

SLIP: Self-supervision meets Language-Image Pre-training

Norman Mu¹ Alexander Kirillov² David Wagner¹ Saining Xie²

¹UC Berkeley, ²Facebook AI Research (FAIR)

Code: <https://github.com/facebookresearch/SLIP>

Abstract

Recent work has shown that self-supervised pre-training leads to improvements over supervised learning on challenging visual recognition tasks. CLIP, an exciting new approach to learning with language supervision, demonstrates promising performance on a wide variety of benchmarks. In this work, we explore whether self-supervised learning can aid in the use of language supervision for visual representation learning. We introduce SLIP, a multi-task learning framework for combining self-supervised learning and CLIP pre-training. After pre-training with Vision Transformers, we thoroughly evaluate representation quality and compare performance to both CLIP and self-supervised learning under three distinct settings: zero-shot transfer, linear classification, and end-to-end finetuning. Across ImageNet and a battery of additional datasets, we find that SLIP improves accuracy by a large margin. We validate our results further with experiments on different model sizes, training schedules, and pre-training datasets. Our findings show that SLIP enjoys the best of both worlds: better performance than self-supervision (+8.1% linear accuracy) and language supervision (+5.2% zero-shot accuracy).

1. Introduction

Much of the recent progress in deep learning has been driven by the paradigm of pre-training powerful, general-purpose representations that transfer well to a variety of specific applications. Within computer vision, supervised learning on image classification and self-supervised learning on unlabeled images comprise the two primary approaches to representation learning. After AlexNet [37], researchers soon realized that supervised pre-training yields a generic visual backbone which can be repurposed for many different tasks [21]. Today, most state-of-the-art results still depend on supervised pre-training, and scaling to massive amounts of data, such as Google’s proprietary JFT dataset, remains one of the most reliable methods for improving downstream performance. Self-supervised learning, a form of unsupervised learning, found tremendous success first in the domain of language [16, 46], but has also

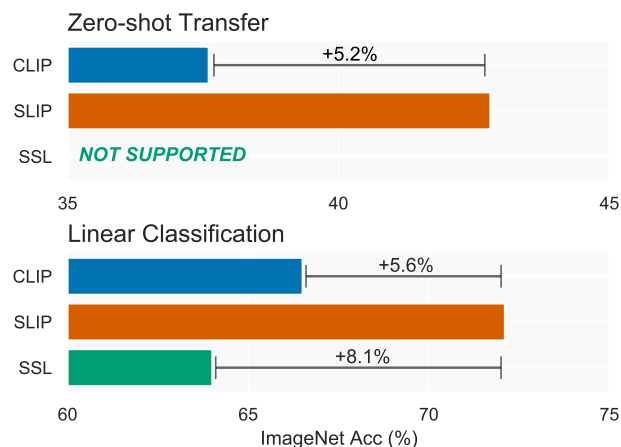


Figure 1. **SLIP pre-training on YFCC15M.** Combining image-only self-supervision and image-text supervision simultaneously improves zero-shot transfer and linear classification on ImageNet.

made significant recent progress in vision. A major motivation for studying self-supervised learning has been a desire to supersede supervised pre-training and its reliance on labor-intensive human annotation. Indeed, self-supervised pre-training has outperformed supervised learning for some time now on small datasets, but only recently with the development of contrastive methods [9, 27] has it begun to improve performance on larger datasets such as ImageNet.

Both supervised and self-supervised pre-training today rely heavily on ImageNet (*i.e.* ImageNet-1K) [47], a highly curated dataset with particular idiosyncrasies and biases [55]. The YFCC100M dataset [53] was released in 2015 and remains the largest publicly-accessible collection of images. To date, the field of representation learning has found much less use for this dataset. On the other hand, the full ImageNet dataset of 14M images (*i.e.* ImageNet-22K) has become very popular for its role in training Vision Transformer models which require a larger amount of data than ImageNet-1K [17, 1]. Why are uncured datasets not more common in the study of representation learning? There are a few possible reasons. Most immediately, uncured datasets also lack labels and so long as supervised pre-training remains the simpler and more accessible option for most researchers, datasets like YFCC100M are a non-starter. As we confirm again later in this work, the

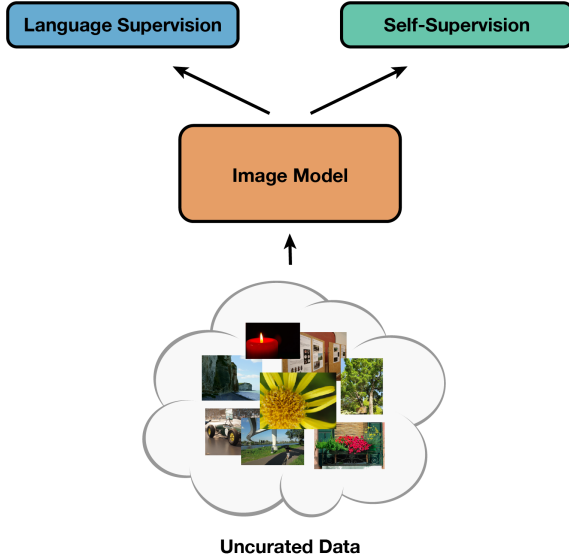


Figure 2. Illustration of SLIP, our multi-task framework. An image model has access to and can be trained with both language supervision from captions and self-supervision on images.

standard self-supervised evaluation task of ImageNet classification from frozen features heavily biases results against models not also pre-trained on ImageNet [5]. Finally, while progress on ImageNet has been encouraging, there has not been strong evidence that current self-supervised methods scale well to larger uncurated datasets [54].

Recently, CLIP [45] introduced an exciting new approach to representation learning. It re-examines language supervision for learning visual representations, and catapults it into contention with label supervision and self-supervision. CLIP requires only images and free-form text captions, thus revitalizing the use of YFCC100M in representation learning. In addition to no longer requiring label annotations, CLIP accuracy also scales well to large datasets and models. The best results for CLIP are achieved with big models on a curated dataset of 400M images and captions, though promising results are also shown on a subset of YFCC100M. CLIP also enables many exciting new applications with its flexible language-guided capabilities.

In this work, we explore whether the momentum of self-supervised learning on images carries into the setting of language supervision. In particular, we investigate whether language supervision in the form of CLIP also benefits from image self-supervision. We note that it is not immediately clear that these two training objectives should be stronger together. The two objectives each require the model to encode qualitatively different and conflicting information about the image, leading to interference.

In order to explore these questions, we introduce SLIP (Self-supervision meets Language-Image Pre-training), a multi-task framework combining language supervision and

self-supervision. We pre-train various SLIP models on a subset of YFCC100M, and thoroughly evaluate representation quality under three distinct settings: zero-shot transfer, linear classification, and end-to-end finetuning. We evaluate downstream performance on ImageNet, in addition to a battery of 25 other classification benchmarks. Additionally, we further validate our findings with experiments on different model sizes, training schedules, and pre-training datasets. Our findings conclusively show that SLIP improves performance across most evaluations by a significant margin, an encouraging signal for the general utility of self-supervision in the context of language supervision. Additionally, we analyze various components of our method in further detail such as the choices of pre-training dataset and data processing method. We conclude with a discussion of our evaluations as well as the ethical and practical limitations of this class of methods.

2. Related Work

Language supervision. Early work explored learning visual representations from image captions, even before the advent of deep learning [44]. DeViSE [19] jointly embeds images and textual class labels within a shared semantic space, allowing the model to recognize classes that were not explicitly trained for. Initial attempts at leveraging the YFCC dataset for representation learning included predicting the bag-of-words representation [33] or n-gram occurrence [39] from images. ICMLM [48] and VirTex [15] showed that language supervision on COCO Captions produced useful visual representations. Prior to CLIP, Multimodal Contrastive Training [62] adds contrastive image-image and language-image losses to VirTex which further improve performance. CLIP [45] quickly garnered significant attention for its simplicity, scale, and strong results. Developed concurrently, ALIGN [31], uses a larger but noisier uncurated dataset and shows similar results.

