Pushing the Limits of Simple Pipelines for Few-Shot Learning: Supplemental Material

In this supplemental material, we present:

- In Section 2, we include additional (per-domain) results for Table 1 in the main paper.
- In Section 3, we include additional (per-domain) results for Table 1 and Table 4 in the main paper.
- In Section 4, we investigate the impact of the hyper-parameters for the fine-tuning phase.
- In Section 5, we show the T-SNE plots before and after ProtoNet meta-training.

1. Per-Episode vs Per-Domain Fine-Tuning

Our fine-tuning results in the main paper were based on per-episode fine-tuning. As discussed in Section 3.3, this means a learning rate is selected per episode based on the augmented support set. Alternatively, if we are able to see a few labeled episodes from a particular domain, a per-domain learning rate selection can be done, which would not impose the same additional computational overhead once tuned on a given domain.These episodes would essentially correspond to a domain-wise validation set which is not provided in standard cross-domain FSL benchmarks, although it could be a reasonable assumption for many practical cross-domain scenarios. In this supplement, we also report per-domain results.

2. Additional results for Meta-Dataset

In this section, we show a complete view of the results presented in Table 1 in the main paper, including the outcomes of different pre-training methods (see Table 1), the outcomes of meta-training on ImageNet domain (see Table 2), and the outcomes of meta-training on eight pre-specified domains (see Table 3).

As indicated in the main paper, our pipeline is named in a form of “P > M > F (backbone)”, where “P”, “M” and “F” are taken from the first letters of pre-training, meta-training and fine-tuning respectively. In this section, we only examine the pre-training and backbone architecture parts with meta-training fixed to ProtoNet. As an example, in Table 2, we use “DINO > PN (ViT-small)” to denote the pipeline that uses DINO pre-training, ProtoNet meta-training with backbone architecture being ViT-small.

To clarify the shorten notations in Table 1, Table 2 and Table 3, we make a list here:

- DINO: self-distillation pre-training on ImageNet-1k dataset by [2].
- BEiT: BERT pre-training on ImageNet-21k dataset by [1].
- CLIP: Contrastive language-image pre-training on YFCC100M dataset by [3].
- Sup21k: Supervised pre-training on ImageNet-21k dataset.
- Sup1k: Supervised pre-training on ImageNet-1k dataset.
- BEiT + Sup21k: BERT unsupervised pre-training first on ImageNet-21k dataset and then using the labels of ImageNet-21k to fine-tune the model.

	INet	Omglot	Acraft	CUB	DTD	QDraw	Fungi	Flower	Sign	COCO	Avg
DINO (ViT-small)	73.48	54.33	62.17	85.37	83.67	60.59	56.26	94.45	53.7	54.58	67.86
DINO (ViT-base)	74.85	59.44	55.36	80.08	84	59.61	56.65	94.84	51.81	57.1	67.374
BEiT (ViT-base)	17.12	23.96	17.21	18.59	39.79	23.89	13.69	45.81	16.16	16.36	23.258
CLIP (ViT-base)	60.66	62.12	54.08	80.26	76.51	62.90	30.76	68.43	47.33	41.95	58.5
DINO (ResNet50)	64.13	52.51	57.02	62.63	84.5	60.78	50.41	92.18	58.27	55.43	63.786
CLIP (ResNet50)	51.67	44.16	44.18	70.2	70.64	47.88	34.13	87.97	39.59	41.63	53.205
Sup21k (ViT-base)	67.00	37.02	47.72	82.9	79.77	52.25	41.98	95.7	46.22	53.46	60.402
BEiT + Sup21k (ViT-base)	33.85	23.95	33.92	52.07	63.79	32.60	28.19	67.3	27.18	29.65	39.25
Sup1k (ViT-base)	89.1	60.71	55.36	79.8	79.75	61.28	47.45	88.44	56.3	57.20	67.539
Sup1k (ResNet50)	76.22	47.31	55.75	76.40	80.40	51.26	43.42	85.48	50.46	57.10	62.38

Table 1. **Pre-training results on Meta-Dataset** – Comparison of different pre-training methods and backbone architectures.

