46.0**	**62.4**	12.4
SynthCLIP	✓	✓	63.7	73.8	33.9	46.0	9.5
SynthCLIP + Captioning	✓	✓	66.5	74.3	43.5	57.1	8.5

(a) Quantitative evaluation


A loyal dog waits by a park bench, watching the world go by.

A dog patiently waits for its owner in a serene park setting.

(b) Captioning examples on SynthCI data

Figure 5. **Which synthetic data modality matters more?** We assess which synthetic modality impacts performance more by experimenting with combinations of real/synthetic captions/images ((a)). Real captions refers to taking the original captions from CC3M. Synthetic captions refers to either captions generated by LLaVA [36] ("Captioning") or an LLM (SynthCLIP). Synthetic images refers to generated images from Stable Diffusion. The source of prompts can be either real (CLIP + Text-to-image) or synthetic (SynthCLIP). Captioning with LLAVA improves performance even in SynthCLIP, due to corrections ((b)), where elements in the prompt missing in generated images are underlined in red.

4.3. Analysis

In this section, we conduct some analysis and ablation studies to examine key aspects of SynthCLIP. Specifically, we analyze the importance of textual and visual data modalities, ablate pipeline components (data filtering technique and LLM used for captions generation) , and quantify the effects of the concept bank size. For all experiments, we train on 3 million samples, *i.e.*, a similar scale to CC3M, due to the high computational cost of the larger experiments.

Do synthetic captions or synthetic images matter more? SynthCLIP uses entirely synthetic text-image pairs. A key question arises: which has a greater impact on the model's performance in downstream tasks – synthetic images or synthetic captions? In Table 5a, we compare the standard CLIP model trained on CC3M, SynthCLIP, and two hybrid CLIP variants. One hybrid uses real captions with synthetic images (CLIP + Text-to-Image), generated using Stable Diffusion v1.5, while the other pairs real images with synthetic captions (CLIP + Captioning), created with the LLaVA [36] model. Note that these hybrids, which require one real modality, are less scalable than SynthCLIP.

Our comparison reveals that CLIP + Captioning significantly outperforms standard CLIP in several benchmarks, indicating the effectiveness of synthetic captions in CLIP training. For instance, this approach improves linear probing by 6.5% and text retrieval by 19.5%, though it slightly decreases zero-shot performance by 2.5%. On the other hand, CLIP + Text-to-Image shows less marked improvements and no gains in few-shot performance. This suggests that keeping images real and recaptioning them is more advantageous than generating images for real captions, possibly due to domain shifts and content generation mismatches in synthetic images as noted in Gani et al. [13], Wu et al. [67].

Following this observation, we introduce SynthCLIP+Captioning as an extra baseline. Given that text-to-image models could miss details in text prompts, recaptioning post-image generation can be beneficial. This is evident in Figure 5b, where recaptioning corrects alignment issues from the image generation process (*e.g.* the missing `bench` in the generated image). Comparing SynthCLIP and SynthCLIP+Captioning in Table 5a (rows 4 and 5) shows significant gains with captioning, such as a 9.6% improvement in image retrieval. These results open future directions for combining SynthCLIP with caption enhancement techniques like VeCLIP [29] and CapsFusion [72] for better performance.

Data Filtering Ablation In creating our SynthCI-X datasets in Section 4.2, we utilized balanced sampling to select a desired number of captions from a larger set of generated ones. In this section we want to assess how different data sampling strategies affect SynthCLIP's performance. We focus on the impact of substituting balanced sampling with a more straightforward *random sampling* approach. For this, we randomly choose a subset of 3×10^6 captions from $\mathcal{T}$. The corresponding images for these randomly selected captions are generated using Stable Diffusion v1.5, following the same procedure presented in Section 4.1.

