CLIP meets Model Zoo Experts: Pseudo-Supervision for Visual Enhancement

Anonymous Author(s)
Affiliation
Address
`email`

Abstract

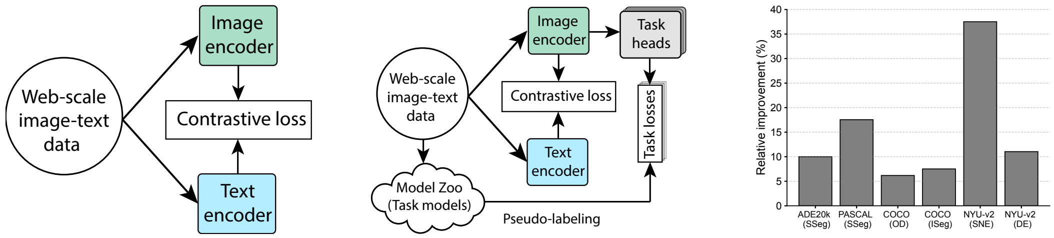
Contrastive language image pretraining (CLIP) is a standard method for training vision-language models. While CLIP is scalable, promptable, and robust to distribution shifts on image classification tasks, it lacks object localization capabilities. This paper studies the following question: *Can we augment CLIP training with task-specific vision models from model zoos to improve its visual representations?* Towards this end, we leverage open-source task-specific vision models to generate pseudo-labels for an uncurated web-scale image-text dataset. Subsequently, we train CLIP models on these pseudo-labels in addition to the contrastive training on image and text pairs. This simple setup shows substantial improvements of up to 16.3% across different vision tasks, including segmentation, detection, depth estimation, and surface normal estimation. Importantly, these enhancements are achieved without compromising CLIP's existing capabilities, including its proficiency in promptable zero-shot classification.

1 Introduction

Foundation Models (FMs) are revolutionizing different domains of artificial intelligence and machine learning, including computer vision [31, 14, 17] and natural language processing [7, 2, 41]. FMs can be trained on web crawled data without relying on crowd or expert annotations, and yet they demonstrate strong generalization capabilities [15, 36].

CLIP, one of the most prominent methods for FM training in vision, uses contrastive learning to align image and text representations [31, 15]. In addition to robustness to data distribution shifts, CLIP offers impressive zero-shot and cross-modal retrieval capabilities on unseen datasets. Nevertheless, computer vision encompasses a broad range of tasks that require the ability to comprehend spatial relationships, semantic content, object localization, and 3D structures. In spite of CLIP's impressive zero-shot open-vocabulary classification accuracy, it exhibits poor localization capabilities and often struggles in associating text with objects in an image [40, 12, 32]. Consequently, in practice, many vision tasks (e.g., detection and segmentation), rely on CLIP through fine-tuning the entire model to compensate for these localization deficiencies.

In this work, we seek to answer the following question: *Can we augment pretrained CLIP models with task-specific vision models from model zoos to improve its visual representations?* That is, we seek to (1) use open-source task-specific vision models to generate *hard* pseudo-labels on a web-scale noisy image-text dataset and, (2) train CLIP on image-text pairs along with pseudo-labels with multiple objectives. An overview of our approach, which we call **CLIP T**raining with **eX**perts (`CLIPTeX`), is shown in Fig. 1. We show that `CLIPTeX` enhances the visual representations of CLIP and yields up to 16.3% enhancement in probing accuracy across a diverse set of vision tasks and datasets while preserving the existing capabilities of CLIP models, including prompting for zero-shot classification.



(a) (b) (c)

Figure 1: **Training CLIP with pseudo-labels improves its visual representations.** (a) shows the standard CLIP training. (b) shows `CLIPTeX` that trains CLIP with pseudo-labels from experts. Note that the main purpose of task heads is to improve CLIP's image encoder with expert knowledge, and the heads can be discarded after training. (c) shows the relative improvement that `CLIPTeX` obtains over CLIP-FT. Here, SSeg, OD, ISeg, SNE, and DE refer to semantic segmentation, object detection, instance segmentation, surface normal estimation, and depth estimation respectively.

2 CLIPTeX

Model `CLIPTeX` extends CLIP with pseudo-supervision from publicly available task experts specializing in localization, depth, and surface normal estimation. Our approach enhances CLIP's representations *without any labeled data collection* (Fig. 1). Similar to CLIP, `CLIPTeX` uses two encoders: (1) an image encoder that takes an RGB image and produces an image embedding and (2) a text encoder that takes the text caption and produces a text embedding.

