CLIPA-v2: Scaling CLIP Training with 81.1% Zero-shot ImageNet Accuracy within a $10,000 Budget

Anonymous Author(s)
Affiliation
Address
email

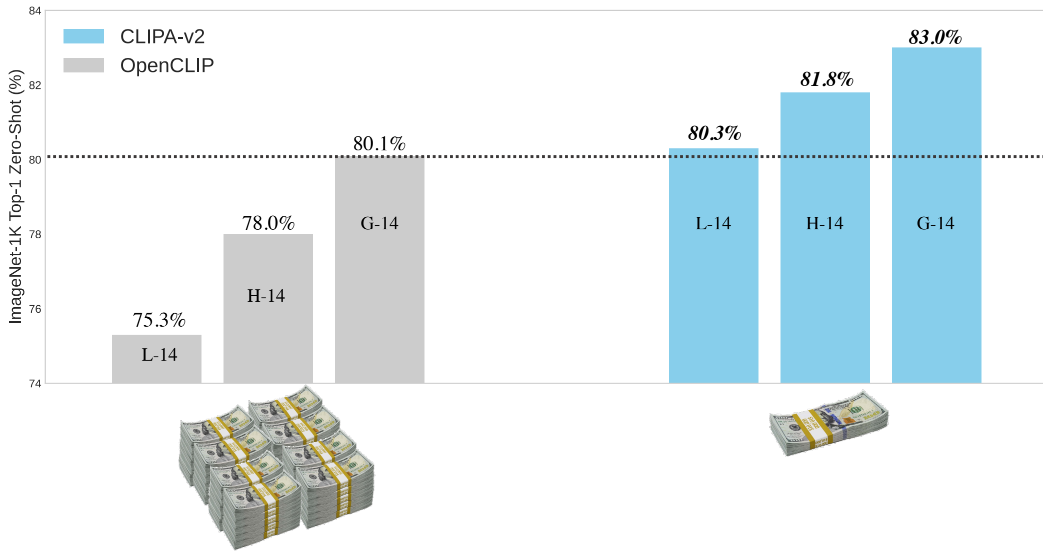

Figure 1: Compared to OpenCLIP [11], our CLIPA-v2 models achieve higher performance with much lower training cost.

Abstract

The recent work CLIPA [13] presents an *inverse scaling law* for CLIP training — whereby the larger the image/text encoders used, the shorter the sequence length of image/text tokens that can be applied in training. This finding enables us to train high-performance CLIP models with significantly reduced computations. Building upon this work, we hereby present CLIPA-v2 with two key contributions. Technically, we find this inverse scaling law is also applicable in the finetuning stage, enabling further reduction in computational needs. Empirically, we explore CLIPA at scale, extending the experiments up to the H/14 model with ~13B image-text pairs seen during training.

Our results are exciting — by only allocating a budget of $10,000, our CLIP model achieves an impressive zero-shot ImageNet accuracy of 81.1%, surpassing the prior best CLIP model (from OpenCLIP, 80.1%) by 1.0% and meanwhile reducing the computational cost by ~39×. Moreover, with an additional investment of $4,000, we can further elevate the zero-shot ImageNet accuracy to 81.8%. By upscaling a G/14 model, we've achieved an impressive state-of-the-art zero-shot ImageNet accuracy of 83.0%, relying solely on open-source data.

Submitted to R0-FoMo: Workshop on Robustness of Few-shot and Zero-shot Learning in Foundation Models at NeurIPS 2023. Do not distribute.

model	# image token	# text token	data source	# seen samples	total compute ($\times 1e11$)	IN-1K
CLIPA-L/16	36	8	LAION-400M	2.56B + 128M	0.5	69.3
CLIPA H/14	36	8	LAION-400M	2.56B + 128M	0.8	72.8
			LAION-2B	2.56B + 128M	0.8	74.1
			LAION-2B	**12.8B + 128M**	4	**77.9**

Table 1: **Scaling up CLIPA-v1 [13].** Specifically, we explore scaling from the aspects of data, model, and schedule. We pretrain the H/14 model with 36 image tokens (84×84) and 8 text tokens; for finetuning, we use 256 (224×224) image tokens and 32 text tokens, following [13].