	In-domain	Out-of-domain									
	INet	Omglot	Acraft	CUB	DTD	QDraw	Fungi	Flower	Sign	COCO	Avg
DINO > PN (ViT-small)	74.69	56.91	60.5	85.04	84.21	61.54	54.78	94.57	54.21	57.35	68.38
DINO > PN (ViT-base)	76.69	62.2	54.76	81.58	84.48	60.64	55.93	95.14	56.81	60.27	68.85
CLIP > PN (ViT-base)	76.03	59	65.75	90.2	83.08	65.45	53.2	96.35	58.65	61.2	70.891
DINO > PN (ResNet50)	67.08	49.21	58.46	72.08	85.01	59.2	50.53	89.91	55.44	53.94	64.086
CLIP > PN (ResNet50)	69.41	60.72	57.53	83.66	80.03	55.58	50.07	93.39	48.56	50.14	64.909
Sup21k > PN (ViT-base)	85.88	39.72	52.03	94.54	83.42	54.58	57.06	99.01	47.74	69.02	68.3
BEiT+Sup21k > PN (ViT-base)	84.39	60.54	74.04	95.66	86.14	65.24	64.25	99.19	63.02	69.91	76.238
Sup1k > PN (ViT-base)	90.48	62.96	54.89	78.88	80.02	61.81	45.52	88.56	55.61	59.12	67.785

Table 2. **Meta-training results on Meta-Dataset (ImageNet only)** – Comparison of different pre-training methods and backbone architectures.

	In-domain								Out-of-domain		
	INet	Omglot	Acraft	CUB	DTD	QDraw	Fungi	Flower	Sign	COCO	Avg
DINO > PN (ViT-small)	73.54	91.79	88.33	91.02	81.64	79.23	74.2	94.12	54.37	57.04	78.528
DINO > PN (ViT-base)	73.55	91.54	89.73	92.94	81.52	80.2	78.28	94.53	53.65	59.13	79.507
CLIP > PN (ViT-base)	74.76	92.26	91.42	93.55	80.97	80.8	79.13	95.64	54.52	56.8	79.985
DINO > PN (ResNet50)	63.7	85.91	80.3	81.67	82.69	72.84	60.03	91.75	54.26	50.67	72.382
CLIP > PN (ResNet50)	64.86	92.09	89.19	89.17	71.67	78.71	76.15	91.25	51.1	45.88	75.007
Sup21k > PN (ViT-base)	84.86	85.71	83.77	95.89	85.1	78.47	74	99.17	59.86	67.57	81.44
BEiT+Sup21k > PN (ViT-base)	81.96	94.19	91.62	93.76	81.3	83.48	81.76	98.84	58.83	61.81	82.755
Sup1k > PN (ViT-small)	83.87	91.22	87.9	89.2	78.11	78.7	70.33	94	56.24	57.16	78.673
Sup1k > PN (ViT-base)	89.75	93.48	91.15	92.48	78.52	80.65	75.97	95.78	53.47	55.89	80.714
Sup1k > PN (ResNet50)	68.04	86.17	80.72	80.48	71.65	70.78	59.58	84.33	50.06	50.29	70.21
None > PN (ViT-small)	37.25	74.14	45.25	49.66	61.49	70.24	43.23	72.03	39.33	35.43	52.805
None > PN (ResNet50)	40.74	90.67	80.67	68.88	62.4	75.96	55.72	75.37	43.11	35.49	62.901

Table 3. **Meta-training results on Meta-Dataset** – Comparison of different pre-training methods and backbone architectures.

3. Additional results for miniImageNet and CIFAR-FS

We also evaluate different pre-training methods and backbones on miniImageNet and CIFAR-FS, which is shown in Table 4. We do not include some of the results to the main paper because supervised pre-training on ImageNet is only useful to check the upper bound performance.