In addition to contrastive training, we would like to train `CLIPTeX` using pseudo-labels. Towards that end, we incorporate task-specific heads that take the output of image encoder as input and generate predictions for the respective task (see Fig. 1b). Previous work have shown that multi-scale representations provides significant benefit in tasks requiring localization and fine-grained visual understanding [49, 22]. However, some image encoders (e.g., ViT) do not inherently possess these capabilities. To ensure `CLIPTeX` can learn better visual representations independent of the image backbone, we include a single shared multi-scale module [49] between image encoder and task-specific heads. We feed the output of the image encoder through a multi-scale module [49], which in turn feeds into the lightweight task-specific classification or regression heads. In our implementation, we use independent point-wise convolution as the head for each task. As the task's head output dimensions should match input dimensions in dense prediction tasks, we perform nearest neighbour interpolation on head's output if necessary.

Training objective To train `CLIPTeX` with pseudo-supervision on n tasks, we generate *hard* pseudo-labels offline using publicly available task-specific experts on an uncurated web-scale dataset. We then train `CLIPTeX` with a weighted sum of contrastive loss and task-specific losses: $\mathcal{L} = \lambda_{\text{clip}} \cdot \mathcal{L}_{\text{clip}} + \sum_{t=1}^{n} \lambda_{\text{task}}^{t} \cdot \mathcal{L}_{\text{task}}^{t}$ where $\mathcal{L}_{\text{task}}^{t}$ is the loss of the t-th task and $\mathcal{L}_{\text{clip}}$ is the contrastive loss. Here, $\lambda_{\text{task}}^{t}$ and λ_{clip} are the loss coefficients of t-th task and the standard CLIP loss, respectively.

3 Experimental Setup

Probing, a standard method to study the representations learnt by neural networks [14, 31], is used to investigate whether pseudo-supervision in `CLIPTeX` can improve CLIP's image backbone.

Task-specific experts We train `CLIPTeX` with hard pseudo-labels generated from following experts: (1) *Semantic segmentation.* We use Mask-RCNN [13] with ViT backbone [8], trained on the COCO [21] with RangeAugment [27], to produce pseudo-labels for segmentation. (2) *Monocular depth estimation:* We use DPT [33], trained on MIX-6 dataset [33], to generate monocular depth map pseudo-labels. (3) *Surface normal estimation:* We use *NLL-AngMF* [1] as our surface normal expert, which is trained on ScanNet dataset [6].

Baselines We compare with following baselines: (1) *CLIP.* We use CLIP model [26] pretrained on 1.2 billion images with a variable resolution and batch sampler whose base input image's spatial

Table 1: **Probing results for different vision tasks.** Pseudo-labeling in `CLIPTeX` significantly improves the visual representations in the image encoder of CLIP.

Model	**Segmentation(↑)**		**Detection(↑)**		**Depth(↓)**		**Surface normal (↑)**		**Classification(↑)**
	Linear	**PSPNet**	**Mask-RCNN**	**SSD**	**Linear**	**PSPNet**	**Linear**	**PSPNet**	**Linear**
ViT-B/16									
CLIP	18.66	45.53	15.20	5.33	0.235	0.168	28.49	47.29	**80.24**
CLIP-FT	62.47	78.22	27.21	16.46	0.215	0.139	29.06	47.91	79.94
`CLIPTeX` **(Ours)**	**73.43**	**80.71**	**28.89**	**17.50**	**0.159**	**0.128**	**39.96**	**50.80**	79.64
ViT-H/16									
CLIP	56.18	75.37	26.65	11.07	0.212	0.132	29.09	49.78	**84.85**
CLIP-FT	62.95	82.94	33.93	20.24	0.213	0.125	29.21	50.48	84.1
`CLIPTeX` **(Ours)**	**79.30**	**84.31**	**34.50**	**21.55**	**0.138**	**0.117**	**43.22**	**53.89**	83.2
ResNet-50									
CLIP	**46.96**	70.92	29.49	20.32	**0.212**	**0.147**	**33.67**	47.28	78.35
CLIP-FT	34.78	74.17	38.13	**30.28**	0.239	0.155	28.72	48.66	78.92
`CLIPTeX` **(Ours)**	40.31	**75.58**	**38.23**	28.62	0.220	0.150	31.56	**49.44**	**78.95**

resolution is 224×224. (2) *CLIP-FT.* Many dense prediction tasks (e.g., segmentation) benefit from using high-resolution input images. To have a fairer baseline trained on the same resolution as `CLIPTeX` , we finetune CLIP with contrastive loss on CC3M. The training is done with variable resolution using a batch sampler whose base input image resolution is 512×512. Any improvements over this baseline signify a pure transfer of knowledge from pseudo-supervision.