We then proceed to train SynthCLIP on this newly

Method	*Lin. Prob.*	*Few-shot*	*Img Ret.*	*Text Ret.*	*IN 0-shot*
SynthCLIP	**63.7**	**73.8**	**33.9**	**46.0**	**9.5**
↪ *w/ rand. sampling*	61.5 (-2.2)	72.0 (-1.8)	31.2 (-2.7)	43.3 (-2.7)	9.4 (-0.1)

(a) Balanced Sampling vs Random Sampling

LLM	*Lin. Prob.*	*Few-shot*	*Img Ret.*	*Text Ret.*	*IN 0-shot*
Mistral 7B	**63.7**	**73.8**	**33.9**	**46.0**	**9.5**
Vicuna 33B	61.4 (-2.3)	69.4 (-4.4)	26.1 (-7.8)	36.5 (-9.5)	8.2 (-1.3)

(b) Results with a different LLM for captions

Table 2. **Ablating Captions Generation Components.** Table ((a)) compares balanced and random sampling methods, revealing balanced sampling's superiority in enhancing task performance, while random sampling notably reduces effectiveness. Table ((b)) contrasts language models Mistral-7B and Vicuna-33B for data generation, showing Mistral-7B's consistent advantage across various tasks.

Concepts	N_c ($\times 10^3$)	*Lin. Prob.*	*Few-shot*	*Img Ret.*	*Text Ret.*	*IN 0-shot*
$\mathcal{C}$	500	63.7	73.8	33.9	46.0	9.5
$\mathcal{C}_{\text{CC3M}}$	40	**65.4** (+1.7)	**74.8** (+1.0)	**37.1** (+3.2)	**49.9** (+3.9)	**12.6** (+3.1)
$\mathcal{C}_{\text{rand}}$	40	63.1 (-0.6)	72.9 (-0.9)	31.8 (-2.1)	44.8 (-1.2)	9.2 (-0.3)

Table 3. **Effect of Concept Bank Size.** We compare SynthCLIP model performance using different concept bank sizes: the full 500×10^3 concepts ($\mathcal{C}$), a 40×10^3 subset from CC3M ($\mathcal{C}_{\text{CC3M}}$), and a randomly selected 40×10^3 subset ($\mathcal{C}_{\text{rand}}$), with each trained on 3 million samples. Results show that models trained on CC3M-specific concepts outperform those using the full concept list or a random selection, when a limited number of samples is used. This justifies scaling $\mathcal{C}$ and suggests a distribution bias in CC3M.

formed dataset. The results, presented in Table 2a, indicate a noticeable decline in performance across various tasks with random sampling, especially in retrieval tasks. Here, we observe a drop of 2.7% in both image and text retrieval compared to balanced sampling. These results underline the critical role of balanced the concept distribution for SynthCLIP.

Evaluating Different Language Models for Caption Generation In Table 2b, we study the effect of changing the language model from Mistral V0.2 7B model to Vicuna 33B. We find that using Mistral V0.2 7B consistently achieves better performance when compared to Vicuna 33B. This might be attributed to Mistral's superior performance on instruction-following benchmarks such as AlpacaEval [34]. Indeed, we phrase caption generation as an instruction-following task as previously described in Section 3.1. This suggests that with increasingly performing models in instruction following, it will be possible to further improve performances of SynthCLIP training.

Concept Bank impact In this section, we explore how the concept bank size $\mathcal{C}$ and the type of concepts it contains affect the downstream performance of the model. For this, we create two distinct subsets of $\mathcal{C}$. The first subset, $\mathcal{C}_{\text{CC3M}}$, is derived by identifying the concepts that appear in CC3M captions, by performing substring matching with concepts included in $\mathcal{C}$. This results in 40×10^3 CC3M-related concepts. The second, $\mathcal{C}_{\text{rand}}$, is formed by randomly selecting the same number of concepts than in $\mathcal{C}_{\text{CC3M}}$ from $\mathcal{C}$.

We generate 3M images for each of $\mathcal{C}_{\text{CC3M}}$ and $\mathcal{C}_{\text{rand}}$ and train SynthCLIP on the generated datasets. The results are summarized in Table 3. Interestingly, we noticed that focusing on CC3M-specific concepts ($\mathcal{C}_{\text{CC3M}}$) enhances performance compared to training with the full $\mathcal{C}$. For example, using $\mathcal{C}_{\text{CC3M}}$ yields a 3.9% improvement in text retrieval and 1.6% in linear probing. We hypothesize that this might be because $\mathcal{C}_{\text{CC3M}}$'s concepts are more aligned with concepts appearing in the downstream tasks, hence indicating a potential distribution bias in CC3M towards concepts prevalent in downstream task images. In contrast, using $\mathcal{C}_{\text{rand}}$ leads to lower performance in all tasks compared to the full $\mathcal{C}$. For example, we observe a 1.2% decrease in text retrieval and 0.8% in linear probing, likely because $\mathcal{C}_{\text{rand}}$'s concepts are less relevant to the downstream tasks. Hence, when specific insights about downstream tasks are unavailable, it is preferable to train on the widest possible range of concepts.