Self-supervised learning. Earlier self-supervised learning methods have shown subpar scaling with dataset size [24]. Contrastive learning methods ushered in rapid progress [57, 60, 27, 9] due to their simplicity and effectiveness. Recent methods for self-supervised learning also propose a variety of alternatives to the contrastive objective such as self-distillation [25, 6], or input reconstruction [1, 26].

Multi-modal multi-task learning. MURAL [30] extends ALIGN to the multi-lingual setting and introduces a cross-lingual objective to improve multi-lingual image and text retrieval. Concurrently to this work, DeCLIP [40] adds several additional training objectives and more data collected in-house to CLIP in order to improve data efficiency.

3. SLIP Framework

We introduce SLIP, a framework for combining language supervision and image self-supervision to learn visual representations without category labels. During pre-training, separate views of each input image are constructed for the language supervision and image self-supervision branches, then fed through a shared image encoder. Through the course of training, the image encoder learns to represent visual input in a semantically meaningful manner. We can then measure the quality of these learned representations by evaluating their utility in downstream tasks.

3.1. Contrastive Language-Image Pre-training

Radford et al. [45] demonstrated the ability of contrastive learning (CLIP) on corresponding images and captions to learn powerful visual representations. CLIP first embeds images and text with separate modality-specific models. These vectors are then projected into a shared embedding space and normalized. The InfoNCE loss is computed using these final embeddings, with corresponding images and captions as positive pairs and all non-matching images and captions as negative pairs.

Non-contrastive alternatives for language supervision include predicting the bag-of-words representation of the caption [33] or the original caption [48, 15] from the image. However, these methods appear to yield weaker results than CLIP. The contrastive objective also enables image classification without re-training dataset-specific classification layers (zero-shot transfer).

3.2. Image Self-Supervision

View-based self-supervised learning, in which models are trained to represent views or augmentations of the same image similarly, has yielded strong results across a variety of different formulations. In this work we primarily use an adaptation of SimCLR [9, 10], a representative example of these methods, as the self-supervised objective in SLIP. However, other frameworks can be swapped in quite easily, and we explore this in Section 6. We focus on the Vision Transformer [17] architecture for its simplicity and good performance. We follow MoCo v3 [11] on the hyperparameter settings for training self-supervised Vision Transformers, which will be described later in Section 4.1.

3.3. Our Method

We outline SLIP with SimCLR for self-supervision (i.e. SLIP-SimCLR) in Algorithm 1. During each forward pass in SLIP, all images are fed through the same encoder. The CLIP and SSL objectives are computed on the relevant embeddings and then summed together into a single scalar loss. The two objectives can be balanced differently by rescaling the SSL objective. We find that a scale of 1.0 for the self-

supervised objective, i.e. no re-scaling, works well for SimCLR. Unless otherwise noted, we refer to SLIP-SimCLR simply as SLIP.

SLIP increases the number of images processed which results in approximately $3\times$ more activations. This expands the model’s memory footprint and slows down the forward pass during training. See Section 7 for further discussion.

4. Improved Training Procedure

The authors of CLIP focus primarily on training with a large private dataset of 400M image-text pairs, where the large scale of data lessens the need for regularization and data augmentation. While re-implementing CLIP, we found some simple adjustments primarily to data augmentation which significantly improved performance when pre-trained on YFCC15M. Our improved training procedure achieves 34.6% zero-shot transfer to ImageNet with a modified¹ ResNet-50, exceeding the original result of 31.3%. Another re-implementation achieves 32.7% accuracy on ImageNet [29]. In our experiments we focus primarily on the Vision Transformer model family for their strong scaling behavior [17]. We train all Vision Transformer models with our improved procedure as well, in order to set strong baselines for comparing our methods.

4.1. Implementation Details

Datasets. We focus primarily on a 15M subset of YFCC100M [53] filtered by Radford et al. [45] consisting of English-only titles and descriptions, which we refer to as YFCC15M. We also evaluate on Conceptual Captions 3M (CC3M) [49] and Conceptual Captions 12M (CC12M) [8].

Data Augmentation. During training, we randomly sample a valid caption for each image (i.e. title or description for YFCC15M). Images for the CLIP branch are randomly resized and cropped to between 50% and 100% of the original image, which we refer to as global cropping. In the self-supervised branch we sample two views with the augmentation from MoCo v3 [9].

Architecture. We use the original ViT-B/16 and ViT-L/16 architectures from the ViT paper [17] for our image encoders, as well as a ViT-S/16 architecture [56] which is comparable to ResNet-50 in FLOPs and parameters. For our text encoders, we use the smallest text Transformer model from CLIP which contains 38M parameters and uses byte-pair encoding with a 49K token vocabulary, and maximum context length of 77.

For the CLIP objective, our model projects the image and caption embeddings into a 512-dim space with separate learned linear projections. In the self-supervised branch, we

¹The initial 7×7 conv is replaced by three 3×3 convs; global average pooling is replaced by a self-attention pooling layer with 14M parameters.

use the 3-layer MLP projection head with 4096-dim hidden layers to transform the image embeddings into a 256-dim output space.

Training. We train with a batch size of 4096 and the AdamW optimizer in all our experiments. Following CLIP, we set the $\beta_2 = 0.98$ to improve training stability, but we keep $\epsilon = 1e-8$. We use a weight decay of 0.5 for CLIP and 0.1 for SLIP. Instead of the custom mixed-precision recipe used in CLIP, we opt for the built-in automatic mixed precision library in PyTorch.

Zero-shot Transfer Evaluation. We evaluate zero-shot transfer to various classification benchmarks including ImageNet. We perform prompt ensembling by averaging the caption embeddings for each class across the prompt templates. This average caption embedding is then used to compute cosine similarity with the image embeddings. CLIP provides prompt templates and class names for these benchmarks, which we use directly for ease of comparison.

Linear Classification Evaluation. We use the same setup as MoCo v3 to evaluate linear classification performance. We use SGD w/ momentum and no weight decay. On ImageNet, we use a learning rate of 0.01 and on the other downstream datasets we tune the learning rate and report the best result. We train for 100 epochs and perform standard cropping and flipping augmentations.

End-to-end Finetuning Evaluation. To finetune our models on ImageNet, we use the training procedure from BeiT [1]. This procedure employs significant regularization and data augmentation, as well as layerwise learning rate decay which exponentially decays the learning rate across layers. We disable relative positional embedding, layer scaling, and average pooling across tokens. For ViT-B and ViT-S we train for 100 epochs, while on ViT-L we train for 50 epochs.

For finetuning on smaller downstream datasets, we use the simpler DeiT training procedure [56].