To show the generality of `CLIPTeX` , we experiment with three image encoder backbones: ViT-B/16, ViT-H/16, and ResNet-50. Also note that we finetune `CLIPTeX` on CC3M's image and text pairs along with pseudo-labels using the same settings as CLIP-FT. We use cross-entropy loss to train on segmentation pseudo-labels, and L1 loss to train on depth and surface normal pseudo-labels.

Evaluation downstream tasks and datasets We evaluate the models using classifier and regressor probes on the following tasks: (1) *Semantic segmentation.* We use PASCAL VOC [9] with 20 classes. We report mean intersection over union (mIoU) on the validation set. (2) *Object detection and instance segmentation.* The models are evaluated on COCO dataset for detection and instance segmentation. Importantly, during training with pseudo-labels, we do not use the bounding boxes. Instead, the instance masks are converted to semantic segmentation pseudo-labels. This allows us to evaluate baselines on both instance segmentation and object detection, which are considered to be more challenging tasks than semantic segmentation. Following standard convention, we evaluate the accuracy on COCO's validation set in terms of mean average precision (mAP). (3) *Monocular depth estimation.* We use NYU-V2 [29] dataset as our depth estimation benchmark. Note that DPT, the expert used for depth pseudo-supervision, is trained on a different dataset, i.e., ScanNet. We use absolute relative error as a metric for evaluation on the validation set. (4) *Surface normal estimation.* We use NYU-V2 for surface normal estimation. We train on the training set used by Bae et al. [1] and Qi et al. [30], and evaluate on the official test set of NYU-V2. Following [1], we use a<30 as the metric for evaluation. (5) *Image classification.* We evaluate on ImageNet 1K [35] classification dataset and top-1 accuracy on the validation set is reported as the evaluation metric.

Classifier and regressor probes for evaluation To study the visual representations of different *frozen* pre-trained models, our experiments involve both classification and regression tasks across different datasets. For dense prediction tasks, such as semantic segmentation, depth, and surface normal estimation we probe frozen image encoders with two types of probes: (1) Linear which is a point-wise convolutional layer. (2) PSPNet [49], a standard non-linear head for dense prediction tasks. For image classification a fully-connected layer is used as the linear probe. For object detection and instance segmentation, Mask R-CNN [13] and SSD heads are used. Additional probing results with different heads (e.g. DeepLabV3) and tasks (e.g. ADE20k) are included in Appendix C.

4 Results

Pseudo-supervision improves visual representations Probing results for all tasks are given in Table 1. In semantic segmentation, `CLIPTeX` shows consistent improvements over the baselines. Particularly noteworthy is the linear probing accuracy of `CLIPTeX` with ViT-B/16 and ViT-H/16 backbones on the PASCAL VOC dataset, which is about 10% and 16.3% better than CLIP-FT.

Table 2: **CLIP's zero-shot knowledge is preserved after training with experts.** (a) report zero-shot top-1 accuracy for ImageNet-1k dataset and (b) reports recall@1/5/10 for Flickr-30k dataset.

(a) 0-shot classification on ImageNet.

Model	0-shot Top-1
CLIP-FT	68.76
CLIPTeX **(Ours)**	68.25

(b) 0-shot retrieval on Flickr-30k.

Model	Text Retrieval			Image Retrieval		
	R@1	R@5	R@10	R@1	R@5	R@10
CLIP-FT	85.90	96.70	98.60	71.66	91.00	94.94
CLIPTeX **(Ours)**	86.00	96.90	98.70	71.40	90.86	95.16

Table 3: **Role of pseudo-labels from each experts in CLIPTeX training.**

Row	Expert			Segmentation (↑)		Detection (↑)		Depth (↓)		Surface Normal (↑)	
#	Segmentation	Depth	Surface Normal	Linear	PSPNet	Mask R-CNN	SSD	Linear	PSPNet	Linear	PSPNet
R1	✗	✗	✗	62.47	78.22	27.21	16.46	0.215	0.139	29.06	47.91
R2	✓	✗	✗	72.21	81.39	28.54	17.58	0.203	0.136	34.86	48.62
R3	✗	✓	✗	64.50	81.16	27.75	16.70	0.170	0.131	35.21	49.51
R4	✗	✗	✓	63.28	81.48	27.69	16.81	0.193	0.134	37.42	50.71
R5	✓	✓	✗	73.96	81.49	28.83	17.57	0.162	0.130	37.05	49.69
R6	✓	✗	✓	72.67	81.30	28.83	17.75	0.188	0.132	38.65	50.48
R7	✗	✓	✓	64.20	81.17	27.90	17.00	0.165	0.129	39.59	51.01
R8	✓	✓	✓	73.43	80.71	28.89	17.50	0.159	0.128	39.96	50.49

For object detection with ViT-B/16 as the frozen backbone and Mask-RCNN as the probing head, CLIPTeX delivers 13.69% and 1.68% better bounding box mAP over CLIP and CLIP-FT respectively. We observe similar gains when CLIPTeX is probed with SSD.