5. Conclusion

SynthCLIP represents a new approach to train CLIP models, addressing the limitations of web-sourced data through the generation of synthetic text-image pairs. Our experiments show SynthCLIP's scalability and capability to match the performance of models trained on real data. This paves new ways for entirely synthetic training at scale, which may further extend the capabilities of CLIP. The release of the SynCI-30M dataset, a substantial collection of synthetic image-caption pairs, along with the generation code, aims to allow further exploration of this direction.

References

[1] Shekoofeh Azizi, Simon Kornblith, Chitwan Saharia, Mohammad Norouzi, and David J. Fleet. Synthetic data from diffusion models improves imagenet classification. *TMLR*, 2023. 2

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *ECCV*, 2014. 5

[3] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. In *CVPR*, 2021. 1

[4] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 5

[5] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020. 1, 2

[6] Yuhua Chen, Wen Li, Xiaoran Chen, and Luc Van Gool. Learning semantic segmentation from synthetic data: A geometrically guided input-output adaptation approach. In *CVPR*, 2019. 2

[7] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 2017. 12

[8] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 5

[9] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *ICLR*, 2021. 5

[10] Lijie Fan, Kaifeng Chen, Dilip Krishnan, Dina Katabi, Phillip Isola, and Yonglong Tian. Scaling laws of synthetic images for model training... for now. *arXiv preprint arXiv:2312.04567*, 2023. 5

[11] Lijie Fan, Dilip Krishnan, Phillip Isola, Dina Katabi, and Yonglong Tian. Improving clip training with language rewrites. In *NeurIPS*, 2023. 1, 2

[12] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. *CVPR Workshop*, 2004. 5

[13] Hanan Gani, Shariq Farooq Bhat, Muzammal Naseer, Salman Khan, and Peter Wonka. Llm blueprint: Enabling text-to-image generation with complex and detailed prompts. *arXiv preprint arXiv:2310.10640*, 2023. 7

[14] Andreas Geiger, Philip Lenz, Christoph Stiller, and Raquel Urtasun. Vision meets robotics: The kitti dataset. *IJRR*, 2013. 12

[15] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *ICLR*, 2018. 2

[16] Jindong Gu, Zhen Han, Shuo Chen, Ahmad Beirami, Bailan He, Gengyuan Zhang, Ruotong Liao, Yao Qin, Volker Tresp, and Philip Torr. A systematic survey of prompt engineering on vision-language foundation models. *arXiv preprint arXiv:2307.12980*, 2023. 3

[17] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022. 1, 2

[18] Ruifei He, Shuyang Sun, Xin Yu, Chuhui Xue, Wenqing Zhang, Philip Torr, Song Bai, and XIAOJUAN QI. Is synthetic data from generative models ready for image recognition? In *ICLR*, 2023. 2

[19] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *JSTARS*, 2019. 12

[20] Micah Hodosh, Peter Young, and Julia Hockenmaier. Framing image description as a ranking task: Data, models and evaluation metrics. *JAIR*, 2013. 5

[21] Ali Jahanian, Xavier Puig, Yonglong Tian, and Phillip Isola. Generative models as a data source for multiview representation learning. In *ICLR*, 2022. 2

[22] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 5

[23] Matthew Johnson-Roberson, Charles Barto, Rounak Mehta, Sharath Nittur Sridhar, Karl Rosaen, and Ram Vasudevan. Driving in the matrix: Can virtual worlds replace human-generated annotations for real world tasks? In *ICRA*, 2017. 2

[24] Wooyoung Kang, Jonghwan Mun, Sungjun Lee, and Byungseok Roh. Noise-aware learning from web-crawled image-text data for image captioning. In *ICCV*, 2023. 1