5. Empirical Evaluations

5.1. ImageNet Classification

We evaluate performance on ImageNet under three distinct settings: zero-shot transfer, linear classification, and end-to-end finetuning. The zero-shot transfer task evaluates model performance on classification benchmarks directly after pre-training without updating any of the model weights. A model trained with contrastive language supervision can be used as an image classifier by simply selecting the class whose caption embedding aligns most closely with the input image. Linear classification, also called linear probing, is a standard evaluation method used to evaluate unsupervised or self-supervised representations. A randomly initialized final classification layer is trained while all other model weights are frozen. Finally, another way

Algorithm 1 SLIP-SimCLR: PyTorch-like Pseudocode

```
# fi, ft: image, text encoders
# hi, ht: CLIP image, text projectors
# hs: SimCLR projector
# c: SimCLR loss scale
def forward(img, text):
    xi, x1, x2 = crop(img), aug(img), aug(img)
    yt = tokenize(text)

    wi, w1, w2 = fi(xi, x1, x2)
    wt = ft(yt)

    z1, z2 = hs(w1), hs(w2) # SSL embed: N x C2
    zi, zt = hi(w1), ht(wt) # CLIP embed: N x C1

    loss = c * simclr(z1, z2) + clip(zi, zt)
    return loss

# s: learnable log logit scale
def clip(zi, zt):
    zi, zt = normalize(zi, zt)
    label = range(N)
    logit = exp(s) * zi @ zt.T

    li = CrossEntropy(logit, label)
    lt = CrossEntropy(logit.T, label)

    loss = (li + lt) / 2
    return loss

# tau: softmax temperature
def simclr(z1, z2):
    z1, z2 = normalize(z1, z2)
    label = range(N)
    mask = eye(N) * 1e9

    logit = z1 @ z2.T
    logit1 = z1 @ z1.T - mask
    logit2 = z2 @ z2.T - mask

    logit1 = cat(logit, logit1)
    logit2 = cat(logit.T, logit2)

    l1 = CrossEntropy(logit1 / tau)
    l2 = CrossEntropy(logit2 / tau)

    loss = (l1 + l2) / 2
    return loss
```

Notes: @ is the matrix multiplication operator. $k.T$ is k 's transpose. `eye` constructs an identity matrix. `cat` concatenates two matrices.

of evaluating representation quality is whether a pre-trained model can improve upon the performance of supervised learning when finetuning the model end-to-end.

Dataset	Method	Linear	Finetuning
ImageNet	SimCLR	74.5	82.8
	MoCo v3	76.6	83.1
YFCC15M	SimCLR	64.0 (-10.5)	82.5 (-0.3)
	MoCo v3	66.1 (-10.5)	82.8 (-0.3)

Table 1. We train ViT-B/16 with two self-supervised frameworks and find that both linear classification and end-to-end finetuning accuracy on ImageNet suffers when pre-training on YFCC15M instead of ImageNet. Accuracy drop show in (red).

One common evaluation setup in the self-supervised learning literature is to train both the model and the linear classifier on ImageNet (i.e. ImageNet-1K), which even without labels is a highly curated and class-balanced dataset. In Table 1 we train ViT-B/16 with SimCLR and

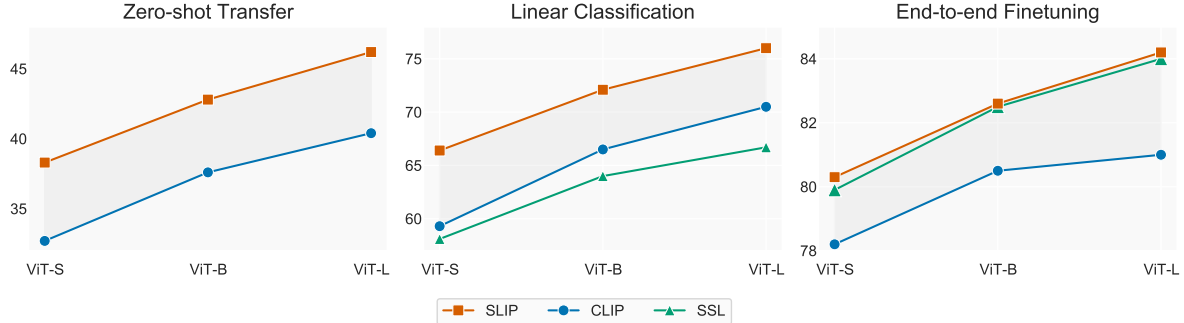


Figure 3. **ImageNet results.** We evaluate the representation quality by testing the performance on ImageNet under different settings: zero-shot transfer using text prompts, linear classification, and end-to-end finetuning. SLIP improves upon the zero-shot transfer and linear classification performance of CLIP by significant margin across all vision Transformer model sizes.

MoCo v3 on both YFCC15M and ImageNet. The resulting models are evaluated on ImageNet using linear classification and end-to-end finetuning. Both SimCLR and MoCo v3 experience a more than 10% drop in linear classification accuracy when pretrained on YFCC15M instead of ImageNet, a dramatic degradation in performance. For this reason, the baseline linear results in our experiments are lower than what is typically reported in the self-supervised literature. Similarly, we observe a less severe but consistent degradation for end-to-end finetuning results as well. We argue that training on uncurated data is a more realistic and informative setting, especially given the original motivations of learning vision from less supervision.

In Table 2, we provide evaluation results for CLIP, SimCLR, and SLIP across three sizes of Vision Transformer and on all three ImageNet settings. All models are trained for 25 epochs on YFCC15M. We find that language supervision and image self-supervision interact constructively in SLIP, improving upon the performance of both methods alone.

Model	Method	0-shot	Linear	Finetuned
ViT-S/16	CLIP	<u>32.7</u>	<u>59.3</u>	78.2
	SimCLR	-	58.1	<u>79.9</u>
	SLIP	38.3 (+5.6)	66.4 (+7.1)	80.3 (+0.4)
ViT-B/16	CLIP	<u>37.6</u>	<u>66.5</u>	80.5
	SimCLR	-	64.0	<u>82.5</u>
	SLIP	42.8 (+5.2)	72.1 (+5.6)	82.6 (+0.1)
ViT-L/16	CLIP	<u>40.4</u>	<u>70.5</u>	81.0
	SimCLR	-	66.7	<u>84.0</u>
	SLIP	46.2 (+4.8)	76.0 (+5.5)	84.2 (+0.2)

Table 2. Full ImageNet results. SLIP significantly improves performance on ImageNet in the zero-shot transfer, linear classification, and end-to-end finetuning settings. Improvements over stronger baseline (underlined) shown in green.

Zero-shot transfer. Self-supervised models do not support zero-shot transfer evaluation since there is no way to directly map the learned representations onto categorical labels. SLIP consistently outperforms CLIP by around +5%

on zero-shot transfer across all three model sizes, a very large margin relative to the original number. The gap between SLIP and CLIP does close slightly between ViT-Small (22M params) and ViT-Large (300M params) from +5.6% to +4.8%. This trend suggests that SLIP would continue to yield benefits over CLIP even for the largest Vision Transformer architectures currently in use.

With ViT-Large SLIP achieves 46.2% top-1 accuracy, which is far below the performance achieved by smaller models pre-trained on massive curated datasets. In absolute terms however, this is a very surprising result considering that YFCC15M contains very little data of the specific form seen during zero-shot transfer evaluation (i.e. object-centered images labeled with captions of the form “a photo of a class name”).

Linear Classification. In this setting we also observe the synergy between language supervision and image self-supervision. CLIP outperforms SimCLR, but by a much smaller margin than SLIP outperforms SimCLR. We see that SLIP significantly outperforms SimCLR in linear classification accuracy across all three model sizes. The gap between SLIP and SimCLR is largest with ViT-L at almost +10%, suggesting that SLIP continues to scale with larger models while SimCLR slightly saturates in performance.

End-to-end Finetuning. We see in Table 1 that finetuning performance is somewhat less affected by pre-training on YFCC15M than linear performance is affected, possibly because the model is allowed to adapt to the target distribution. Both SimCLR and MoCo v3 experience -0.3% drops in finetuning accuracy when pre-trained on YFCC15M instead of ImageNet, which is still quite significant for this setting. We re-iterate that the results in Table 2 are not directly comparable with methods which are pre-trained on ImageNet-1K.

When finetuning on ImageNet, CLIP is particularly weak: ViT-S and ViT-B performance is below even that of training from a random weight initialization [56]. The

performance of CLIP does not scale well with model size either, as CLIP ViT-L performance is only +0.5% above CLIP ViT-B. On the other hand, self-supervised learning does quite well in this setting, especially with the larger models. SimCLR ViT-L enjoys a +3.0% gain in accuracy over CLIP ViT-L, and SLIP ViT-L does slightly better than SimCLR ViT-L, though by a very marginal amount. These results suggest that the subpar finetuning performance of CLIP is mostly solved with self-supervision.