For depth estimation, CLIPTeX obtains lower error rate, while for surface normal estimation, CLIPTeX obtains higher value of a<30 compared to CLIP and CLIP-FT baselines. These results indicate a positive transfer of distance and surface orientation knowledge to CLIPTeX 's image backbone, contributing to the better performance.

Unlike other dense prediction tasks, CLIP achieves similar or slightly better accuracy compared to CLIP-FT and CLIPTeX . This outcome can be attributed to the characteristics of image classification tasks as it primarily focuses on recognizing objects without requiring detailed information about spatial relationships or 3D structure of the scene.

Zero-shot capabilities are preserved in CLIPTeX One of the important and powerful characteristics of CLIP is *prompting*, which enables zero-shot transfer to new datasets. Pseudo-supervision with experts can potentially lead to catastrophic forgetting of previously learned knowledge, which may in turn affect model's zero-shot generalization capabilities. Table 2 compares the zero-shot capabilities of CLIP-FT and CLIPTeX in classification on ImageNet-1k [35] and retrieval on Flickr-30k [45] tasks respectively. CLIPTeX 's zero-shot performance is on par with that of CLIP-FT, indicating that enhanced representations do not result in catastrophic forgetting.

Ablation on the importance of pseudo-labels in CLIPTeX . Incorporating pseudo-supervision from task-specific experts, even from a single expert during training, results in substantial improvements in performance. These improvements are observed when evaluating models on various downstream tasks with different probes (see R1 vs. rest; Table 3). Overall, our findings indicate that incorporating knowledge from all experts contributes to learning better visual representations. Therefore, we use all experts for pseudo-supervision while training CLIPTeX .

5 Conclusion

As the field of machine learning research embraces openness, a growing number of specialized expert models become publicly available. Our study showcased the potential of leveraging these publicly available expert models to enhance CLIP's visual representations, all without the necessity of collecting task-specific data. Our experiments revealed that CLIPTeX yields improvements across a wide range of tasks, highlighting its versatility and effectiveness.

References

[1] G. Bae, I. Budvytis, and R. Cipolla. Estimating and exploiting the aleatoric uncertainty in surface normal estimation. *CoRR*, abs/2109.09881, 2021. URL `https://arxiv.org/abs/2109.09881`.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

[3] M. Caron, H. Touvron, I. Misra, H. Jégou, J. Mairal, P. Bojanowski, and A. Joulin. Emerging properties in self-supervised vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 9650–9660, 2021.

[4] R. Caruana. Multitask learning. *Machine learning*, 28:41–75, 1997.

[5] T. Chen, S. Saxena, L. Li, T.-Y. Lin, D. J. Fleet, and G. Hinton. A unified sequence interface for vision tasks, 2022.

[6] A. Dai, A. X. Chang, M. Savva, M. Halber, T. A. Funkhouser, and M. Nießner. Scannet: Richly-annotated 3d reconstructions of indoor scenes. *CoRR*, abs/1702.04405, 2017. URL `http://arxiv.org/abs/1702.04405`.

[7] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

[8] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, J. Uszkoreit, and N. Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. *CoRR*, abs/2010.11929, 2020. URL `https://arxiv.org/abs/2010.11929`.

[9] M. Everingham, L. V. Gool, C. K. I. Williams, J. M. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88:303–338, 2010. URL `https://api.semanticscholar.org/CorpusID:4246903`.

[10] S. Y. Gadre, G. Ilharco, A. Fang, J. Hayase, G. Smyrnis, T. Nguyen, R. Marten, M. Wortsman, D. Ghosh, J. Zhang, et al. Datacomp: In search of the next generation of multimodal datasets. *arXiv preprint arXiv:2304.14108*, 2023.

[11] G. Ghiasi, B. Zoph, E. D. Cubuk, Q. V. Le, and T.-Y. Lin. Multi-task self-training for learning general representations, 2021.

[12] G. Ghiasi, X. Gu, Y. Cui, and T.-Y. Lin. Scaling open-vocabulary image segmentation with image-level labels, 2022.

[13] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick. Mask R-CNN. *CoRR*, abs/1703.06870, 2017. URL `http://arxiv.org/abs/1703.06870`.

[14] K. He, X. Chen, S. Xie, Y. Li, P. Dollár, and R. Girshick. Masked autoencoders are scalable vision learners. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 16000–16009, 2022.