[25] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[26] Alex Krizhevsky, Geoffrey Hinton, et al. Learning multiple layers of features from tiny images. 2009. 5

[27] Gihyun Kwon and Jong Chul Ye. Clipstyler: Image style transfer with a single text condition. In *CVPR*, 2022. 1

[28] Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient memory management for large language model serving with pagedattention. In *SOSP*, 2023. 13

[29] Zhengfeng Lai, Haotian Zhang, Wentao Wu, Haoping Bai, Aleksei Timofeev, Xianzhi Du, Zhe Gan, Jiulong Shan, Chen-Nee Chuah, Yinfei Yang, et al. From scarcity to efficiency: Improving clip training via visual-enriched captions. *arXiv preprint arXiv:2310.07699*, 2023. 1, 2, 7

[30] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 1998. 12

[31] Alexander Cong Li, Ellis Langham Brown, Alexei A Efros, and Deepak Pathak. Internet explorer: Targeted representation learning on the open web. In *ICML*, 2023. 1

[32] Guohao Li, Hasan Abed Al Kader Hammoud, Hani Itani, Dmitrii Khizbullin, and Bernard Ghanem. CAMEL: Communicative agents for ”mind” exploration of large language model society. In *NeurIPS*, 2023. 2

[33] Junnan Li, Dongxu Li, Caiming Xiong, and Steven Hoi. Blip: Bootstrapping language-image pre-training for unified vision-language understanding and generation. In *ICML*, 2022. 2

[34] Xuechen Li, Tianyi Zhang, Yann Dubois, Rohan Taori, Ishaan Gulrajani, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. Alpacaeval: An automatic evaluator of instruction-following models. https://github.com/tatsu-lab/alpaca_eval, 2023. 8

[35] 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 *ECCV*, 2014. 5

[36] Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. Visual instruction tuning. In *NeurIPS*, 2023. 2, 7

[37] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *arXiv preprint arXiv:2303.05499*, 2023. 1

[38] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013. 5

[39] Norman Mu, Alexander Kirillov, David Wagner, and Saining Xie. Slip: Self-supervision meets language-image pre-training. In *ECCV*, 2022. 2, 5

[40] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *ICVGIP*, 2008. 5

[41] Mehdi Noroozi and Paolo Favaro. Unsupervised learning of visual representations by solving jigsaw puzzles. In *ECCV*, 2016. 2

[42] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *CVPR*, 2012. 5

[43] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A Efros. Context encoders: Feature learning by inpainting. In *CVPR*, 2016. 2

[44] Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Dmytro Okhonko, Samuel Broscheit, Gautier Izacard, Patrick Lewis, Barlas Oğuz, Edouard Grave, Wen-tau Yih, et al. The web is your oyster-knowledge-intensive nlp against a very large web corpus. *arXiv preprint arXiv:2112.09924*, 2021. 1

[45] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 2, 4, 5

[46] 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. In *ICML*, 2021. 1, 5, 12

[47] Tianhe Ren, Shilong Liu, Ailing Zeng, Jing Lin, Kunchang Li, He Cao, Jiayu Chen, Xinyu Huang, Yukang Chen, Feng Yan, et al. Grounded sam: Assembling open-world models for diverse visual tasks. *arXiv preprint arXiv:2401.14159*, 2024. 1

[48] Stephan R Richter, Vibhav Vineet, Stefan Roth, and Vladlen Koltun. Playing for data: Ground truth from computer games. In *ECCV*, 2016. 2

[49] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 4, 5

[50] German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The synthia dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *CVPR*, 2016. 2

[51] Nick Rossenbach, Albert Zeyer, Ralf Schlüter, and Hermann Ney. Generating synthetic audio data for attention-based speech recognition systems. In *ICASSP*, 2020. 2

[52] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. ImageNet Large Scale Visual Recognition Challenge. *IJCV*, 2015. 5

[53] Mert Bulent Sariyildiz, Karteek Alahari, Diane Larlus, and Yannis Kalantidis. Fake it till you make it: Learning transferable representations from synthetic imagenet clones. In *CVPR*, 2023. 2