5.2. Model and Compute Scaling

We also investigate the scaling behavior of SLIP with more compute (longer training) and larger vision models. We note that 100 epochs of training on YFCC15M corresponds to around 1200 epochs of training on ImageNet-1K. In Table 3 we experimented with holding model size fixed (ViT-B/16) and training for longer as well as training different model sizes for an extended training schedule (100 epochs). Our results indicate that SLIP scales well with both longer training and larger models. We show full results simultaneously varying model and compute scaling with SLIP in the appendix.

Model	#params.	0-shot	Linear	Finetuned
ViT-S	22M	39.5	68.3	80.7
ViT-B	86M	45.0	73.6	83.4
ViT-L	307M	47.9	75.1	84.8

(a) Comparing ViT model variants of different capacities (ViT-S/B/L). All models are pre-trained for 100 epochs.

Epochs	0-shot	Linear	Finetuned
25	42.8	72.1	82.6
50	44.1	73.0	82.9
100	45.0	73.6	83.4

(b) ViT-B/16 with longer pre-training schedules (25/50/100 epochs).

Table 3. SLIP pre-training performance (in terms of zero-shot transfer, linear classification, and end-to-end finetuning) can scale well with both model size and number of training epochs.

5.3. Additional Benchmarks

While evaluating classification performance on ImageNet gives a broad overview of representation quality, it is also informative to measure performance on a variety of narrowly targeted downstream datasets. In Table 4 we evaluate zero-shot transfer on a battery of downstream image classification tasks compiled by [45]. These datasets span many different domains including everyday scenes such as traffic signs, specialized domains such as medical and satellite imagery, video frames, rendered text with and without visual context, and more. We remove Pascal VOC and replace NABirds with CUB-200-2011. To preprocess the datasets into a unified pipeline we use the extra scripts included in VISSL [23]. We catalog chance performance

along with short descriptions of the datasets in the appendix.

For a given model size, the relative ranking between methods appears surprisingly inconsistent. On a few datasets such as Rendered SST2, KITTI depth, and Patch-Camelyon (PCAM) it appears at first glance that smaller models and less training improve performance. However we note that performance on these datasets is only around chance performance, likely because these datasets have little overlap with the semantic distribution of YFCC15M, and thus an unreliable indicator of representation quality. Performance is stronger on categories are well represented in YFCC15M, such as Food-101, Oxford Pets, Caltech-101, and STL-10. On these datasets we see that larger models and training for longer with SLIP more generally improve zero-shot transfer accuracy. We view these results on tasks with more reasonable representation in YFCC15M as more informative of representation quality.

Zero-shot performance on the low-resolution datasets (MNIST, CIFAR-10, CIFAR-100) is also very poor. On many datasets performance is several multiples of chance performance yet still much lower than what is achievable with a lightweight model trained on a modest amount of application-specific data. This suggests that language supervision alone is an inefficient way of training models for specific tasks of interest.

We also provide linear classification results on these benchmarks in the appendix.

5.4. Additional Pre-training Datasets

In addition to YFCC15M, we experiment with two additional image-text datasets: CC12M and CC3M. In Table 5, we train ViT-B/16 with both SLIP and CLIP on CC12M and CC3M, and compare against our previous numbers on YFCC15M. SLIP maintains its margin of improvement over CLIP in all ImageNet evaluation settings. Notably, pre-training SLIP on CC12M instead of YFCC15M yields lower zero-shot accuracy but actually results in higher linear and finetuning performance. CLIP sees an even more surprising boost to finetuning performance of +1.6%.

Our improved training recipe (see Section 4.1) largely alleviates overfitting by CLIP on YFCC15M and CC12M, but on the smaller CC3M dataset CLIP overfits quite dramatically. This may be due to the hypernymization used in CC3M to make the captions more amenable to image captioning. CLIP reaches its highest zero-shot ImageNet accuracy after just 15 out of 40 epochs of training on CC3M, after which we observe a steady decline in ImageNet accuracy. In contrast, on CC3M SLIP reaches its highest zero-shot ImageNet performance after 35 epochs.

5.5. Alternative Self-Supervised Frameworks

As noted in Section 3.2, SLIP enables the use of many different self-supervision methods. We ran several experi-

		Food-101	CIFAR-10	CIFAR-100	CUB	SUN397	Cars	Aircraft	DTD	Pets	Caltech-101	Flowers	MNIST	FER-2013	STL-10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCAM	UCF101	Kinetics700	CLEVR	HatefulMemes	SST2	ImageNet	Average
ViT-S	CLIP	43.4	61.0	29.9	31.1	43.9	3.1	4.7	17.9	25.0	53.3	47.8	9.8	29.1	86.8	22.3	16.1	9.5	34.1	8.7	64.8	26.0	18.8	14.7	56.1	49.5	32.7	32.3
	SLIP (25 ep)	51.6	73.0	35.4	36.3	49.2	4.2	6.1	25.7	30.9	62.8	54.3	9.9	31.3	91.6	22.4	21.9	11.0	39.9	9.6	50.8	32.8	22.9	14.8	49.6	50.1	38.3	35.6
	SLIP (100 ep)	53.0	68.4	39.3	36.5	49.8	4.6	5.1	26.6	33.6	68.3	55.8	2.7	37.8	91.9	18.2	22.2	13.8	38.4	8.5	62.8	33.3	23.5	19.2	51.4	49.4	39.5	36.7
ViT-B	CLIP	50.6	66.0	34.5	38.8	51.1	4.0	5.4	21.2	28.5	60.9	53.3	8.4	17.3	90.5	30.2	21.5	6.1	35.1	10.5	53.5	28.5	22.1	10.8	52.4	50.7	37.6	34.2
	SLIP (25 ep)	59.5	78.6	45.2	38.7	53.4	5.4	5.7	26.1	31.1	71.0	56.6	9.8	19.6	94.4	20.3	28.9	14.5	34.0	11.6	55.4	37.7	26.9	17.5	52.8	51.1	42.8	38.0
	SLIP (100 ep)	63.3	79.2	50.4	44.7	52.0	8.1	8.4	26.2	34.7	74.0	61.3	17.1	40.8	95.4	20.8	27.8	11.7	35.2	11.5	52.1	37.1	25.8	13.0	55.1	49.9	45.0	40.0
ViT-L	CLIP	59.5	72.9	41.5	40.3	53.6	6.9	6.4	20.6	27.9	65.4	55.0	10.3	34.5	94.2	22.7	28.8	5.8	41.4	12.6	54.9	34.3	24.0	12.9	54.3	50.1	40.4	37.4
	SLIP (25 ep)	64.4	87.8	56.4	39.8	58.9	8.6	7.8	26.8	32.0	76.6	59.4	13.2	36.0	96.6	27.7	36.5	7.2	28.8	15.6	54.4	42.6	30.0	14.1	53.4	50.1	46.2	41.2
	SLIP (100 ep)	69.2	87.5	54.2	39.8	56.0	9.0	9.5	29.9	41.6	80.9	60.2	14.9	39.6	96.2	34.5	46.0	8.6	30.7	14.2	50.6	44.1	30.5	17.4	55.0	49.8	47.9	43.0

Table 4. Zero-shot transfer evaluation with ViT S, B, and L on a variety of classification benchmarks. Best results in **bold**. SLIP outperforms CLIP on most of the tasks, frequently with a significant margin. With longer pre-training epochs, the performance can be further improved.

