MobileCLIP: Fast Image-Text Models through Multi-Modal Reinforced Training

Supplementary Material

A. Image Encoder Configurations

In our work, we introduce 3 stage configurations for FastViT architecture that substantially improves the model with limited impact on latency. The three configurations are described in Tab. 9. Comparison of our image encoders with FastViT image encoder when trained on ImageNet-1k dataset in a supervised setting (described in Appx. B) is shown in Fig. 7.

Variant	$\{C_1, C_2, C_3, C_4\}$	$\{L_1, L_2, L_3, L_4\}$
MCi0	$\{64, 128, 256, 512\}$	$\{2, 6, 10, 2\}$
MCi1	$\{64, 128, 256, 512\}$	$\{4, 12, 20, 4\}$
MCi2	$\{80, 160, 320, 640\}$	$\{4, 12, 24, 4\}$

Table 9. Configurations of MCi.

B. Experimental Setup

Additional details of our training and evaluation are provided in this section. Table 12 summarizes the hyperparameters we used to train MobileCLIP-B on DataCompDR-1B. For other variants of MobileCLIP (S0, S1, and S2) we use the same hyperparameters except using $\lambda = 1.0$. For experiments on DataCompDR-12M we use global batch size of 8192. All models trained on DataComp(-DR) use strong image augmentation unless stated otherwise.

For our ensemble distillation ablations in Appx. D, we use 32 total A100 GPUs but we use the same global batch size of 8192 as our other ablations. We also use a smaller uniformly sampled DataComp-8M for ablations in Apps. C and D that results in a slightly lower performance than DataCompDR-12M used for the rest of ablations.

The seen samples reported for DataCompDR is a triplet of one randomly augmented image, one ground-truth caption, and one randomly picked synthetic caption. The reported number of iterations is the number of seen samples divided by the global batch size.

For ImageNet-1k experiments, we follow the training recipe prescribed in [38, 59], i.e. the models are trained for 300 epochs using AdamW optimizer with weight decay of 0.05 and peak learning rate 10^{-3} for a total batch size of 1024. The number of warmup epochs is set to 5 and cosine schedule is used to decay the learning rate. The teacher model for distillation is RegNetY-16GF [48] Our implementation uses Timm library [66] and all the models were trained on single machine with 8×NVIDIA A100 GPUs. The hyperparameters for the three variants of MCi are detailed in Tab. 10. The performance of MCi variants is detailed in Tab. 11 and compared against recent state-of-art efficient architectures. MCi obtains the best trade-off amongst recent efficient architectures as seen in Fig. 7.

Hyperparameter	Training MCi0, MCi1, MCi2
Stochastic depth rate	[0.0, 0.05, 0.15]
Input resolution	256×256
Data augmentation	RandAugment
Mixup α	0.8
CutMix α	1.0
Random erase prob.	0.25
Label smoothing	0.1
Train epochs	300
Warmup epochs	5
Batch size	1024
Optimizer	AdamW
Peak learning rate	1e-3
LR. decay schedule	cosine
Weight decay rate	0.05
Gradient clipping	✗
EMA decay rate	0.9995

Table 10. Training hyperparameters for ImageNet-1k experiments.

C. Image Augmentation

In this section we provide a detailed ablation on the effect of image augmentations. The training setup is the same as training with DataCompDR-12M presented in Sec. 5.2, except we used an 8M subset for this ablation. In Tab. 13 we show classification and retrieval performance of a ViT-B/16 based CLIP model trained with our final loss as in Eq. (3) ($\lambda = 1$) and different image augmentations. Note that we