[54] Axel Sauer, Dominik Lorenz, Andreas Blattmann, and Robin Rombach. Adversarial diffusion distillation. *arXiv preprint arXiv:2311.17042*, 2023. 13

[55] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In *CVPR*, 2015. 1

[56] Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *NeurIPS*, 2022. 4

[57] Piyush Sharma, Nan Ding, Sebastian Goodman, and Radu Soricut. Conceptual captions: A cleaned, hypernymed, image alt-text dataset for automatic image captioning. In *ACL*, 2018. 5

[58] Tianhao Shen, Renren Jin, Yufei Huang, Chuang Liu, Weilong Dong, Zishan Guo, Xinwei Wu, Yan Liu, and Deyi Xiong. Large language model alignment: A survey. *arXiv preprint arXiv:2309.15025*, 2023. 5

[59] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. How good is my gan? In *ECCV*, 2018. 2

[60] Mohit Shridhar, Lucas Manuelli, and Dieter Fox. Cliport: What and where pathways for robotic manipulation. In *CoRL*, 2022. 1

[61] StabilityAI. Stable Diffusion Discord Server Rules, 2022. 5

[62] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and Christian Igel. The german traffic sign recognition benchmark: a multi-class classification competition. In *IJCNN*, 2011. 12

[63] Yonglong Tian, Lijie Fan, Kaifeng Chen, Dina Katabi, Dilip Krishnan, and Phillip Isola. Learning vision from models rivals learning vision from data. *arXiv preprint arXiv:2312.17742*, 2023. 13

[64] Yonglong Tian, Lijie Fan, Phillip Isola, Huiwen Chang, and Dilip Krishnan. Stablerep: Synthetic images from text-to-image models make strong visual representation learners. In *NeurIPS*, 2023. 1, 2, 5

[65] Simon Vandenhende, Stamatios Georgoulis, Wouter Van Gansbeke, Marc Proesmans, Dengxin Dai, and Luc Van Gool. Multi-task learning for dense prediction tasks: A survey. *TPAMI*, 2021. 5

[66] Gül Varol, Javier Romero, Xavier Martin, Naureen Mahmood, Michael J. Black, Ivan Laptev, and Cordelia Schmid. Learning from synthetic humans. In *CVPR*, 2017. 2

[67] Weijia Wu, Zhuang Li, Yefei He, Mike Zheng Shou, Chunhua Shen, Lele Cheng, Yan Li, Tingting Gao, Di Zhang, and Zhongyuan Wang. Paragraph-to-image generation with information-enriched diffusion model. *arXiv preprint arXiv:2311.14284*, 2023. 7

[68] J. Xiao, J. Hays, K. A. Ehinger, A. Oliva, and A. Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *CVPR*, 2010. 5

[69] Hu Xu, Saining Xie, Xiaoqing Ellen Tan, Po-Yao Huang, Russell Howes, Vasu Sharma, Shang-Wen Li, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Demystifying clip data. *arXiv preprint arXiv:2309.16671*, 2023. 3, 4

[70] Yiben Yang, Chaitanya Malaviya, Jared Fernandez, Swabha Swayamdipta, Ronan Le Bras, Ji-Ping Wang, Chandra Bhagavatula, Yejin Choi, and Doug Downey. Generative data augmentation for commonsense reasoning. In *EMNLP*, 2020. 2

[71] Peter Young, Alice Lai, Micah Hodosh, and Julia Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *TACL*, 2014. 5

[72] Qiying Yu, Quan Sun, Xiaosong Zhang, Yufeng Cui, Fan Zhang, Yue Cao, Xinlong Wang, and Jingjing Liu. Capsfusion: Rethinking image-text data at scale. *arXiv preprint arXiv:2310.20550*, 2023. 2, 7

[73] Zhuoran Yu, Chenchen Zhu, Sean Culatana, Raghuraman Krishnamoorthi, Fanyi Xiao, and Yong Jae Lee. Diversify, don't fine-tune: Scaling up visual recognition training with synthetic images. *arXiv preprint arXiv:2312.02253*, 2023. 2

[74] Jianhao Yuan, Jie Zhang, Shuyang Sun, Philip Torr, and Bo Zhao. Real-fake: Effective training data synthesis through distribution matching. In *ICLR*, 2024. 2