[15] C. Jia, Y. Yang, Y. Xia, Y.-T. Chen, Z. Parekh, H. Pham, Q. Le, Y.-H. Sung, Z. Li, and T. Duerig. Scaling up visual and vision-language representation learning with noisy text supervision. In *International conference on machine learning*, pages 4904–4916. PMLR, 2021.

[16] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, P. Dollár, and R. Girshick. Segment anything, 2023.

[17] A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, T. Xiao, S. Whitehead, A. C. Berg, W.-Y. Lo, et al. Segment anything. *arXiv preprint arXiv:2304.02643*, 2023.

[18] K. Lasinger, R. Ranftl, K. Schindler, and V. Koltun. Towards robust monocular depth estimation: Mixing datasets for zero-shot cross-dataset transfer. *CoRR*, abs/1907.01341, 2019. URL `http://arxiv.org/abs/1907.01341`.

[19] D.-H. Lee et al. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *Workshop on challenges in representation learning, ICML*, volume 3, page 896. Atlanta, 2013.

[20] L. H. Li, P. Zhang, H. Zhang, J. Yang, C. Li, Y. Zhong, L. Wang, L. Yuan, L. Zhang, J.-N. Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10965–10975, 2022.

[21] T. Lin, M. Maire, S. J. Belongie, L. D. Bourdev, R. B. Girshick, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft COCO: common objects in context. *CoRR*, abs/1405.0312, 2014. URL `http://arxiv.org/abs/1405.0312`.

[22] T.-Y. Lin, P. Dollár, R. Girshick, K. He, B. Hariharan, and S. Belongie. Feature pyramid networks for object detection, 2017.

[23] S. Liu, E. Johns, and A. J. Davison. End-to-end multi-task learning with attention, 2019.

[24] S. Liu, L. Fan, E. Johns, Z. Yu, C. Xiao, and A. Anandkumar. Prismer: A vision-language model with an ensemble of experts, 2023.

[25] J. Lu, C. Clark, R. Zellers, R. Mottaghi, and A. Kembhavi. Unified-io: A unified model for vision, language, and multi-modal tasks, 2022.

[26] S. Mehta, S. Naderiparizi, F. Faghri, M. Horton, L. Chen, A. Farhadi, O. Tuzel, and M. Rastegari. Rangeaugment: Efficient online augmentation with range learning. *arXiv preprint arXiv:2212.10553*, 2022.

[27] S. Mehta, S. Naderiparizi, F. Faghri, M. Horton, L. Chen, A. Farhadi, O. Tuzel, and M. Rastegari. Rangeaugment: Efficient online augmentation with range learning, 2022.

[28] I. Misra, A. Shrivastava, A. Gupta, and M. Hebert. Cross-stitch networks for multi-task learning, 2016.

[29] P. K. Nathan Silberman, Derek Hoiem and R. Fergus. Indoor segmentation and support inference from rgbd images. In *ECCV*, 2012.

[30] X. Qi, R. Liao, Z. Liu, R. Urtasun, and J. Jia. Geonet: Geometric neural network for joint depth and surface normal estimation. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 283–291, 2018.

[31] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever. Learning transferable visual models from natural language supervision. *CoRR*, abs/2103.00020, 2021. URL `https://arxiv.org/abs/2103.00020`.

[32] K. Ranasinghe, B. McKinzie, S. Ravi, Y. Yang, A. Toshev, and J. Shlens. Perceptual grouping in contrastive vision-language models. ICCV, 2023.

[33] R. Ranftl, A. Bochkovskiy, and V. Koltun. Vision transformers for dense prediction. *CoRR*, abs/2103.13413, 2021. URL `https://arxiv.org/abs/2103.13413`.

[34] S. Ruder. An overview of multi-task learning in deep neural networks. *arXiv preprint arXiv:1706.05098*, 2017.

[35] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. S. Bernstein, A. C. Berg, and L. Fei-Fei. Imagenet large scale visual recognition challenge. *CoRR*, abs/1409.0575, 2014. URL `http://arxiv.org/abs/1409.0575`.

[36] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman, et al. Laion-5b: An open large-scale dataset for training next generation image-text models. *Advances in Neural Information Processing Systems*, 35: 25278–25294, 2022.

[37] M. Shukor, C. Dancette, A. Rame, and M. Cord. Unified model for image, video, audio and language tasks, 2023.

[38] T. Sun, M. Segu, J. Postels, Y. Wang, L. Van Gool, B. Schiele, F. Tombari, and F. Yu. Shift: a synthetic driving dataset for continuous multi-task domain adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 21371–21382, 2022.

[39] X. Sun, R. Panda, R. Feris, and K. Saenko. Adashare: Learning what to share for efficient deep multi-task learning, 2020.