Model	Eval Image Size	Param (M)	FLOPs (G)	Mobile Latency (ms)	Top-1 Acc. (%)
MobileViG-M [44]	224	14.0	1.5	1.4	80.6
SwiftFormer-L1 [53]	224	12.1	1.6	1.5	80.9
EfficientFormerV2-S2 [38]	224	12.6	1.3	1.6	81.6
FastViT-SA12 [62]	256	11.5	1.9	1.5	81.9
MCi0 (ours)	256	11.8	2.4	1.5	82.2
MobileViG-B [44]	224	26.7	2.8	2.3	82.6
SwiftFormer-L3 [53]	224	28.5	4.0	2.6	83.0
EfficientFormerV2-L [38]	224	26.1	2.6	2.6	83.3
FastViT-SA24 [62]	256	21.5	3.8	2.4	83.4
MCi1 (ours)	256	21.9	4.7	2.5	83.8
FastViT-MA36 [62]	256	43.9	7.8	4.3	84.5
MCi2 (ours)	256	36.3	7.8	3.6	84.5

Table 11. Comparison of MCi variants with recent state-of-the-art models on ImageNet classification task.

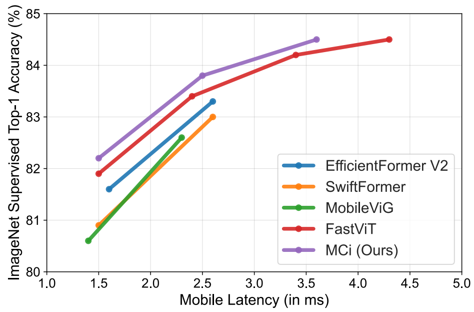

Figure 7. Top-1 Accuracy on ImageNet v/s latency plot of MCi variants and recent state-of-the-art architectures.

Hyperparameter	Value MobileCLIP-B, S0, S1, S2
Input resolution	224^2, 256^2, 256^2, 256^2
Context length	77
Data augmentation	RandAugment
Random resize crop scale	[0.08, 1.0]
Random resized crop ratio	[0.75, 1.33]
RangeAugment target value	(40, 20)
Train iterations	200k
Warmup iterations	2k
Global batch size	65536
Optimizer	AdamW
AdamW beta1	0.9
AdamW beta2	0.95
Max learning rate	1e-3
Min learning rate	1e-6
LR. decay schedule	cosine
Weight decay rate	0.2
Gradient clipping	✗
Mixed precision	BFloat16
EMA decay rate	0.9995
CLIP loss weight	0.25
KD loss weight	0.75
GT caption weight	1.0
Synth. caption weight	1.0
Synth. teacher	coca_ViT-L-14
Teacher 1	openai-ViT-L-14
Teacher 2	datacomp_xl_s13b_b90k-ViT-L-14
Teacher resolution	224×224

Table 12. Training hyperparameters for our CLIP experiments on DataCompDR.

feed the same augmented image to both teacher and student models. First, we consider `RandomResizedCrop` (RRC) with three magnitudes (0.08, 0.4, 0.9) determining the lower bound of random area of the crop (smaller lower bound means stronger augmentation). We observe that strong RRC results in significant accuracy improvement both for classification and retrieval metrics. While using strong RRC augmentation is standard for supervised training, for CLIP training the widely used recipe [47] includes weak RRC (lower-bound for scale= 0.9).

We further utilize `RangeAugment` [42] to automatically adjust Brightness, Contrast, and Noise. We use PSNR metric with target range [20, 40] and a Cosine curriculum. Since in `RangeAugment` individual augmentation magnitudes are adjusted dynamically during training, they cannot be stored as part of the dataset reinforcement process. Hence, we only apply it to images fed to the student model. We show that if the same augmentation is applied to both student and teacher (not feasible for our dataset reinforcement approach) further improvement can be obtained (56.6% vs 55.9% on ImageNet-val).

Finally, we consider `RandomHorizontalFlip`, `RandomErasing` [78], and `RandAugment` [6], and find that only `RandAugment` is beneficial in our setup. Our reinforced datasets include parameters of RRC and `RandAugment` and during training time we apply `RangeAugment` to images fed to the student model.