[75] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. *ICCV*, 2023. 2

[76] Yongchao Zhou, Hshmat Sahak, and Jimmy Ba. Training on thin air: Improve image classification with generated data. *arXiv preprint arXiv:2305.15316*, 2023. 2, 5

In this supplementary material, we provide further details and evaluations of SynthCLIP. More specifically, in Section A we prevent an analysis on the number of concepts covered in real and synthetic datasets highlighting the gap between both. In Section B we evaluate the performance of training CLIP models on SynthCI and real datasets on OOD downstream tasks, showing small gaps between training on real and synthetic data. Section C shows the failed attempts to generate synthetic captions. Finally, in Section D we further discuss the importance of further studying end-to-end synthetic approaches.

A. Concept Appearance

In this section, we examine the presence of concepts from our extensive 500×10^3 concept bank within real text-image datasets like CC3M and CC12M, as well as our SynthCI synthetic datasets. Our method involves substring matching, where we identify and count the occurrences of each concept within the captions of these datasets. This count reveals how frequently different concepts appear, particularly those occurring more than a specified number of times (k).

Table 4 summarizes these findings. Notably, even the smallest SynthCI-3M dataset contains significantly more concepts than the larger real CC12M dataset, surpassing it by nearly 2.5 times in terms of concepts appearing at least once ($k = 1$). This trend of broader concept coverage in SynthCI datasets persists even when increasing the threshold to $k = 25$ or $k = 50$. An intriguing aspect is the average number of samples per concept. The last column of Table 4 shows the average frequency of concept occurrences, considering only those appearing at least 25 times. While CC3M and CC12M, with fewer overall concepts, exhibit a higher average of samples per concept, our SynthCI datasets generally show lower averages. However, SynthCI-30M shows the same average as real datasets, particularly CC12M. This similarity in samples per concept at 30M scale could be a key factor in SynthCI-30M matching the performance of CC12M.

B. OOD Evaluation

We need something better for this concept appearance table I think. Not priority but I'd like it to be more beautiful, it looks like random stats thrown there. Can we have a small plot maybe? this is not how you start a new paragraph. state first what are you after. state what is the objective. or at least what is the observation from before that prompted this experiment Although a vast enough $\mathcal{C}$ ensures variability of the generated content, SynthCI is composed mostly by captioned images representing single elements in context.not clear .. given an example to "single elements in context" While this is expected due to the prompt used, which specifically encourages single concept-centered scenes, one could argue that this may impact the generalization capabilities of SynthCLIP on out-of-distribution data. this needs clarification terribly writtenahah my fault

Dataset	Concept Appearance			Average Appearance
	$k = 1$	$k = 25$	$k = 50$	$k \geq 25$
CC3M	3.9×10^4	1.8×10^4	1.4×10^4	1.4×10^3
SynthCI-3M	3.0×10^5	3.6×10^4	2.3×10^4	1.0×10^3
CC12M	1.3×10^5	4.8×10^4	3.7×10^4	2.0×10^3
SynthCI-8.8	3.4×10^5	2.3×10^5	5.6×10^4	6.2×10^2
SynthCI-10M	3.4×10^5	2.3×10^5	8.5×10^4	7.0×10^2
SynthCI-20M	3.4×10^5	2.3×10^5	1.9×10^5	1.4×10^3
SynthCI-30M	3.5×10^5	2.4×10^5	1.9×10^5	2.0×10^3

Table 4. **Concept Appearance in Real vs. Synthetic Datasets.** This table compares the frequency of concept appearances in real datasets (CC3M, CC12M) and their synthetic counterparts. It shows the number of concepts that appear at least k times, along with the average appearances for concepts occurring at least 25 times.