[40] T. Thrush, R. Jiang, M. Bartolo, A. Singh, A. Williams, D. Kiela, and C. Ross. Winoground: Probing vision and language models for visio-linguistic compositionality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5238–5248, 2022.

[41] H. Touvron, T. Lavril, G. Izacard, X. Martinet, M.-A. Lachaux, T. Lacroix, B. Rozière, N. Goyal, E. Hambro, F. Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[42] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin. Attention is all you need, 2023.

[43] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang. Ofa: Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework, 2022.

[44] Z. Yang, Z. Gan, J. Wang, X. Hu, F. Ahmed, Z. Liu, Y. Lu, and L. Wang. Crossing the format boundary of text and boxes: Towards unified vision-language modeling. *CoRR*, abs/2111.12085, 2021. URL `https://arxiv.org/abs/2111.12085`.

[45] P. Young, A. Lai, M. Hodosh, and J. Hockenmaier. From image descriptions to visual denotations: New similarity metrics for semantic inference over event descriptions. *Transactions of the Association for Computational Linguistics*, 2:67–78, 2014. doi: 10.1162/tacl_a_00166. URL `https://aclanthology.org/Q14-1006`.

[46] J. Yu, Z. Wang, V. Vasudevan, L. Yeung, M. Seyedhosseini, and Y. Wu. Coca: Contrastive captioners are image-text foundation models, 2022.

[47] H. Zhang, P. Zhang, X. Hu, Y.-C. Chen, L. H. Li, X. Dai, L. Wang, L. Yuan, J.-N. Hwang, and J. Gao. Glipv2: Unifying localization and vision-language understanding, 2022.

[48] Y. Zhang, K. Gong, K. Zhang, H. Li, Y. Qiao, W. Ouyang, and X. Yue. Meta-transformer: A unified framework for multimodal learning, 2023.

[49] H. Zhao, J. Shi, X. Qi, X. Wang, and J. Jia. Pyramid scene parsing network, 2017.

A Related Work

Vision FMs. Vision FMs extended the concept of pre-training to vast datasets containing hundreds of millions or even billions of images. This was in part driven by the introduction of ViTs [8] which demonstrated the scalability of training Transformers [42] to such large-scale datasets in the field of computer vision. Since then, numerous large-scale pre-training methods have emerged in the domain of computer vision [e.g., 31, 46, 3, 14]. Arguably, one of the most prominent classes of vision FMs is CLIP that specializes in aligning noisy image-text pairs from the web [31, 36, 10]. This distinction is not only attributed to its scalability, but also to its prompting capabilities and robustness in handling dataset distribution shifts. Nevertheless, these models often face challenges in associating text with individual objects and localizing them [40, 12, 32]. This work focuses on enhancing this capability through pseudo-supervision.

Pseudo-supervision with experts. The primary objective of pseudo-supervision [19] is to facilitate model training by generating pseudo-labels for unlabeled data, typically leveraging experts trained on a subset of the data containing ground truth labels. This methodology has also been applied to the training of foundation models (FMs). To the best of our knowledge, current approaches involve the acquisition of crowd labels for a portion of the data on a *single* task, with the subsequent training of experts on this labeled subset [e.g., 11, 47, 16, 24]. These trained experts are then utilized to create pseudo-labels for the remaining unlabeled data. Essentially, these methods employ experts that have been trained on the same or similar data distribution as the unlabeled data, aiming to achieve positive transfer. For example, in GLIP [20], a subset of web data is crowd-sourced to obtain localization labels, which is then used for expert training. Following expert training, these experts are employed to generate pseudo-labels for the remaining unlabeled web data. This combination of crowd labels and pseudo-labels is subsequently used to train the GLIP V2 [47] model. SAM [16] also follows similar paradigm for creating large-scale segmentation dataset. Unlike previous approaches, our proposed method uses publicly accessible experts trained on diverse tasks with different data distributions and objectives.

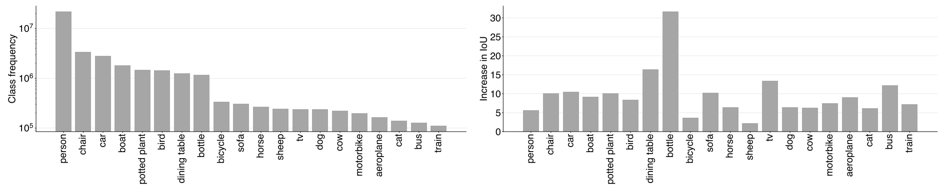
Multi-task learning for FMs. Multi-tasking [4, 34], a standard method for training on multiple tasks simultaneously, is widely used in machine learning [23, 28, 39], including FMs [e.g., 43, 5, 44, 37, 48]. Existing multi-task FMs creates a unified multi-task datasets by either collecting a new labeled dataset [e.g., 38] or mixing existing labeled datasets [e.g., 25], to facilitate positive transfer of knowledge to down-stream tasks. In contrast, `CLIPTeX` does not need any data collection and uses pseudo-supervision for training.