Image Augmentations	Zero-shot CLS		Flickr30k Ret.		COCO Ret.		Avg Perf.
	IN-val	IN-shift	I2T	T2I	I2T	T2I	on 38
RandomResizedCrop: `0.9-1.0` Student-RangeAugment [42]	51.0	40.1	54.2	68.5	30.5	45.3	45.9
RandomResizedCrop: `0.4-1.0` Student-RangeAugment	55.0	43.9	60.4	76.0	34.1	48.4	48.9
RandomResizedCrop: `0.08-1.0` Student-RangeAugment	55.9	44.6	58.8	76.1	34.2	49.0	49.6
RandomResizedCrop: `0.08-1.0`	56.4	44.6	59.8	74.6	34.4	49.3	49.1
RandomResizedCrop: `0.08-1.0` Student&Teacher-RangeAugment	56.6	44.9	60.2	74.0	34.9	50.5	50.8
RandomResizedCrop: `0.08-1.0` Student-RangeAugment RandomHorizontalFlip: `p=0.5`	55.9	44.7	59.4	75.9	34.4	49.2	48.8
RandomResizedCrop: `0.08-1.0` Student-RangeAugment RandomErasing [78]: `p=0.25`	55.8	44.5	59.4	75.3	34.5	49.7	49.1
RandomResizedCrop: `0.08-1.0` Student-RangeAugment RandAugment [6]	56.6	45.4	60.9	78.3	35.0	51.0	50.2

Table 13. Ablation on different augmentations for distillation. We highlight our choice with blue.

D. CLIP Ensembles

In this section we provide a detailed ablation on CLIP ensembles. First, we show that we can construct more accurate zero-shot models by ensembling pretrained individual CLIP models. For inference, we concatenate normalized embeddings of each modality followed by a re-normalization. In Tab. 14 we show performance of some CLIP ensemble models that we picked from OpenCLIP [29]. We also include performance of individual models. Evidently, ensembling results in improved performance. For example, an ensemble of two pretrained `ViT-L-14`-based CLIP models from `datacomp_xl_s13b_b90k` and `openai` results in average performance of 67.3%, while each individual model has 66.3% and 61.7% performance, respectively. Further, ensembling can be a more parameter efficient approach to obtain a stronger model. For instance, the ensemble of two `ViT-L-14`-based CLIP models has less parameters than the one with `ViT-bigG-14` image encoder, but comes with the same ImageNet-val performance (80.1%). In general, given a set of pretrained CLIP models (e.g., as in OpenCLIP [29]) with this approach we can push state-of-the-art and obtain stronger zero-shot performance. Here, we show and ensemble of four CLIP models can reach up to 81.7% zero-shot classification performance on ImageNet-val, while individual models' performance is not more than 80.1%. As stronger individual models become publicly available, one can create stronger ensembles with this approach.

In this work, we are interested in creating a strong ensemble model to be used as a teacher in the context of distillation. In Tab. 15 we show performance of a ViT-B/16 CLIP model trained with different CLIP models as teacher (both individual models and ensembles). Training setup is the same as that of in Sec. 5.2, except we use a uniformly sampled 8M subset. Similar to standard distillation for classification task [26], we observe that more accurate CLIP models are not necessarily better teachers. We picked the ensemble of two `ViT-L-14`-based CLIP models as the teacher model (highlighted in blue) in our dataset reinforcement process.

E. Ablations on Lossy Compressions

In general, the storage size of datasets depends on the file format and the tradeoff between load time and the compression rate. In Tab. 4c we presented the storage sizes for DataCompDR-12M and DataCompDR-1B with BFloat16 compression of the embeddings. In this section, we further analyze the storage reduction by i) reducing the number of augmentations, and ii) lossy compression of embeddings.

We report the total storage size for 12.8k samples of DataCompDR in Tab. 16. The storage size for DataCompDR-12M can be easily deduced by multiplying the numbers by 1000 (TBs instead of GBs) and by 10^5 for DataCompDR-1B.

Table 17 shows the accuracy of training with BFloat16 embeddings achieves accuracies within the standard deviation of the training on DataComp-12M.