1 Introduction

CLIP [18] has emerged as the pioneering foundation model that bridges the gap between text and images, ushering computer vision research into the "post-ImageNet" era [11, 14, 29, 1, 19, 21, 23, 27, 4]. However, the demanding computational requirements of CLIP hinder its widespread exploration. The recent work CLIPA [13] offers a computationally efficient solution — with the introduction of an *inverse scaling law* for CLIP training, it reveals that larger models can be trained with fewer input tokens. Building upon this observation, CLIPA demonstrates its efficacy in scenarios with limited computational resources, leading to a substantial reduction in the training cost of CLIP.

This report provides a follow-up on CLIPA. Firstly, we validate that the inverse scaling law is also applicable when finetuning models with input tokens at full resolution. This further reduces the training cost of CLIPA. Secondly, we investigate the performance of CLIPA at scale across various aspects, including model size (up to H/14), data (up to DataComp-1B [7] and LAION-2B [23] datasets), and training schedule (up to ~13B samples seen).

With these two contributions, we can train CLIP models with strong zero-shot performance on ImageNet [5], meanwhile significantly reducing training costs. For instance, we can train a H/14 model with 81.1% accuracy within a \$10,000 budget. We stress that, compared to the best publicly available CLIP model from OpenCLIP [11], ours is both better (**+1.0%**) and faster (by ~**39**$\times$). Moreover, we can further boost this accuracy to 81.8%, with an additional \$4,000 investment. These results are exciting as no prior work has thus far reached a similar performance within this small budget limitation. By open-sourcing our training code and models, we hope to contribute to the broader advancement and adoption of advanced CLIP models.

masking ratio	random	block	grid
25%	**78.2**	78.0	77.9
50%	**77.7**	77.6	77.6
75%	**76.2**	74.3	**76.2**

Table 2: Comparison of different masking strategy. The results are obtained on on the LAION-2B dataset with H/14 model.

case	masking ratio	resolution	# seen samples	training FLOPs	IN-1K
CLIPA-v1	0%	224^2	128M	177.0G	77.9
(1)	30%	224^2	128M	135.9G	78.0
(2)	30%	224^2	512M	135.9G	78.6
(3)	30%	224^2	640M	135.9G	78.5
(4)	40%	336^2	$640M$	237.8G	78.9
(5)	30%+40%	$224^2 + 336^2$	512M+128M	156.3G	**79.1**

Table 3: **Ablation of CLIPA-v2.** In case (5), we use 224×224 input with a masking ratio of 30% for the first 512M samples, and 336×336 input with a masking ratio of 40% for the rest 128M samples.

2 Background

CLIP has been a prominent foundation model due to its exceptional zero-shot capability and remarkable versatility [18, 12]. The tremendous success of CLIP can be attributed to the extensive scale of both the data [18, 22, 12, 3, 29, 30] and the model [28, 16, 24] it is built upon. Nevertheless, it also poses a significant cost barrier to researchers who wish to train a strong CLIP model. To reduce the computational burden, the recent work by Li et al. [13] presents an inverse scaling law, which reveals that larger models can effectively utilize fewer input tokens for training without severe performance drop, therefore enabling highly efficient CLIP training. As a byproduct of this discovery, the CLIPA models are introduced, which attain a zero-shot top-1 ImageNet accuracy of 69.3% and can be trained on an 8 A100-GPU machine in just 4 days.

Our work is built upon CLIPA [13], but focuses on furthering its efficiency and scaling it up.