Dataset	Method	0-shot	Linear	Finetuned
CC3M	CLIP	17.1	53.3	79.5
	SLIP	23.0	65.4	81.4
CC12M	CLIP	36.5	69.0	82.1
	SLIP	40.7	73.7	83.1
YFCC15M	CLIP	37.6	72.1	80.5
	SLIP	42.8	66.5	82.6

Table 5. SLIP vs. CLIP ImageNet with ViT-B/16 on CC3M [49], CC12M [8], two smaller uncuration datasets.

Method	0-shot	Linear	Finetuned
SLIP-SimCLR [9]	42.8	72.1	82.6
SLIP-MoCo v3 [11]	41.8	71.4	82.4
SLIP-BYOL [25]	41.3	71.1	82.2
SLIP-BEiT [1]	39.1	66.5	82.2
None (CLIP)	37.6	66.5	80.5

Table 6. We evaluate ViT-B/16 with several SLIP variants using different self-supervised frameworks. SLIP works the best with SimCLR among several other self-supervised frameworks, but all variants outperform CLIP.

ments on ViT-B/16 with different alternatives to SimCLR, in particular MoCo v3 [11], BYOL [25], and BEiT [1]. Similar to how we tuned the hyperparameters for SLIP-SimCLR, we largely keep the original self-supervised hyperparameters and add in the CLIP objective and text encoder. MoCo v3 and BEiT are already designed for ViT, but with BYOL we tuned the learning rate and weight decay while copying the data augmentation and projector/predictor architecture from MoCo v3. We also lightly tune different scaling parameters for the self-supervised loss. All models are trained for 25 epochs on YFCC15M.

Our results in Table 6 show that all three alternatives underperform SLIP-SimCLR, despite being individually stronger self-supervised methods. Most surprising is the result that SLIP-BEiT performs the worst despite BEiT being the strongest self-supervised method tested here. This may be due to a greater input discrepancy between pre-training and deployment stage. Nonetheless, all these suboptimal variants of SLIP still improve performance over CLIP.

6. Further Analysis

Why not pre-train with SSL and finetune with CLIP?

An alternative to SLIP would be to simply initialize the image encoder of CLIP with SSL-trained weights. We tried training CLIP ViT-B/16 under this setting but found worse performance than training jointly with CLIP and SSL. Progress after a few training epochs exceeds that of SLIP at the same point in training, but stalls throughout the rest of training (25 epochs). In 7, we see this approach underperforms SLIP in all three ImageNet evaluation settings.

Is SLIP just CLIP with data augmentation?

We examine the effects of adding further data augmentation to CLIP and whether this explains the performance improvements seen in SLIP. The SimCLR augmentation can be separated into two components: color (jitter or grayscale) + blur, and re-size crop + flip. We train CLIP with these two components individually and also with the full SimCLR augmentation. When training with color + blur, we use the original CLIP cropping strategy from [45] in which we resize the shorter side to 224px then perform a random square crop. Our results are shown in Table 8. While augmentation and re-size crop + flip hurt performance, color + blur do improve zero-shot transfer performance by +0.8% which is still far below the gain by SLIP.

Can we fully decouple self-supervision from language supervision?

We experimented with a version of SLIP we call SLIP-decoupled in which the self-supervised objective is computed on a disjoint set of 15M images from the YFCC15M images used in the text supervision object. During training, the images are sampled independently from both sets, effectively decoupling the language-image supervision and self-supervision signals. In Table 9, we find that SLIP-decoupled does just as well as SLIP.

7. Discussion

Our results on ImageNet and other classification benchmarks show that language supervision and self-supervision are indeed highly synergistic. As shown in Table 2, SLIP

Method	0-shot	Linear	Finetuned
SimCLR \rightarrow CLIP	41.1	68.2	82.3
SLIP-SimCLR	42.8	72.1	82.6

Table 7. Finetuning vs. multi-task training. One alternative to SLIP consists of initializing the image encoder of CLIP with weights trained through self-supervised learning. With ViT-B/16 trained for 25 epochs, finetuning with CLIP performs noticeably worse across all three ImageNet evaluating settings.

Augmentation	0-shot	Linear	Finetuned
global crop (CLIP)	37.6	66.5	80.5
color + blur	38.4	68.5	81.5
resize crop + flip	36.0	66.1	80.5
color + blur + resize crop + flip	36.3	65.2	80.6
SLIP	42.8	72.1	82.6

Table 8. We train CLIP with different data augmentations and compare ImageNet performance to SLIP. Color + blur slightly improve performance over our improved training recipe using global image crops, but by a much smaller margin than SLIP does.

Method	0-shot	Linear	Finetuned
SLIP	42.8	72.1	82.6
Decoupled SLIP	42.7	72.0	82.8

Table 9. Decoupling self-supervision and text-supervision has no effect on performance. We sampled an additional 15M images disjoint from the YFCC15M images to use only in the self-supervised objective and observe that this performs nearly identically.

improves zero-shot ImageNet performance across model sizes by large margins of +4.8% to +5.6%. Similar gains can be seen in the linear classification setting, with consistent but marginal improvements in the end-to-end finetuning setting.

These trends remain consistent on longer training schedules with the exception of linear probe performance on SLIP ViT-L which actually decreases with more training. With SLIP ViT-L pre-trained on YFCC15M for 100 epochs, we achieve our strongest result of 47.9% zero-shot accuracy on ImageNet. SLIP also shows significant improvements on CC3M and CC12M. Finally, we also confirm our findings with zero-shot and linear evaluations on additional downstream benchmarks.

Evaluating representation quality. Prior work on representation learning has argued against end-to-end finetuning for its sensitivity to optimization hyperparameters [24], and against linear classification for being too contrived [63]. We note that zero-shot transfer, along with linear classification and end-to-end finetuning, can be viewed as one cohesive paradigm for evaluating representation quality. Zero-shot transfer represents the strictest setting, where the exemplar vector for each class must be specified through natural language. Linear classification is a relaxation of zero-shot transfer, in which the class exemplars are optimized on training data. Finally, end-to-end finetuning represents

a further relaxation of linear classification where all model parameters are allowed to adapt to training data. Representation quality should be assessed by performance across multiple settings, in the same way that an ROC curve offers a more holistic account of model performance than evaluation at a single operating point.

Zero-shot ImageNet monitor. SLIP may also serve as a useful framework within which to evaluate new methods for self-supervised learning. Training loss on the pre-text task is a poor predictor of downstream performance, so a simple external metric like kNN accuracy is important for quickly estimating performance and diagnosing training issues such as overfitting or instability. However, kNN classification requires encoding and storing every single training image and naive inference requires very expensive matrix multiplications. The memory bank kNN monitor [11] alleviates this cost but is not feasible when pre-training on unlabeled datasets such as YFCC100M. Instead, zero-shot evaluations on ImageNet are virtually as fast as evaluating validation accuracy in the supervised setting.

Ethical considerations. SLIP faces all of the same ethical considerations as CLIP, both in terms of the harmful applications it may enable, as well as the potential for amplifying and perpetuating problematic behavior in the real world. CLIP’s ability to leverage noisy and minimally filtered data scraped from the open internet has already spurred researchers to begin collecting data in a more careless manner than previously possible for supervised learning [3]. A more cautious and responsible approach to selecting training data may alleviate the most egregious model behaviors.

Practical limitations. SLIP computes embeddings of image views for both the self-supervised objective and the CLIP objective. This increases the activation count and memory footprint of the model during the forward pass, which results in slower training (30.5 hours for SLIP vs 22.3 hours for CLIP to train ViT-B/16 on 64 V100 GPUs). After pre-training, SLIP incurs no additional cost since its vision backbone can be used in the same manner as CLIP or a self-supervised model.

From the downstream results in Table 4, we note that pre-training on uncuration data alone appears to be an inefficient route to recognizing specific visual concepts, especially concepts unlikely to be widely shared on social media or the broader internet. Even with a massive amount of curated data CLIP’s zero-shot performance on many datasets is still far below what can easily be achieved by finetuning a small pre-trained model on a modest amount of labeled data. This can be easily addressed by simply finetuning CLIP for specific applications or even including more pre-training data from the domain of interest.