F. Hybrid Text Encoder

In this section, we ablate over kernel dimensions for our hybrid text encoder. For this ablation, we use a 6-layered fully convolutional text encoder and systematically increase the kernel size. We use ViT-B/16 as the image encoder for these runs. These models were trained on DataCompDR-12M for 30k iterations. From Tab. 18, we notice that zero-shot IN-val performance does improve with increased kernel size, but it is significantly more expensive to run the model on mobile device. For zero-shot IN-val performance improvement of 1.1%, the model is 4.5× slower. From Tab. 18, kernel size of 11 obtains the best accuracy-latency trade-off.

For the hybrid design, we use depth-wise 2D convolutional layers. We reshape the 3 dimensional input tensor to `(BC1S)` format, i.e. `(Batch Size, Channel Dim., 1, Seq. length)` before feeding the tensor to the convolutional layer. For CLIP, the sequence length is set to 77. The depth-wise convolutions enable interactions between tokens across the sequence. The FFN layers enable interactions between token's channel dimensions. Since the convolution layer is 2D, we simply reuse the reparameterization process described in [62].

G. Performance of other models on DataCompDR-12M

In Tab. 19, we compare performance of CLIP models with different sized image encoders when trained on DataCompDR-12M. All models enjoy significant accuracy improvement when trained on DataCompDR-12M with no training overhead. For example, even the smallest model like MobileNetV3-L with only 4.9M parameters obtains a significant 10.6% improvement in zero-shot IN-val performance.

H. Extended Results

In this section we provide extended zero-shot results of our proposed family of CLIP models: MobileCLIP-S0, MobileCLIP-S1, MobileCLIP-S2, and MobileCLIP-B. Zero-shot classification and retrieval results are provided in Tab. 20. We also include additional results from related works where only partial evaluation is available.

I. Long training

In Tab. 21 we provide results for training MobileCLIP-B on more than 13B seen samples. We explore continuing the training of MobileCLIP-B to reduce the cost of training from scratch. Recently, [19] has shown that large scale CLIP models can be continually pretrained as the data distribution varies with time. We utilize some of their recipes for continual training where we initialize the training with a model previously trained with cosine or constant learning rate schedule and restart the training on DataCompDR-1B. We utilize a short warmup to stabilize the training and then use another constant or cosine learning rate schedule with maximum and minimum values equal to the original training. We train using 64 nodes with 8xA100-80GB GPUs and a per-GPU batch size of either 128 or 256. One seen sample for DataCompDR is a triplet of one randomly augmented image, one ground-truth caption, and one randomly picked synthetic caption. Number of iterations is the number of seen samples divided by the global batch size. Note that training wall-clock time is the same for DataCompDR vs DataComp (Tab. 4d).

Compared with our initial training on 13B seen samples, our long training with 39B total seen samples achieves 0.6% improvement in average performance on 38 datasets as well as 0.4% improvement in zero-shot IN-val accuracy. We reach