	Models	Data Source	# seen samples@input size	GPU hours[1]	Est. cost [2]	zero-shot classification						zero-shot retrieval			
						IN-1K	IN-V2	IN-A	IN-R	ObjectNet	IN-SK	COCO		Flickr30k	
												image	text	image	text
OpenCLIP	H/14	LAION-2B	32.0B@224^2	216,712	\$247,864	78.0	70.8	59.2	89.3	69.7	66.6	49.5	66.0	77.8	90.8
CLIPA-v2	H/14	LAION-2B	12.8B@84^2 + 512M@224^2 + 128M@336^2	8,640	\$13,613	79.1	72.3	71.7	92.7	69.9	70.0	50.2	67.5	78.2	92.3
OpenCLIP	L/14	DataComp-1B	12.8B@224^2	41,472	\$47,434	79.2	72.1	69.6	90.8	74.3	68.0	45.7	63.3	73.4	89.5
OpenCLIP	G/14*	LAION-2B	32.0B@224^2 + 6.7B@224^2	232,448	\$366,105	80.1	73.6	69.4	92.2	73.0	68.9	**51.4**	67.3	**79.6**	**92.9**
CLIPA-v2	H/14	DataComp-1B	12.8B@70^2 + 512M@224^2	5,920	\$9,324	81.1	74.7	76.2	93.7	72.7	72.4	49.1	67.1	76.1	92.4
CLIPA-v2	L/14	DataComp-1B	12.8B@84^2 + 512M@224^2	4,008	\$6,318	79.7	72.8	73.2	92.1	71.1	69.3	46.3	64.1	73.0	89.1
			+128M@336^2	+512	+\$806	80.3	73.5	77.7	93.3	73.1	70.9	47.2	65.5	74.6	90.5
	H/14		12.8B@84^2 + 512M@224^2	7,776	\$12,247	81.5	75.0	76.9	94.3	74.1	72.7	49.1	67.0	75.7	90.6
			+128M@336^2	+864	+\$1,366	81.8	75.6	82.7	94.4	77.4	72.8	49.2	67.2	76.3	90.3
	G/14		12.8B@84^2 + 512M@224^2	21,998†	\$34,646†	82.7	76.9	81.7	95.1	77.1	74.3	50.0	67.9	77.7	91.8
			+128M@336^2	+1,744†	+\$4,410†	**83.0**	**77.3**	**85.9**	**95.4**	**79.7**	**74.5**	50.4	**67.8**	78.2	92.1

Table 4: **Comparison with OpenCLIP [11].** Our CLIPA-v2's GPU hour is estimated using an 8-A100 80GB GPU machine on Google Cloud, while the OpenCLIP's GPU hour is calculated based on their report[1]. The corresponding training cost is estimated based on 80GB A100's cloud pricing[2]. * denotes this model is trained with FLIP at a masking ratio of 50%. † denotes gradient accumulation is adopted in compute and cost estimation to accommodate large models given A100 GPUs.

3 Experiments

Our experiments contain three parts. Firstly, we check the applicability of *inverse scaling law* during the finetuning stage with full-resolution tokens. Next, we scale up CLIPA in terms of data, model, and schedule. Lastly, we compare with other advanced CLIP models in terms of performance and computation cost. Our pretraining setup strictly follows CLIPA [13]. We report the corresponding zero-shot top-1 accuracy on ImageNet [5].

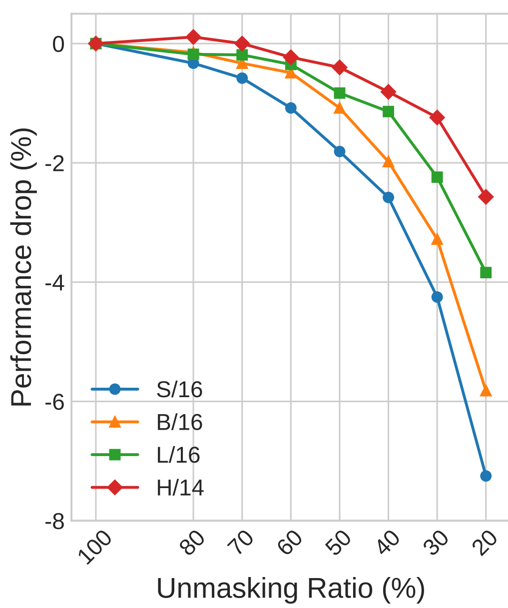

Figure 2: **The *inverse scaling law* on finetuning.** All models are finetuned with 128M samples, where we employ random masking for token reduction.

***Inverse scaling law* in the finetuning stage.** Following [13], we choose four different scales of models: S/16, B/16, L/16, and H/14, and train them on LAION-400M dataset. Random masking [14, 8] is used as the image token reduction strategy. As shown in Figure 2, larger models consistently exhibit a lower performance drop compared to smaller models when finetuning with the same number of input tokens. For instance, retaining 50% of the input tokens merely results in a performance drop of 0.4% for the H/14 model, compared to much higher drops of 0.8% for L/16, 1.1% for B/16, and 1.8% for S/16.

These results confirm the existence of the inverse scaling law in the finetuning stage, which enables us to reduce the required computations for CLIP training further.

Scaling up CLIPA [13]. We next investigate the scaling behavior beyond the largest case studied in CLIPA. Specifically, our scaling efforts cover three aspects: model, data, and training schedule. The results are reported in Table 1.