References

- [1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. *ArXiv*, abs/2106.08254, 2021.
- [2] Babak Ehteshami Bejnordi, Mitko Veta, Paul Johannes van Diest, Bram van Ginneken, Nico Karssemeijer, Geert J. S. Litjens, Jeroen A. van der Laak, Meyke Hermesen, Quirine F. Manson, Maschenka C. A. Balkenhol, Oscar G. F. Geessink, Nikolaos Stathonikos, Marcory Crf van Dijk, Peter Bult, Francisco Beca, Andrew H. Beck, Dayong Wang, Aditya Khosla, Rishab Gargeya, Humayun Irshad, Aoxiao Zhong, Qi Dou, Quanzheng Li, Hao Chen, Huangjing Lin, Pheng-Ann Heng, Christian Hass, Elia Bruni, Quincy Wong, Ugur Halici, Mustafa Ümit Öner, Rengul Cetin-Atalay, Matt Berseth, Vitali Khvatkov, A I Vylegzhanin, Oren Z. Kraus, Muhammad Shaban, Nasir M. Rajpoot, Ruqayya Awan, Korsuk Sirinukunwattana, Talha Qaiser, Yee-Wah Tsang, David Tellez, Jonas Annuscheit, Peter Hufnagl, Mira Valkonen, Kimmo Kartasalo, Leena Latonen, Pekka Ruusuvaari, Kaisa Liimatainen, Shadi Albarqouni, Bharti Mungal, Ami George, Stefanie Demirci, Nassir Navab, Seiryo Watanabe, Shigeto Seno, Yoichi Takenaka, Hideo Matsuda, Hady Ahmady Phoulady, Vassili A. Kovalev, Alexander Kalinovskiy, Vitali Liauchuk, Gloria Bueno, M. del Milagro Fernández-Carrobles, Ismael Serrano, Oscar Deniz, Daniel Racoceanu, and Rui Venâncio. Diagnostic assessment of deep learning algorithms for detection of lymph node metastases in women with breast cancer. *JAMA*, 318:2199–2210, 2017.
- [3] Abeba Birhane, Vinay Uday Prabhu, and Emmanuel Kahembwe. Multimodal datasets: misogyny, pornography, and malignant stereotypes. *ArXiv*, abs/2110.01963, 2021.
- [4] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *European Conference on Computer Vision*, 2014.
- [5] Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Matthijs Douze. Deep clustering for unsupervised learning of visual features. In *ECCV*, 2018.
- [6] Mathilde Caron, Hugo Touvron, Ishan Misra, Hervé Jégou, Julien Mairal, Piotr Bojanowski, and Armand Joulin. Emerging properties in self-supervised vision transformers. *ArXiv*, abs/2104.14294, 2021.
- [7] João Carreira, Eric Noland, Chloe Hillier, and Andrew Zisserman. A short note on the kinetics-700 human action dataset. *ArXiv*, abs/1907.06987, 2019.
- [8] Soravit Changpinyo, Piyush Kumar Sharma, Nan Ding, and Radu Soricut. Conceptual 12m: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 3557–3567, 2021.
- [9] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. A simple framework for contrastive learning of visual representations. *ArXiv*, abs/2002.05709, 2020.
- [10] Ting Chen, Simon Kornblith, Kevin Swersky, Mohammad Norouzi, and Geoffrey E. Hinton. Big self-supervised models are strong semi-supervised learners. *ArXiv*, abs/2006.10029, 2020.
- [11] Xinlei Chen, Saining Xie, and Kaiming He. An empirical study of training self-supervised vision transformers. *ArXiv*, abs/2104.02057, 2021.
- [12] Gong Cheng, Junwei Han, and Xiaoqiang Lu. Remote sensing image scene classification: Benchmark and state of the art. *Proceedings of the IEEE*, 105:1865–1883, 2017.
- [13] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2014.
- [14] Adam Coates, A. Ng, and Honglak Lee. An analysis of single-layer networks in unsupervised feature learning. In *AISTATS*, 2011.
- [15] Karan Desai and Justin Johnson. Virtex: Learning visual representations from textual annotations. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11157–11168, 2021.
- [16] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*, 2019.
- [17] 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. *ArXiv*, abs/2010.11929, 2021.
- [18] Li Fei-Fei, Rob Fergus, and Pietro Perona. One-shot learning of object categories. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 28:594–611, 2006.
- [19] Andrea Frome, Gregory S. Corrado, Jonathon Shlens, Samy Bengio, Jeffrey Dean, Marc’Aurelio Ranzato, and Tomas Mikolov. Devise: A deep visual-semantic embedding model. In *NIPS*, 2013.
- [20] Andreas Geiger, Philip Lenz, and Raquel Urtasun. Are we ready for autonomous driving? the kitti vision benchmark suite. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3354–3361, 2012.
- [21] Ross B. Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 580–587, 2014.
- [22] Ian J. Goodfellow, D. Erhan, Pierre Luc Carrier, Aaron C. Courville, Mehdi Mirza, Benjamin Hamner, William J. Cukierski, Yichuan Tang, David Thaler, Dong-Hyun Lee, Yingbo Zhou, Chetan Ramaiah, Fangxiang Feng, Ruifan Li, Xiaojie Wang, Dimitris Athanasakis, John Shawe-Taylor, Maxim Milakov, John Park, Radu Tudor Ionescu, Marius Claudiu Popescu, Cristian Grozea, James Bergstra, Jingjing Xie, Lukasz Romaszko, Bing Xu, Chuang Zhang, and Yoshua Bengio. Challenges in representation learning: A report on three machine learning contests. *Neural networks : the official journal of the International Neural Network Society*, 64:59–63, 2013.
- [23] Priya Goyal, Quentin Duval, Jeremy Reizenstein, Matthew Leavitt, Min Xu, Benjamin Lefaudeaux, Mannat Singh, Vinicius Reis, Mathilde Caron, Piotr Bojanowski, Armand Joulin, and Ishan Misra. Vissl. <https://github.com/facebookresearch/vissl>, 2021.