Teacher Models(s)	Teacher Pre-taining(s)	Teacher Resolution(s)	Zero-shot CLS		Flickr30k Ret.		COCO Ret.		Avg Perf. on 38
			IN-val	IN-shift	I2T	T2I	I2T	T2I	
ViT-bigG-14	laion2b_s39b_b160k	224	80.1	69.1	79.6	92.9	51.4	67.4	66.7
EVA01-g-14-plus	merged2b_s11b_b114k	224	79.3	69.3	79.0	91.7	50.3	68.2	66.2
ViT-L-14	datacomp_xl_s13b_b90k	224	79.2	67.9	73.4	89.0	45.7	63.3	66.3
ViT-L-14	openai	224	75.5	64.9	65.0	85.2	36.5	56.3	61.7
ViT-L-14-336	openai	336	76.6	67.1	66.9	87.7	37.1	57.9	62.8
ViT-L-14 ViT-L-14	datacomp_xl_s13b_b90k openai	224 224	80.1	69.6	74.5	92.3	46.7	66.5	67.3
ViT-L-14 ViT-L-14-336	datacomp_xl_s13b_b90k openai	224 336	80.5	70.6	75.8	91.8	47.0	67.0	67.8
EVA01-g-14-plus ViT-L-14 ViT-L-14	merged2b_s11b_b114k datacomp_xl_s13b_b90k openai	224 224 224	81.1	70.9	78.1	93.8	50.2	69.7	68.5
EVA01-g-14-plus ViT-L-14 ViT-L-14-336	merged2b_s11b_b114k datacomp_xl_s13b_b90k openai	224 224 336	81.2	71.6	78.8	93.7	50.2	69.9	68.9
convnext_xxlarge ViT-L-14 ViT-L-14-336	laion2b_s34b_b82k_augreg_soup datacomp_xl_s13b_b90k openai	256 224 336	81.5	71.7	79.0	94.5	50.5	69.5	68.7
ViT-bigG-14 EVA01-g-14-plus ViT-L-14 ViT-L-14	laion2b_s39b_b160k merged2b_s11b_b114k datacomp_xl_s13b_b90k openai	224 224 224 224	81.6	71.7	79.9	94.6	52.4	71.3	69.4
EVA01-g-14-plus ViT-L-14-336 ViT-L-14 convnext_xxlarge	merged2b_s11b_b114k openai datacomp_xl_s13b_b90k laion2b_s34b_b82k_augreg_soup	224 336 224 256	81.7	72.1	80.0	95.0	52.0	70.8	69.3
ViT-L-14 ViT-L-14-336 RN50x64 RN50x16	openai openai openai openai	224 336 384 448	78.2	68.9	73.4	89.7	42.0	63.5	65.5

Table 14. Zero-shot evaluation of (ensemble of) clip models. Each group of rows corresponds to an ensemble teacher. All models are taken from OpenCLIP [29] on `Aug-2023`. We highlight our choice with blue.

similar improvements in average performance on 38 datasets (0.4%) with only 18B total seen samples by continuing our original training on 13B seen samples with a short training using Cosine(40k, 131k, 2k).

Teacher Models(s)	Teacher Pre-taining(s)	Teacher Resolution(s)	Zero-shot CLS		Flickr30k Ret.		COCO Ret.		Avg Perf. on 38
			IN-val	IN-shift	I2T	T2I	I2T	T2I	
`ViT-bigG-14`	`laion2b_s39b_b160k`	224	53.4	42.6	59.6	76.2	35.8	52.1	47.8
`EVA01-g-14-plus`	`merged2b_s11b_b114k`	224	54.5	43.3	59.6	74.6	35.4	50.8	47.7
`ViT-L-14`	`datacomp_xl_s13b_b90k`	224	54.0	43.4	58.9	74.3	34.3	50.1	48.3
`ViT-L-14`	`openai`	224	54.4	42.7	54.5	69.1	29.7	44.6	47.2
`ViT-L-14-336`	`openai`	336	54.2	43.3	53.6	68.7	30.1	44.3	47.2
`ViT-L-14` `ViT-L-14`	`datacomp_xl_s13b_b90k` `openai`	224 224	56.3	44.8	59.2	74.5	34.4	49.9	49.6
`ViT-L-14` `ViT-L-14-336`	`datacomp_xl_s13b_b90k` `openai`	224 336	55.9	44.6	58.8	76.1	34.2	49.0	49.6

`EVA01-g-14-plus`
`ViT-L-14`
`ViT-L-14`
	`merged2b_s11b_b114k`
	`datacomp_xl_s13b_b90k`
	`openai`
		224
		224
		224
			56.2	45.0	59.6	76.9	35.7	51.5	49.4

`EVA01-g-14-plus`
`ViT-L-14`
`ViT-L-14-336`
	`merged2b_s11b_b114k`
	`datacomp_xl_s13b_b90k`
	`openai`
		224
		224
		336
			56.0	44.5	60.1	76.5	35.3	50.6	49.5

`convnext_xxlarge`
`ViT-L-14`
`ViT-L-14-336`
	`laion2b_s34b_b82k_augreg_soup`
	`datacomp_xl_s13b_b90k`
	`openai`
		256
		224
		336
			55.8	44.4	59.4	75.1	35.0	49.5	50.1