First, we can observe that scaling the model size from L/14 to H/14 boosts the performance from 69.3% to 72.8%. Furthermore, we note switching the training dataset from LAION-400M [23] to LAION-2B [22] yields another 1.3% improvement, suggesting the importance of data diversity. Lastly, by increasing the training schedule by a factor of 5, resulting in a total of ~13B seen samples, we achieve an impressive performance of 77.9%. We stress that this scaled version of CLIPA H/14 model readily outperforms its counterpart in FLIP [14] by 0.3% while requiring only $1/3$ of the training budget.

These results confirm the efficiency and effectiveness of training CLIPA at scale. Next, we set this CLIPA H/14 with 77.9% performance as our baseline for further ablation in the finetuning stage.

Ablation. In addition to random masking, we hereby investigate how grid masking and block masking affect finetuning performance. The results are reported in Table 2. Interestingly, compared to finetuning input tokens at the full resolution, we observe that 25% masked random finetuning and block finetuning all lead to a slight performance improvement. With a larger masking ratio, all these masking strategies will lead to worse performance than full-resolution fine-tuning; but overall, random masking consistently yields stronger performance than the other two masking strategies.

We next ablate different finetuning setups and summarize the results in Table 3. We choose 30% masked random finetuning as the default strategy, as it leads to a slight performance improvement (+0.1%) and enables a $1.3\times$ speedup of the finetuning process. Furthermore, adopting a $4\times$ finetuning schedule results in an additional improvement of 0.6%. However, further increasing the finetuning schedule does not lead to any substantial performance gains.

Following [11], we also investigate progressively finetuning with large image resolutions. Initially, for the first 512 million samples, we finetune the model using a 224×224 input size with a masking ratio of 30%; subsequently, for the remaining 128 million samples, we adopt a larger 336×336 input size with a masking ratio of 40% and a smaller learning rate. As shown in the last row of Table 3, *i.e.*, case (5), progressive finetuning results in a slight performance improvement of 0.2% compared to direct finetuning with a 336×336 input size and meanwhile achieving a notable $1.5\times$ speedup of the finetuning process.

Comparison with OpenCLIP [11]. We summarize the results in Table 4. Firstly, when trained on the LAION-2B dataset, our CLIPA-v2 H/14 model outperforms OpenCLIP's version by 1.1% (79.1% *vs* 78.0%) and meanwhile significantly reducing the training cost by $\sim 18\times$. Furthermore, when upgrading to the DataComp-1B dataset, our CLIPA-v2 H/14 (pretrained on images at 70×70) achieves an impressive zero-shot ImageNet accuracy of **81.1%**, while keeping the training cost within $10,000. Notably, this 81.1% accuracy is 1.0% higher than the prior best CLIP model, which is OpenCLIP's G/14 model with a zero-shot ImageNet accuracy of 80.1%.

With an additional investment of $4000, we can further enhance CLIPA-v2's training by 1) pretraining with a larger resolution (the image size from 70 to 84) and 2) applying the progressive finetuning with a larger image resolution of 336. These enhancements lead to an additional 0.7% improvement, resulting in the *best-performing CLIP model to date with an 81.8% zero-shot ImageNet accuracy*.

We also validate the superiority of CLIPA-v2 models on zero-shot robustness. For example, our 81.8% H/14 model consistently yields much stronger performance than OpenCLIP's 80.1% G/14 model on IN-V2 [20] (75.6% *vs* 73.6%), IN-A [10] (82.7% *vs* 69.4%), IN-R [9] (94.4% *vs* 92.2%), ObjectNet [2] (77.4% *vs* 73.0%), and IN-SK [26] (72.8% *vs* 68.9%). However, we note that, when evaluating zero-shot retrieval performance on COCO [15] and Flickr30k [17], OpenCLIP's 80.1% G/14 model still performs better We conjecture this performance advantage should be attributed to the difference in training datasets, as Table 4's results empirically suggest models trained with LAION-2B are better at retrieval tasks than models trained with DataComp-1B. Nonetheless, when also scaling to G/14, our CLIPA-v2 model achieves an unprecedented ImageNet zero-shot top-1 accuracy of **83.0%**, surpassing the previous best record of 82.0% made by EVA-02-CLIP-E/14+ [25, 6], with only **1/2** number of parameters.

References

[1] 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. In *NeurIPS*, 2022.

[1] We measure OpenCLIP [11]'s training time based on `https://laion.ai/blog/large-openclip/` and `https://laion.ai/blog/giant-openclip/`.