4. Ablation study on fine-tuning's hyper-parameters

There are three hyper-parameters for the fine-tuning stage: the learning rate, the number of gradient descent steps and the probability of switching on data augmentation for the support set. We show in Figure 1 that the dominant hyper-parameter is the learning rate. From the results, we also see that the higher the probability of switching on data augmentation the better, while 50 gradient steps give relatively good performance with the right learning rate. Therefore, we fix the probability to 0.9

	miniImageNet		CIFAR-FS	
	5w1s	5w5s	5w1s	5w5s
DINO > PN (ViT-small)	93.1	98.0	81.1	92.5
DINO > PN (ViT-base)	95.3	98.4	84.3	92.2
CLIP > PN (ViT-base)	93.1	98.1	85.3	93.2
DINO > PN (ResNet50)	79.2	92.0	73.7	84.0
CLIP > PN (ResNet50)	78.9	92.2	71.4	82.6
Sup21k > PN (ViT-base)	97.2	99.2	92.3	96.7
BEiT+Sup21k > PN (ViT-base)	96.6	99	93.8	97.5
Sup1k > PN (ViT-small)	97.7	99.4	86.2	93.6
Sup1k > PN (ViT-base)	99.2	99.8	88.2	94.3
Sup1k > PN (ResNet50)	91.7	97.4	77	87.6
None > PN (ViT-small)	36.5	49.1	45.9	59.8
None > PN (ResNet50)	46.1	60.3	54.1	68.4

Table 4. **miniImageNet & CIFAR-FS** – Comparison of different pre-training methods and backbone architectures.

and let the numbers of steps to be 50 in the fine-tuning phase.

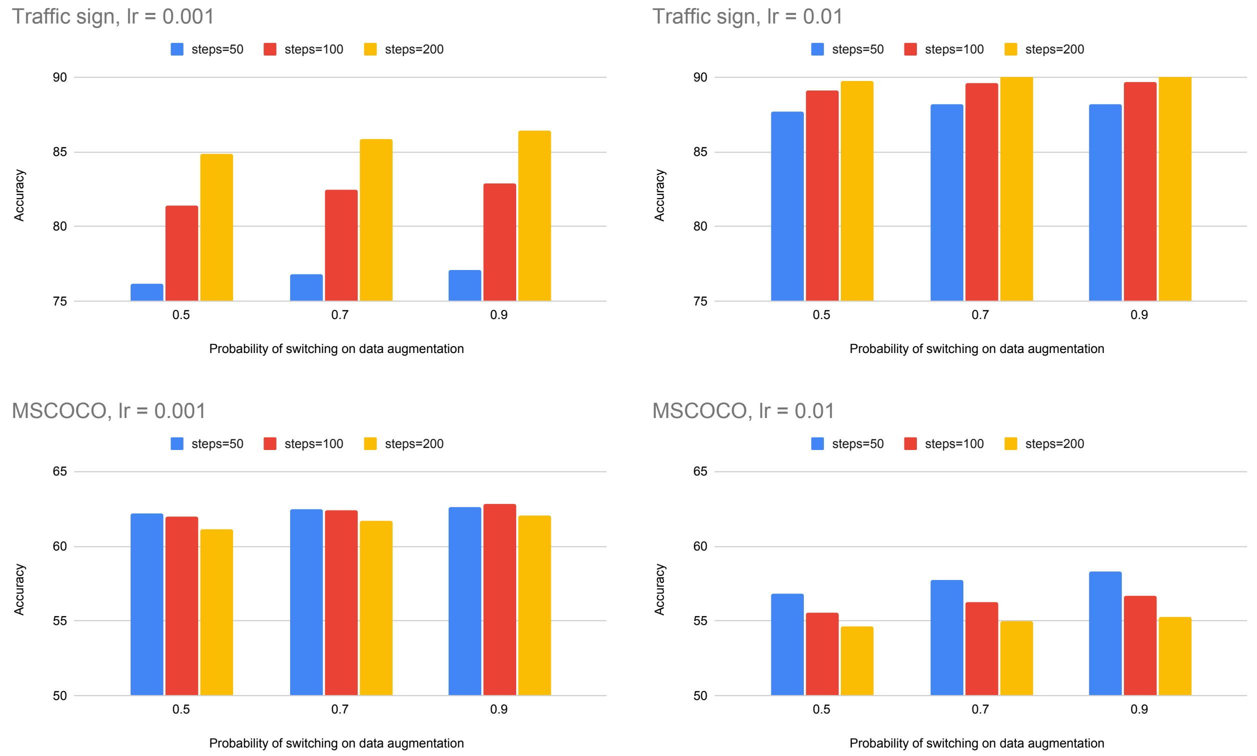


Figure 1. **Ablation study of fine-tuning's hyper-parameters** – The experiments are done in the validation set of the traffic sign domain and the MSCOCO domain with learning rate fixed to either 0.001 or 0.01.