`ViT-bigG-14`
`EVA01-g-14-plus`
`ViT-L-14`
`ViT-L-14`
	`laion2b_s39b_b160k`
	`merged2b_s11b_b114k`
	`datacomp_xl_s13b_b90k`
	`openai`
		224
		224
		224
		224
			56.3	44.6	60.8	76.2	35.8	51.4	49.2

`EVA01-g-14-plus`
`ViT-L-14-336`
`ViT-L-14`
`convnext_xxlarge`
	`merged2b_s11b_b114k`
	`openai`
	`datacomp_xl_s13b_b90k`
	`laion2b_s34b_b82k_augreg_soup`
		224
		336
		224
		256
			55.9	44.6	60.4	75.1	35.6	52.3	49.4

`ViT-L-14`
`ViT-L-14-336`
`RN50x64`
`RN50x16`
	`openai`
	`openai`
	`openai`
	`openai`
		224
		336
		384
		448
			56.4	44.6	57.9	72.0	31.7	47.0	48.6

Table 15. Ablation on using different (ensemble of) teacher models in our multi-modal distillation. Each group of rows demonstrate an ensemble teacher. Student architecture is fixed to ViT-B/16 for image encoder and base 12-layer Transformer for text encoder (MobileCLIP-B setup). For this ablation, we use an 8M subset of DataComp and train all experiments for 20k iterations with global batch size of 8k. All models are imported from OpenCLIP [29] on `Aug-2023`. We highlight our choice with blue.

Image	Text	Syn.	Aug. Params	Text Emb.	Image Emb.	BFloat16	Sparsity	Size (GBs)
✓	✓	✗	✗	✗	✗	✗	✗	0.9
✓	✓	✓	✓	✗	✗	✗	✗	0.9
✓	✓	✓	✓	5+1	30	✗	✗	3.3
✓	✓	✓	✓	5+1	30	✓	✗	1.9
✓	✓	✓	✓	5+1	30	✗	50%	1.8
✓	✓	✓	✓	5+1	30	✓	50%	1.3
✓	✓	✓	✓	5+1	10	✗	✗	1.9
✓	✓	✓	✓	5+1	10	✓	✗	1.4
✓	✓	✓	✓	5	5	✗	✗	1.5
✓	✓	✓	✓	5	5	✓	✗	1.2
✓	✓	✓	✓	2	2	✗	✗	1.1
✓	✓	✓	✓	2	2	✓	✗	1.0

Table 16. Total storage for 12.8k samples stored in individual Pickle Gzip files. Storage for 12.8M and 1.28B samples are approximately the same numbers in TBs and 100 TBs.

Num. Aug.	1	2	5	10	15	20	25	30
w/o BFloat16	60.63	63.27	64.81	64.74	64.49	64.92	64.78	64.74
w/ BFloat16	-	-	64.32	64.88	64.57	64.81	65.13	64.91

Table 17. Effect of BFloat16 and the number of augmentations on ImageNet-val zero-shot Accuracy. We train on DataCompDR-12M for approximately 30 epochs.

Kernel Size	3	11	31
Num Params. (M)	38.2	38.3	38.4
Latency (ms)	1.0	1.2	5.4
IN-val	56.3	57.9	59.0

Table 18. Ablation on kernel size for text encoder. We train for 30k iterations. We highlight our choice with blue

Image Enc.	Dataset	# Image Enc. Params (M)	Latency (ms) (img+txt)	0-shot IN-val	Δ
MobileNetv3-L	DataComp-12M DataCompDR-12M (Ours)	4.9	1.1 + 3.3	34.1 **44.7**	↑+10.6
ViT-T/16	DataComp-12M DataCompDR-12M (Ours)	5.6	3.0 + 3.3	32.9 **44.1**	↑+11.2
ResNet-50	DataComp-12M DataCompDR-12M (Ours)	24.6	2.6 + 3.3	40.4 **51.9**	↑+11.5
FastViT-MA36	DataComp-12M DataCompDR-12M (Ours)	43.5	4.3 + 3.3	45.2 **58.9**	↑+13.7

Table 19. DataCompDR-12M vs. DataComp-12M. All the models are trained for 30k iterations ($\sim$ 0.24B seen samples).