We evaluate SynthCLIP on downstream tasks on datasets that do not encompass object-centric scenes. This includes satellite landmark classification (EUROSAT [19] and RESISC45 [7]), character (MNIST [30]) and sign (GTSRB [62]) recognition. Moreover, we compare on tasks departing from object classification, to show generalization capabilities to different tasks. These involve distance estimation in street scenarios (KITTI [14]), and geo-localization (Country211 [46]). As shown in Table 5, SynthCLIP exhibits a competitive performance with its CLIP baseline counterpart trained on real text-image pairs. This serves as a further confirmation that SynthCLIP learns to extract representations transferable to various applications from our fully synthetic SynthCI.

several problems with the above. (1) as usual you should discuss a few numbers. you are not here to laundtry list tables and let people read it. might as well screenshot youre excel sheets into the paper without any discussion. the reviewers want to be spoonfed into reading results (at least one example) otherwise they will take the easy route out 2. i have no idea from the text above why that section is important. you need to spend sometime on the observation and motivation discussing set C. and also why do you think these datasets, in particular, are actually out-of-distribution can we do min distance clustering, pcs, tsnet, to show that these datasets are sufficiently different 3. back again at the general message, what is the point here? synthetic data helps more on out of dist? or the gap to real data improves at a faster rate on out of dist? or is it to do with the choice of elements in the concept list?

Method	Data	Size	Synth. data	EUROSAT	GTSRB	MNIST	RESISC45	Country211	KITTI
CLIP	CC3M	3M	✗	94.9	64.7	97.9	85.5	12.3	71.9
SynthCLIP	SynthCI-3M	3M	✓	95.3	61.4	98.2	86.8	12.6	72.5
CLIP	CC12M	8.8M	✗	96.3	**73.7**	**98.8**	**90.3**	17.4	72.1
SynthCLIP	SynthCI-12M	8.8M	✓	96.2	66.8	98.7	88.9	15.4	73.5
SynthCLIP	SynthCI-30M	30M	✓	**96.8**	68.0	98.5	89.2	**17.9**	**74.3**

Table 5. **Generalization on out-of-distribution datasets.** We provide additional linear probing results on out-of-distribution tasks such as satellite image classification (EUROSAT, RESISC45), character (MNIST) and sign recognition (GTSRB), country classification (Country211), and distance estimation (KITTI). We achieve competitive results with baselines trained on real data. This attests to the transferability of features learned on synthetic data.i think we should add Delta metric everywhereHere it's only LP so it's not needed. Also we won't get good perf on this

C. Failed Attempts for Synthetic Captions Generation

In this section we showcase the failed attempts to generate synthetic captions:

Attempt 1 - Generate Captions without Any Conditioning In our first attempt, we tried to let LLM generate any topic it wants without any conditioning. This was done using the prompt shown in Figure 6. Unfortunately, the captions were overly descriptive and hard for the text-to-image model to generate images for and they were always focused on nature, resulting in low variability unsuitable for CLIP training. Examples of generated captions are:

- A sunlit garden: vibrant roses bloom against a brick wall, butterflies dance around, water droplets sparkle on leaves, soft focus, balanced composition.
- Sunset over tranquil lake: A solitary kayaker paddles through golden reflection, mountains in distance bathed in warm light. Focus on kayaker's determined face, balanced composition. Soft toned, impressionistic brushstrokes.
- A sunlit garden: vibrant roses bloom against a weathered brick wall, butterflies dance around ripe strawberries on a red table, children play nearby, laughter echoes softly. Warmth radiates from every detail.

Attempt 2 - Generate Captions Using a Topics Bank Instead of having a concept bank that we generate synthetic captions for, our first attempt was to try having a broader list of topics, *i.e.* a topic bank, used for conditional generation. Particularly, we used the topics shown in Figure 7 and then used the prompt shown in Figure 8 to generate the captions.

> You are an expert image descriptions generator. Your task is to write an image caption to describe a scene that can be used with text-to-image generation model such as DALL-E.
> Your description should vividly and descriptively detail the scene to guide the image generator in producing its visualizations of the caption depiction. Use simple words, and the rewrite has to be less than 15 words.
> Use modifiers such as lighting, focus, balance, composition, angle, reflections, textures, color palette, style, tone, effects, lens type, mood, artist or photographer name, and more.

Figure 6. **Attempt 1 - Captions Generation Prompt**

The observed issue is that for each topic the LLM had some kind of favourable instance. For example for ”Wild Animals”, most generated captions were about Leopards:

- In the desert, a leopard is dragging its kill.
- A leopard carries its prey through the arid desert landscape.
- The majestic snow leopard roams high within Himalayas mountain range territory.