A.1 Positive Transfer of Representations from `CLIPTeX` to Downstream Tasks

The CC3M dataset is uncurated and noisy, and may have a skewed distribution towards specific object classes or scenes. Consequently, knowledge transfer from experts to `CLIPTeX` may also be skewed towards more frequent objects in the data. To explore this phenomenon, we quantified the frequency of objects (bounding boxes or instances) in the pseudo-labels generated by the Mask R-CNN expert (Fig. 2a) on the CC3M dataset. Additionally, we examined class-wise improvements in IoU of `CLIPTeX` with respect to CLIP-FT on the PASCAL VOC dataset (Fig. 2b). `CLIPTeX` improves the IoU for all classes in the PASCAL VOC dataset and is not biased towards the most frequently occurring object classes. These findings, combined with insights in Section 4 suggests positive transfer of representations from `CLIPTeX` to down-stream tasks.

B Task-head complexity

As discussed in Section 2, we use light-weight heads to improve visual representations in CLIP's image encoder. We replace these heads with heavier counterparts (comprising of three standard convolutional layers) when training `CLIPTeX` with CC3M pseudo-labels. Table 4 shows that light-weight heads deliver similar performance to heavy-weight heads in most cases. Therefore, we use light-weight heads for pseudo-supervision in our experiments to make the training more efficient.



(a) (b)

Figure 2: **Positive transfer with CLIPTeX.** (a) Bounding box frequency for PASCAL VOC classes in CC3M's pseudo-labels obtained with Mask R-CNN. (b) Class-wise IoU gap (in %) between CLIP-FT and CLIPTeX when linear probed on the PASCAL VOC.

Table 4: **Role of head complexity (light and heavy) when training with pseudo-labels on CC3m.** #layers denote the number of convolutional layers used in the task head. Results with different probes for different dense prediction tasks are reported (see Section 3 for details). For segmentation, we report the results on the PASCAL VOC dataset. We observe similar trends in ADE20k dataset.

# layers	Segmentation (↑)		Detection (↑)		Depth (↓)		Surface Normal (↑)	
	Linear	PSPNet	Mask R-CNN	SSD	Linear	PSPNet	Linear	PSPNet
1	73.43	80.71	28.89	17.50	0.159	0.128	39.96	50.80
3	66.70	80.24	28.64	17.43	0.155	0.127	40.55	51.72

C Results

Tables 5 to 7 compares the results of CLIPTeX with other baselines on different tasks and datasets with different heads. We observe that pseudo-supervision via experts in CLIPTeX improves performance by large across different tasks and datasets.

Table 5: **Probing results for semantic segmentation.** A higher value of mIoU is better.

Model	ADE20k			PascalVOC		
	Linear	DeepLabV3	PSPNet	Linear	DeepLabV3	PSPNet
ViT-B/16						
CLIP	6.78	16.15	17.32	18.66	43.75	45.53
CLIP-FT	26.60	37.11	38.80	62.47	77.67	78.22
CLIPTeX **(Ours)**	**29.26**	**39.20**	**39.70**	**73.43**	**80.57**	**80.71**
ViT-H/16						
CLIP	24.18	33.39	34.86	56.18	73.12	75.37
CLIP-FT	32.20	43.05	44.24	62.95	81.73	82.94
CLIPTeX **(Ours)**	**36.17**	**45.43**	**45.63**	**79.30**	**84.06**	**84.31**
ResNet-50						
CLIP	11.98	29.51	28.22	**46.96**	70.34	70.92
CLIP-FT	11.30	34.86	33.97	34.78	73.70	74.17
CLIPTeX **(Ours)**	**12.93**	**35.45**	**34.80**	40.31	**75.82**	**75.58**

D Hyperparameters

Hyper-parameters used during training and probing CLIPTeX and other models are given in Table 8 and Table 9 respectively.

Table 6: **Probing results for object detection, instance segmentation, and image classification.** In (a), for Mask R-CNN, we report mAP (higher is better) for bounding box and instance segmentation while for SSD, we report mAP only for bounding box on the COCO dataset. In (b) top-1 accuracy (higher is better) is reported.

(a) Detection and instance segmentation on COCO.