Name	ImageNet Shifts CLS							Flickr30k Retrieval						COCO Retrieval					
								T→I			I→T			T→I			I→T		
	val	A	R	O	S	V2	Obj	@1	@5	@10	@1	@5	@10	@1	@5	@10	@1	@5	@10
MobileCLIP-B	76.8	58.7	89.6	41.4	64.5	69.8	69.4	77.3	94.4	96.7	91.4	99.1	99.9	50.6	74.9	82.9	68.8	88.3	92.9
MobileCLIP-S2	74.4	49.3	87.0	46.9	62.2	66.8	66.6	73.4	92.3	95.6	90.3	98.9	99.6	45.4	70.1	79.0	63.4	85.1	91.4
MobileCLIP-S1	72.6	40.3	84.7	50.5	60.3	64.9	63.4	71.0	91.3	95.3	89.2	98.0	99.5	44.0	68.9	77.7	62.2	84.3	90.1
MobileCLIP-S0	67.8	26.5	78.6	53.8	55.5	59.9	55.9	67.7	88.8	93.3	85.9	97.1	98.8	40.4	66.0	75.9	58.7	81.1	88.2
DIME-FM-B/32 [56]	66.5	32.2	69.8	(-)	46.5	58.9	43.2	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)	(-)
VeCLIP-B/16 [32]	64.6	(-)	(-)	(-)	(-)	57.7	(-)	76.3	93.5	96.4	91.1	98.5	99.7	48.4	73.3	81.8	67.2	87.3	92.7
TinyCLIP-63M/32 [68]	64.5	22.8	74.1	(-)	50.8	55.7	31.2	66.0	(-)	(-)	84.9	(-)	(-)	38.5	(-)	(-)	56.9	(-)	(-)
CLIPA-B/16 [34]	63.2	26.8	73.2	(-)	48.7	55.6	44.3	58.3	(-)	(-)	75.3	(-)	(-)	35.2	(-)	(-)	53.1	(-)	(-)

Table 20. Extended zero-shot evaluations. We also include additional results from related works where the full DataComp [18] evaluation was not accessible. Numbers are read from the corresponding papers. For each method we picked their best model up to ViT-B/16 size. Please see Tab. 7 for additional details including runtime benchmarking. Models are sorted by their zero-shot classification performance on ImageNet-val. Here our MobileCLIP-S1 is fully trained with 13B seen samples.

LR Schedule	Seen Samples	Zero-shot CLS		Flickr30k Ret.		COCO Ret.		Avg. Perf. on 38
		IN-val	IN-shift	T→I	I→T	T→I	I→T	
Cosine(200k, 65k, 2k)	13B	76.8	65.6	**77.3**	91.4	**50.6**	68.8	65.2
Const(300k, 65k, 2k) + Cosine(40k, 131k, 2k)	25B	**77.1**	65.8	77.0	91.8	50.2	68.7	65.2
Const(300k, 65k, 2k) + Cosine(300k, 65k, 2k)	39B	**77.2**	**66.1**	76.9	92.3	50.0	68.7	**65.8**
Const(200k, 65k, 2k) + Cosine(40k, 131k, 2k)	18B	**77.1**	**65.9**	77.0	**92.8**	50.3	**69.1**	64.6
Cosine(200k, 65k, 2k) + Cosine(40k, 131k, 2k)	18B	76.8	65.6	76.8	92.1	**50.4**	**69.1**	**65.6**
Cosine(100k, 131, 2k) + Cosine(40k, 131k, 2k)	18B	**77.0**	65.6	**77.2**	91.3	50.2	**69.2**	64.2

Table 21. **MobileCLIP-B long and continual training.** Retrieval performances are reported @1. Last column shows average performance on 38 datasets as in OpenCLIP [29]. The learning rate schedules are specified as Cosine/Const(num-iterations, global batch-size, warmup-iterations). We highlight numbers within 0.2% of the maximum in each column.