This issue was not resolvable by adjusting the prompt or parameters of the LLM including the seed, temperature and top-p value. Interestingly, this signals that biases in concept-oriented generations in LLM are significant regardless the amount of data they are trained on. Since we were mostly interested in maximizing the variability of generated concepts, we opted for the concept-based generation pipeline presented in Section 3.1.

D. Further Discussions

In this section, we delve into the significance of using an end-to-end synthetic training methods for model training. Recent advancements in text-to-image and large language models have not only enhanced generation quality but also accelerated inference speeds [28, 54]. Our generation process currently takes approximately 6.5 days using a 48-A100-80GB GPU cluster, equivalent to 313 GPU days. However, with continual technological advancements, we anticipate a reduction in the time required for generation, leading to more efficient and scalable end-to-end approaches. A related recent concurrent work [63] utilizing a similar pipeline to ours, but focused on vision-only tasks, demonstrates that such methods can scale up to 600 million samples, requiring around 6260 GPU days on A100-80GB GPUs. Future research should include further experimentation with various language models, text-to-

Space, Celestial Bodies, Nature, Natural Landscapes, Plants, Trees, Flowers, Domestic Animals, Wild Animals, Gadgets and Electronics, Historical Landmarks and Monuments, Oceans, Marine Life, Underwater Scenery, Mountains, Geographical Features, Urban Landscapes, Cityscapes, Art, Sculptures, Visual Arts, Festivals ,Cultural Events, Celebrations, Vehicles and Transportation, Sports ,Recreational Activities, Architecture,Buildings, Fashion, Clothing, Accessories, Food, Cuisine, Culinary Arts, Weather and Atmospheric Phenomena, Astronomy ,Astrophysics, Musical Instruments, Performances, Traditional and Folk Crafts, Books, Literature, Written Works, Films, Movies, Theater, Dance and Performing Arts, Educational and Scientific Concepts, Health, Medicine, Wellness, Fantasy, Mythology, and Folklore, Video Games and Virtual Worlds, Historical Eras and Civilizations, Celebrities,Public Figures, Influencers, Insects, Microscopic Life, Small Creatures, Tools, Machinery, Industrial Equipment, Toys, Games, Children's Entertainment, Work Environments, Professions, and Occupations, Religious, Spiritual, and Mystical Symbols, Political, Social, and Environmental Movements, Everyday Household Objects and Utilities, Landscapes of Other Planets and Moons, Dinosaurs, Prehistoric Life, Paleo-Scenery, Kitchen Utensils, Cooking Tools, Home Accessories, Interior Decor, Office Furniture, Home Furniture, Gardens, Horticulture, Landscaping, Pets and Companion Animals, Aquatic and Water-based Activities, Educational and Learning Materials, Traditional Clothing, Ethnic Clothing, Body Art and Tattoos, Streets Roads, and Highways, Forests, Jungles, and Wilderness Areas, Planes, Boats, Photography, Robots, Futuristic

Figure 7. **Attempt 2 - Topics Bank**

image generators, and caption generation prompts to identify optimal configurations for improvement. Additionally, exploring how to best leverage the generated data remains a crucial area for future research.

Image captions are usually composed of four components: (1) Subject: This is the main focus of the sentence. (2) Action or State: This describes what the subject is doing or the state they are in. (3) Setting or Context: This provides additional information about where the action is taking place or the context surrounding the subject. (4) Additional Descriptors: These are adjectives or additional details that provide more depth or description to the subject or setting.
Your task is to write image captions as described above. You are provided with a ”Concept List” below which contains categories you can write captions for.
Concept List: {sampled_topics_string}
Rules:
(1) You are allowed a maximum of 15 words per image caption.
(2) Select at random a subject from the concept list or be creative and go beyond the list.
(3) Select a highly specific subject from the concept list for your caption. Avoid general categories; instead, choose a detailed and particular item, creature, or concept. The chosen subject should be a distinct and unique example within its broader category, reflecting your creativity and precision. This means you will provide names, breeds, locations, brands ... all of which ensure specificity.
(4) Write down the captions without specifying what category it belongs to.
Rule (3) is very important.
Please provide me with 10 captions following all the rules above.

Figure 8. **Attempt 2 - Captions Generation Prompt**