- [24] Priya Goyal, Dhruv Kumar Mahajan, Abhinav Gupta, and Ishan Misra. Scaling and benchmarking self-supervised visual representation learning. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 6390–6399, 2019.
- [25] Jean-Bastien Grill, Florian Strub, Florent Altch’e, Corentin Tallec, Pierre H. Richemond, Elena Buchatskaya, Carl Doversch, Bernardo Ávila Pires, Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Bilal Piot, Koray Kavukcuoglu, Rémi Munos, and Michal Valko. Bootstrap your own latent: A new approach to self-supervised learning. *ArXiv*, abs/2006.07733, 2020.
- [26] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Doll’ar, and Ross B. Girshick. Masked autoencoders are scalable vision learners. 2021.
- [27] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. Momentum contrast for unsupervised visual representation learning. *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 9726–9735, 2020.
- [28] Patrick Helber, Benjamin Bischke, Andreas R. Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12:2217–2226, 2019.
- [29] Gabriel Ilharco, Mitchell Wortsman, Nicholas Carlini, Rohan Taori, Achal Dave, Vaishaal Shankar, Hongseok Namkoong, John Miller, Hannaneh Hajishirzi, Ali Farhadi, and Ludwig Schmidt. Openclip, July 2021. If you use this software, please cite it as below.
- [30] Aashi Jain, Mandy Guo, Krishna Srinivasan, Ting Chen, Sneha Kudugunta, Chao Jia, Yinfei Yang, and Jason Baldridge. Mural: Multimodal, multitask retrieval across languages. *ArXiv*, abs/2109.05125, 2021.
- [31] Chao Jia, Yinfei Yang, Ye Xia, Yi-Ting Chen, Zarana Parekh, Hieu Pham, Quoc V. Le, Yun-Hsuan Sung, Zhen Li, and Tom Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *ICML*, 2021.
- [32] Justin Johnson, Bharath Hariharan, Laurens van der Maaten, Li Fei-Fei, C. Lawrence Zitnick, and Ross B. Girshick. Clevr: A diagnostic dataset for compositional language and elementary visual reasoning. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1988–1997, 2017.
- [33] Armand Joulin, Laurens van der Maaten, A. Jabri, and Nicolas Vasilache. Learning visual features from large weakly supervised data. In *ECCV*, 2016.
- [34] Douwe Kiela, Hamed Firooz, Aravind Mohan, Vedanuj Goswami, Amanpreet Singh, Pratik Ringshia, and Davide Testuggine. The hateful memes challenge: Detecting hate speech in multimodal memes. *ArXiv*, abs/2005.04790, 2020.
- [35] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *4th International IEEE Workshop on 3D Representation and Recognition (3dRR-13)*, Sydney, Australia, 2013.
- [36] Alex Krizhevsky. Learning multiple layers of features from tiny images. 2009.
- [37] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. Imagenet classification with deep convolutional neural networks. *Communications of the ACM*, 60:84 – 90, 2012.
- [38] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. 1998.
- [39] Ang Li, A. Jabri, Armand Joulin, and Laurens van der Maaten. Learning visual n-grams from web data. *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 4193–4202, 2017.
- [40] Yangguang Li, Feng Liang, Lichen Zhao, Yufeng Cui, Wanli Ouyang, Jing Shao, Fengwei Yu, and Junjie Yan. Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm. *ArXiv*, abs/2110.05208, 2021.
- [41] S. Maji, J. Kannala, E. Rahtu, M. Blaschko, and A. Vedaldi. Fine-grained visual classification of aircraft. Technical report, 2013.
- [42] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. *2008 Sixth Indian Conference on Computer Vision, Graphics & Image Processing*, pages 722–729, 2008.
- [43] Omkar M. Parkhi, Andrea Vedaldi, Andrew Zisserman, and C. V. Jawahar. Cats and dogs. *2012 IEEE Conference on Computer Vision and Pattern Recognition*, pages 3498–3505, 2012.
- [44] Ariadna Quattoni, Michael Collins, and Trevor Darrell. Learning visual representations using images with captions. *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.
- [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.
- [46] Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018.
- [47] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael S. Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115:211–252, 2015.
- [48] Mert Bulent Sariyildiz, Julien Perez, and Diane Larlus. Learning visual representations with caption annotations. In *ECCV*, 2020.
- [49] 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.
- [50] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, A. Ng, and Christopher Potts. Recursive deep models for semantic compositionality over a sentiment treebank. In *EMNLP*, 2013.
- [51] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. Ucf101: A dataset of 101 human actions classes from videos in the wild. *ArXiv*, abs/1212.0402, 2012.

- [52] Johannes Stallkamp, Marc Schlipsing, Jan Salmen, and C. Igel. The german traffic sign recognition benchmark: A multi-class classification competition. *The 2011 International Joint Conference on Neural Networks*, pages 1453–1460, 2011.
- [53] Bart Thomee, David A. Shamma, Gerald Friedland, Benjamin Elizalde, Karl S. Ni, Douglas N. Poland, Damian Borth, and Li-Jia Li. Yfcc100m: the new data in multimedia research. *Commun. ACM*, 59:64–73, 2016.
- [54] Yonglong Tian, Olivier J Henaff, and Aaron van den Oord. Divide and contrast: Self-supervised learning from uncurated data. *arXiv preprint arXiv:2105.08054*, 2021.
- [55] Antonio Torralba and Alexei A. Efros. Unbiased look at dataset bias. *CVPR 2011*, pages 1521–1528, 2011.
- [56] Hugo Touvron, Matthieu Cord, Matthijs Douze, Francisco Massa, Alexandre Sablayrolles, and Hervé Jégou. Training data-efficient image transformers & distillation through attention. In *ICML*, 2021.
- [57] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *ArXiv*, abs/1807.03748, 2018.
- [58] Bastiaan S Veeling, Jasper Linmans, Jim Winkens, Taco Cohen, and Max Welling. Rotation equivariant CNNs for digital pathology. June 2018.
- [59] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie. The Caltech-UCSD Birds-200-2011 Dataset. Technical Report CNS-TR-2011-001, California Institute of Technology, 2011.
- [60] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance-level discrimination. *ArXiv*, abs/1805.01978, 2018.
- [61] Jian xiong Xiao, James Hays, Krista A. Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 3485–3492, 2010.
- [62] Xin Yuan, Zhe L. Lin, Jason Kuen, Jianming Zhang, Yilin Wang, Michael Maire, Ajinkya Kale, and Baldo Faieta. Multimodal contrastive training for visual representation learning. *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6991–7000, 2021.
- [63] Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruysen, Carlos Riquelme, Mario Lucic, Josip Djolonga, André Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, Lucas Beyer, Olivier Bachem, Michael Tschannen, Marcin Michalski, Olivier Bousquet, Sylvain Gelly, and Neil Houlsby. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv: Computer Vision and Pattern Recognition*, 2019.

A. Additional Implementation Details

Datasets. YFCC15M [46, 53] contains raw HTML captions and titles which we lightly preprocess before training. We unescape the HTML then remove HTML tags and urls with simple regex matching.

CC3M [49] is collected from an initial set of 5B candidate images, of which 99.9% are filtered out according to simple image and text heuristics for quality and content. Many of these filters are relaxed by CC12M [8] in order to collect a bigger and potentially noisier dataset. CC3M also hypernymizes proper nouns, numbers, and infrequent entities to make the dataset more amenable to training and evaluating image captioning systems, the original design for the dataset. In contrast, CC12M only replaces person names for privacy. Our versions of these datasets contain 3.1M and 11.0 M images respectively, due to asset removal.

Pre-training. During pre-training we use a cosine learning rate decay schedule with 1 epoch (~ 3500 iterations) of linear warmup when training on YFCC15M. When pre-training for 100 epochs we use 2 warmup epochs. On YFCC15M (14.6M images), we train for 25 epochs and on CC12M (11.0M images) we train for 35 epochs. This amounts to approximately the same number of iterations as 300 epochs on ImageNet-1K [47]. Due to the smaller size of CC3M (3.1M images), we train for 40 epochs to reduce overfitting. We trained on up to sixteen $8 \times$ V100-32GB servers, and to fit SLIP ViT-Large/16 in memory we accumulated gradients over two steps.

End-to-end Finetuning. We use a similar training recipe for finetuning all models on ImageNet based on the ImageNet finetuning recipe from BeiT [1] using AdamW and a batch size of 1024 with learning rate of $4e-3$ and weight decay of 0.05, along with various data augmentations and regularization methods. As we increase model size we also increase regularization. For ViT-S we set drop path to 0 and layer decay to 0.65, for ViT-B we set drop path to 0.1 and layer decay to 0.65, and for ViT-L we set drop path to 0.

B. Full Scaling Results

We include the full results of our scaling experiments in Table 10, in which we simultaneously increase model size and training epochs. As measured by ImageNet classification accuracy under the three settings (zero-shot transfer, linear classification, and end-to-end finetuning), both large models and longer training generally improve performance.

The exception to this trend is the linear classification performance of SLIP ViT-L/16, which degrades slightly with longer training. This behavior also persists across the various other downstream benchmarks, where SLIP ViT-L/16 does worse on average when trained for 100 epochs than when trained for 25 epochs. We note that both the zero-shot

transfer and end-to-end finetuning performance of SLIP ViT-L/16 improve with longer training, contrary to the behavior seen with linear classification. Thus we cannot declare this behavior to be a case of simple overfitting, as the representations are still improved for the other evaluation settings.