[2] We estimate the total training cost based on `https://cloud.google.com/compute/gpus-pricing`, which is $1.575 per GPU hour, and `https://lambdalabs.com/service/gpu-cloud/pricing`, which is $1.5 per GPU hour.

[2] Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. Objectnet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *NeurIPS*, 2019.

[3] 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.

[4] Yuchen Cui, Scott Niekum, Abhinav Gupta, Vikash Kumar, and Aravind Rajeswaran. Can foundation models perform zero-shot task specification for robot manipulation? *arXiv preprint arXiv:2204.11134*, 2022.

[5] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *CVPR*, 2009.

[6] Yuxin Fang, Quan Sun, Xinggang Wang, Tiejun Huang, Xinlong Wang, and Yue Cao. Eva-02: A visual representation for neon genesis. *arXiv preprint arXiv:2303.11331*, 2023.

[7] Samir Yitzhak Gadre, Gabriel Ilharco, Alex Fang, Jonathan Hayase, Georgios Smyrnis, Thao Nguyen, Ryan Marten, Mitchell Wortsman, Dhruba Ghosh, Jieyu Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023.

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

[9] Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, Dawn Song, Jacob Steinhardt, and Justin Gilmer. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *ICCV*, 2021.

[10] Dan Hendrycks, Kevin Zhao, Steven Basart, Jacob Steinhardt, and Dawn Song. Natural adversarial examples. In *CVPR*, 2021.

[11] Gabriel Ilharco, Mitchell Wortsman, Ross Wightman, Cade Gordon, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021.

[12] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.

[13] Xianhang Li, Zeyu Wang, and Cihang Xie. An inverse scaling law for clip training. *arXiv preprint arXiv:2305.07017*, 2023.

[14] Yanghao Li, Haoqi Fan, Ronghang Hu, Christoph Feichtenhofer, and Kaiming He. Scaling language-image pre-training via masking. In *CVPR*, 2023.

[15] 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.

[16] OpenAI. Gpt-4 technical report. 2023.

[17] Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *ICCV*, 2015.

[18] 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.

[19] Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. Zero-shot text-to-image generation. In *ICML*, 2021.

[20] Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? In *ICML*, 2019.

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

[22] 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. In *NeurIPS*, 2022.

[23] Christoph Schuhmann, Richard Vencu, Romain Beaumont, Robert Kaczmarczyk, Clayton Mullis, Aarush Katta, Theo Coombes, Jenia Jitsev, and Aran Komatsuzaki. Laion-400m: Open dataset of clip-filtered 400 million image-text pairs. *arXiv preprint arXiv:2111.02114*, 2021.

[24] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[25] Quan Sun, Yuxin Fang, Ledell Wu, Xinlong Wang, and Yue Cao. Eva-clip: Improved training techniques for clip at scale. *arXiv preprint arXiv:2303.15389*, 2023.

[26] Haohan Wang, Songwei Ge, Zachary Lipton, and Eric P Xing. Learning robust global representations by penalizing local predictive power. In *NeurIPS*, 2019.

[27] Hu Xu, Saining Xie, Po-Yao Huang, Licheng Yu, Russell Howes, Gargi Ghosh, Luke Zettlemoyer, and Christoph Feichtenhofer. Cit: Curation in training for effective vision-language data. *arXiv preprint arXiv:2301.02241*, 2023.

[28] Jiahui Yu, Zirui Wang, Vijay Vasudevan, Legg Yeung, Mojtaba Seyedhosseini, and Yonghui Wu. Coca: Contrastive captioners are image-text foundation models. *arXiv preprint arXiv:2205.01917*, 2022.

[29] Lu Yuan, Dongdong Chen, Yi-Ling Chen, Noel Codella, Xiyang Dai, Jianfeng Gao, Houdong Hu, Xuedong Huang, Boxin Li, Chunyuan Li, et al. Florence: A new foundation model for computer vision. *arXiv preprint arXiv:2111.11432*, 2021.

[30] Wanrong Zhu, Jack Hessel, Anas Awadalla, Samir Yitzhak Gadre, Jesse Dodge, Alex Fang, Youngjae Yu, Ludwig Schmidt, William Yang Wang, and Yejin Choi. Multimodal c4: An open, billion-scale corpus of images interleaved with text. *arXiv preprint arXiv:2304.06939*, 2023.