Model	Mask R-CNN		SSD
	BBox	**Instance**	**BBox**
ViT-B/16			
CLIP	15.20	12.16	5.33
CLIP-FT	27.21	23.18	16.46
CLIPTeX **(Ours)**	**28.89**	**24.92**	**17.50**
ViT-H/16			
CLIP	26.65	21.29	11.07
CLIP-FT	33.93	28.92	20.24
CLIPTeX **(Ours)**	**34.50**	**29.60**	**21.55**
ResNet-50			
CLIP	29.49	25.61	20.32
CLIP-FT	38.13	34.02	**30.28**
CLIPTeX **(Ours)**	**38.23**	**34.04**	28.62

(b) Image classification.

Model	**ImageNet**	**Places365**
ViT-B/16		
CLIP	**80.24**	**55.52**
CLIP-FT	79.94	55.21
CLIPTeX **(Ours)**	79.64	55.36
ViT-H/16		
CLIP	**84.85**	**56.96**
CLIP-FT	84.1	55.81
CLIPTeX **(Ours)**	83.2	55.96
ResNet-50		
CLIP	78.35	56.55
CLIP-FT	78.92	56.98
CLIPTeX **(Ours)**	**78.95**	**57.22**

Table 7: **Probing results for depth and surface normal estimation on NYU-V2 dataset.** Following Lasinger et al. [18], we report absolute relative error (lower is better) for depth estimation. For surface normal estimation, we report $a<30$ following Bae et al. [1] (higher is better).

(a) Depth estimation.

Model	**Linear**	**DeepLabV3**	**PSPNet**
ViT-B/16			
CLIP	0.235	0.189	0.168
CLIP-FT	0.215	0.145	0.139
CLIPTeX **(Ours)**	**0.159**	**0.129**	**0.128**
ViT-H/16			
CLIP	0.212	0.151	0.132
CLIP-FT	0.213	0.131	0.125
CLIPTeX **(Ours)**	**0.138**	**0.118**	**0.117**
ResNet-50			
CLIP	**0.212**	0.156	**0.147**
CLIP-FT	0.239	0.160	0.155
CLIPTeX **(Ours)**	0.220	**0.153**	0.150

(b) Surface normal estimation.

Model	**Linear**	**DeepLabV3**	**PSPNet**
ViT-B/16			
CLIP	28.49	45.17	47.29
CLIP-FT	29.06	47.74	47.91
CLIPTeX **(Ours)**	**39.96**	**50.95**	**50.80**
ViT-H/16			
CLIP	29.09	47.31	49.78
CLIP-FT	29.21	49.73	50.48
CLIPTeX **(Ours)**	**43.22**	**53.23**	**53.89**
ResNet-50			
CLIP	**33.67**	46.05	47.28
CLIP-FT	28.72	46.99	48.66
CLIPTeX **(Ours)**	31.56	**47.92**	**49.44**

Table 8: **Hyper-parameters for training** CLIPTeX **on CC3M dataset..**

Hyper-parameter	**Value**
Epochs	30
LR scheduler	cosine
Warmup Steps	1000
Warmup Init LR	1e-06
Maximum LR	3e-05
Minimum LR	1e-06
Batch size	32
λ_{depth}	1.0
λ_{clip}	1.0
λ_{seg}	0.1
$\lambda_{\text{surface normal}}$	1.0

Table 9: Hyper-parameters used for probing on different downstream tasks.

Hyper-paramater	**Segmentation**			**Detection**		**Depth**			**Surface Normal**			**Classification**
	Linear	DeepLabv3	PSPNet	Mask R-CNN	SSD	Linear	DeepLabv3	PSPNet	Linear	DeepLabv3	PSPNet	Linear
Epochs	50	50	50	25	200	50	50	50	50	50	50	40
LR scheduler	cosine	cosine	cosine	multi-step	cosine	cosine	cosine	cosine	cosine	cosine	cosine	cosine
Warmup Steps	500	500	500	250	500	1000	1000	1000	1000	1000	1000	1000
Warmup Init LR	1e-06	1e-06	1e-06	1e-05	9e-05	1e-06	1e-06	1e-06	1e-06	1e-06	1e-06	1e-06
Maximum LR	3e-05	3e-05	3e-05	3e-04	9e-04	1e-04	1e-04	1e-04	1e-05	1e-05	1e-05	3e-05
Minimum LR	3e-06	3e-06	3e-06	NA	1e-06	1e-06	1e-06	1e-06	1e-06	1e-06	1e-06	1e-06
LR Milestones	NA	NA	NA	[22, 24]	NA	NA	NA	NA	NA	NA	NA	NA
LR Gamma	NA	NA	NA	0.1	NA	NA	NA	NA	NA	NA	NA	NA
Batch size	32	32	32	4	32	16	16	16	16	16	16	128