C. Additional Linear Classification Benchmarks

In Table 11 we show linear classification results on all 26 downstream datasets (including ImageNet). With ViT-B and ViT-S, SLIP pre-training for 100 epochs does best. As with ImageNet, SLIP ViT-L also does worse on average when trained for 100 epochs than when trained for 25 epochs. The dataset average is 0.5 points lower for the 100 epoch model.

As expected, linear classification accuracy is much higher than zero-shot transfer accuracy (shown in Table 4). However, the gap between zero-shot and linear performance varies between datasets. On datasets which are straightforward vision tasks but poorly represented among the YFCC100M imagery, such as Patch Camelyon, MNIST, KITTI distance, and GTSRB, linear classification massively improves accuracy, often from a baseline of around chance performance. On datasets which share more overlap with YFCC100M, such as Food-101, Caltech-101, and Caltech-UCSD Birds 2011, we see significant improvements as well.

However, with HatefulMememes and Rendered SST2, two datasets which require OCR capabilities, the linear classification performance of all models is still around chance. These results suggest, perhaps unsurprisingly, that zero-shot transfer results are much more dependent on what visual and semantic concepts were seen during training than linear classification, since they do not enjoy the benefit of further training examples. We also note that relative rankings within each model size are also quite unstable where the best results alternate between the 25 and 100 epoch models. This is very similar to what we see in the zero-shot transfer evaluations, as discussed in Section 5.

Model	0-shot			Linear			Finetuned		
	25	50	100	25	50	100	25	50	100
ViT-S/16	38.3	39.3	39.5	66.4	67.6	68.3	80.3	80.7	80.7
ViT-B/16	42.8	44.1	45.0	72.1	73.0	73.6	82.6	82.9	83.4
ViT-L/16	46.2	47.4	47.9	76.0	75.8	75.1	84.2	84.7	84.8

Table 10. Full scaling experiment results. SLIP scales well to larger models and longer training as measured by zero-shot transfer, linear classification, and end-to-end finetuning, with the exception of linear classification performance using ViT-L.

		Food-101	CIFAR-10	CIFAR-100	CUB	SUN397	Cars	Aircraft	DTD	Pets	Caltech-101	Flowers	MNIST	FER-2013	STL-10	EuroSAT	RESISC45	GTSRB	KITTI	Country211	PCAM	UCF101	Kinetics700	CLEVR	HatefulMemes	SST2	ImageNet	Average
ViT-S	CLIP	71.1	82.5	63.2	66.5	70.7	28.2	26.0	61.6	64.1	82.5	91.8	89.6	45.2	92.1	92.4	83.6	65.7	63.6	22.0	80.4	67.7	31.8	44.3	53.0	51.0	59.3	63.5
	SLIP (25 ep)	77.4	80.7	63.5	67.0	74.2	39.3	33.8	70.7	69.5	87.2	94.5	91.3	52.7	95.7	94.3	90.5	68.2	65.8	22.4	81.9	76.9	39.2	51.6	55.6	54.3	66.4	67.9
	SLIP (100 ep)	78.7	84.1	66.3	66.0	73.9	40.6	32.6	71.6	72.0	87.9	95.3	90.7	50.4	96.4	95.2	89.0	68.2	66.8	23.3	82.9	77.7	40.5	50.5	53.4	53.4	68.3	68.3
ViT-B	CLIP	77.6	86.2	70.7	70.9	73.7	41.8	32.0	66.0	70.5	85.3	94.2	93.8	49.7	94.9	94.5	88.3	72.5	65.8	24.9	82.9	72.6	36.4	48.7	53.4	54.8	66.5	68.0
	SLIP (25 ep)	83.0	87.7	71.6	70.9	76.3	47.4	36.8	73.9	74.4	89.7	96.4	94.5	53.5	97.4	95.9	92.8	75.5	68.6	25.1	84.4	80.4	43.7	54.2	57.8	55.0	72.1	71.5
	SLIP (100 ep)	83.1	88.9	71.5	72.0	76.4	49.0	37.0	75.9	75.8	90.7	96.2	94.5	54.4	98.2	95.5	92.0	75.9	67.9	25.6	83.0	82.0	44.4	53.9	55.2	56.1	73.6	71.9
ViT-L	CLIP	81.8	91.2	75.1	75.1	75.4	46.9	34.7	66.2	73.0	86.4	95.4	95.7	54.7	96.5	95.1	90.6	76.4	68.7	27.2	83.6	75.9	39.9	51.9	57.0	53.8	70.5	70.7
	SLIP (25 ep)	86.5	92.9	77.2	76.6	78.0	52.0	38.8	75.2	79.4	90.7	97.7	95.8	56.8	98.7	96.6	93.3	79.9	68.9	29.1	84.8	83.0	47.7	57.5	52.6	54.5	76.0	73.9
	SLIP (100 ep)	84.1	91.0	74.5	72.5	76.7	51.1	38.4	77.0	79.8	91.4	97.2	95.8	60.2	98.7	97.0	93.2	75.6	70.6	28.2	85.2	83.1	45.8	55.0	54.0	56.2	75.1	73.4

Table 11. Linear classification evaluation with ViT S, B, and L on a variety of classification benchmarks. Best results in **bold**. SLIP outperforms CLIP on most of the tasks, frequently with a significant margin.

Dataset	Metric	Chance performance	Description
Food-101 [4]	acc	1.0	101 categories of food dishes
CIFAR-10 [36]	acc	10.0	10 categories of animals and vehicles
CIFAR-100 [36]	acc	1.0	100 categories of animals, vehicles, plants, objects, scenes, people
CUB-200-2011 [59]	acc	0.8	200 species of mostly North American birds
SUN397 [61]	acc	2.2	397 categories of various indoor and outdoor scenes
Stanford Cars [35]	acc	0.8	196 categories of cars (make, model, and year)
FGVC Aircraft [41]	mean per class	1.0	102 categories of aircraft (manufacturer, family, and variant)
Describable Textures [13]	acc	2.1	47 categories of texture patches
Oxford Pets [43]	mean per class	2.7	37 breeds of cats and dogs
Caltech-101 [18]	mean per class	5.2	101 categories of objects
Oxford Flowers [42]	mean per class	1.5	102 species of common UK flowers
MNIST [38]	acc	10.0	10 categories of handwritten digits
FER-2013 [22]	acc	24.7	7 categories of human facial emotions
STL-10 [14]	acc	11.4	10 categories of animals and vehicles
EuroSat [28]	acc	10.0	10 categories of land from satellite imagery
RESISC45 [12]	acc	2.2	45 categories of land from satellite imagery and aerial photography
GTSRB [52]	acc	5.9	43 categories of German traffic signs
KITTI Distance [20]	acc	31.0	4 categories of traffic scenes with nearby cars in varying positions
Country211 [45, 53]	acc	0.5	211 countries represented by geo-tagged images
Patch Camelyon [58, 2]	acc	50.0	2 classes of metastatic or benign lymph node slide patches
UCF101 Frames [51]	acc	1.3	101 categories of human actions using the middle frame of each clip
Kinetics 700 Frames [7]	mean(acc1, acc5)	0.4	700 categories of human actions using the middle frame of each clip
Clevr Counts [32]	acc	12.9	8 categories of rendered scenes with varying numbers of objects
Hateful Memes [34]	ROC AUC	50.0	2 categories of hateful or not hateful image macros
Rendered SST2 [45, 50]	acc	50.1	2 classes of positive or negative movie reviews rendered as text
ImageNet [47]	acc	0.1	1000 categories of objects

Table 12. Info sheet for classification datasets. Chance performance is computed by assuming random predictions of the labels in proportion to their frequency in the test set.