5. T-SNE plots: before and after meta-training

By using T-SNE visualization, We identify that the feature representation of DINO pre-training is already of high quality in multiple domains. Three examples are shown in Figure 2, Figure 3 and Figure 4. In general, many semantic clusters have already emerged, even though these domains where the clusters are sitting are not necessarily similar to ImageNet. This gives

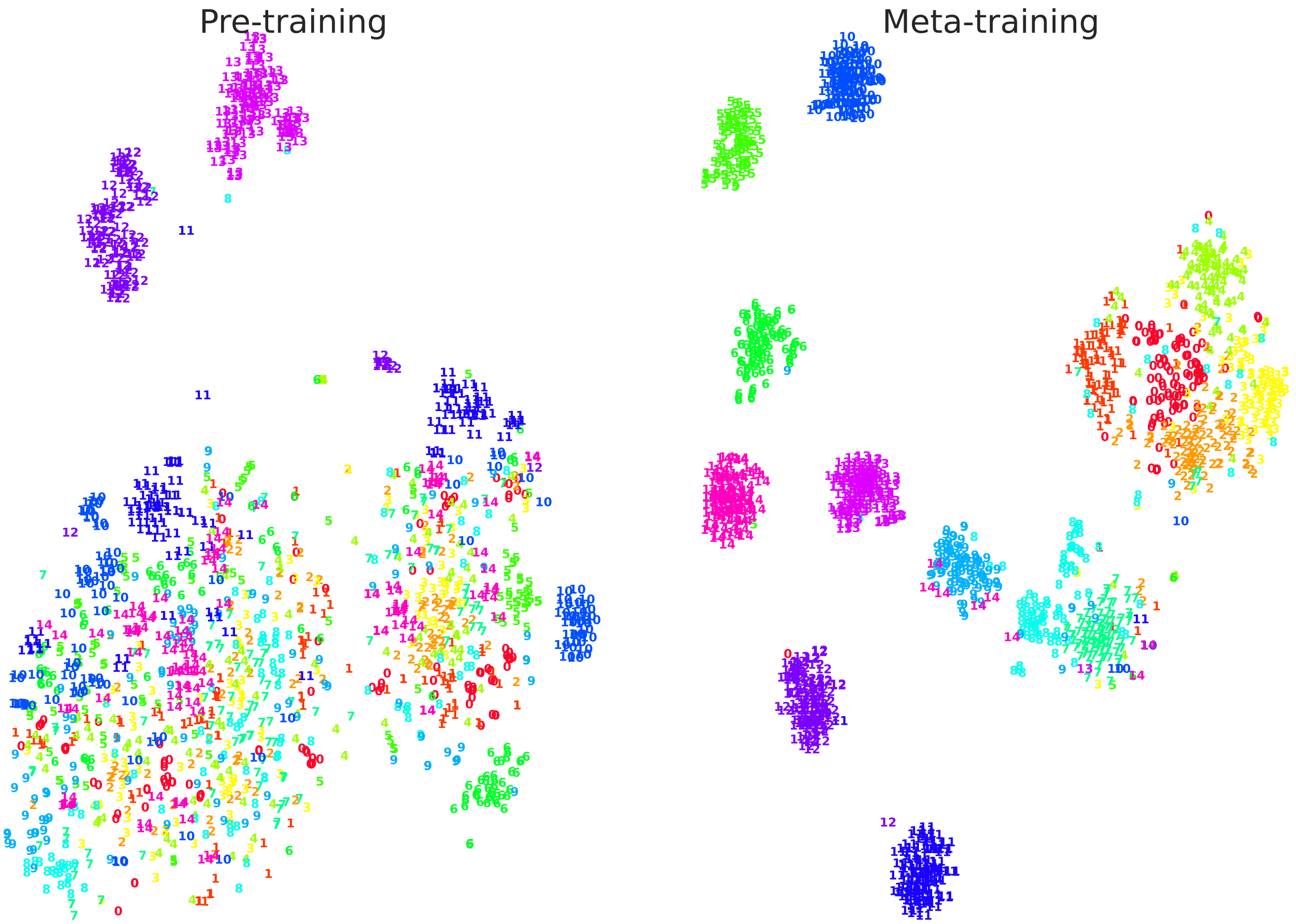
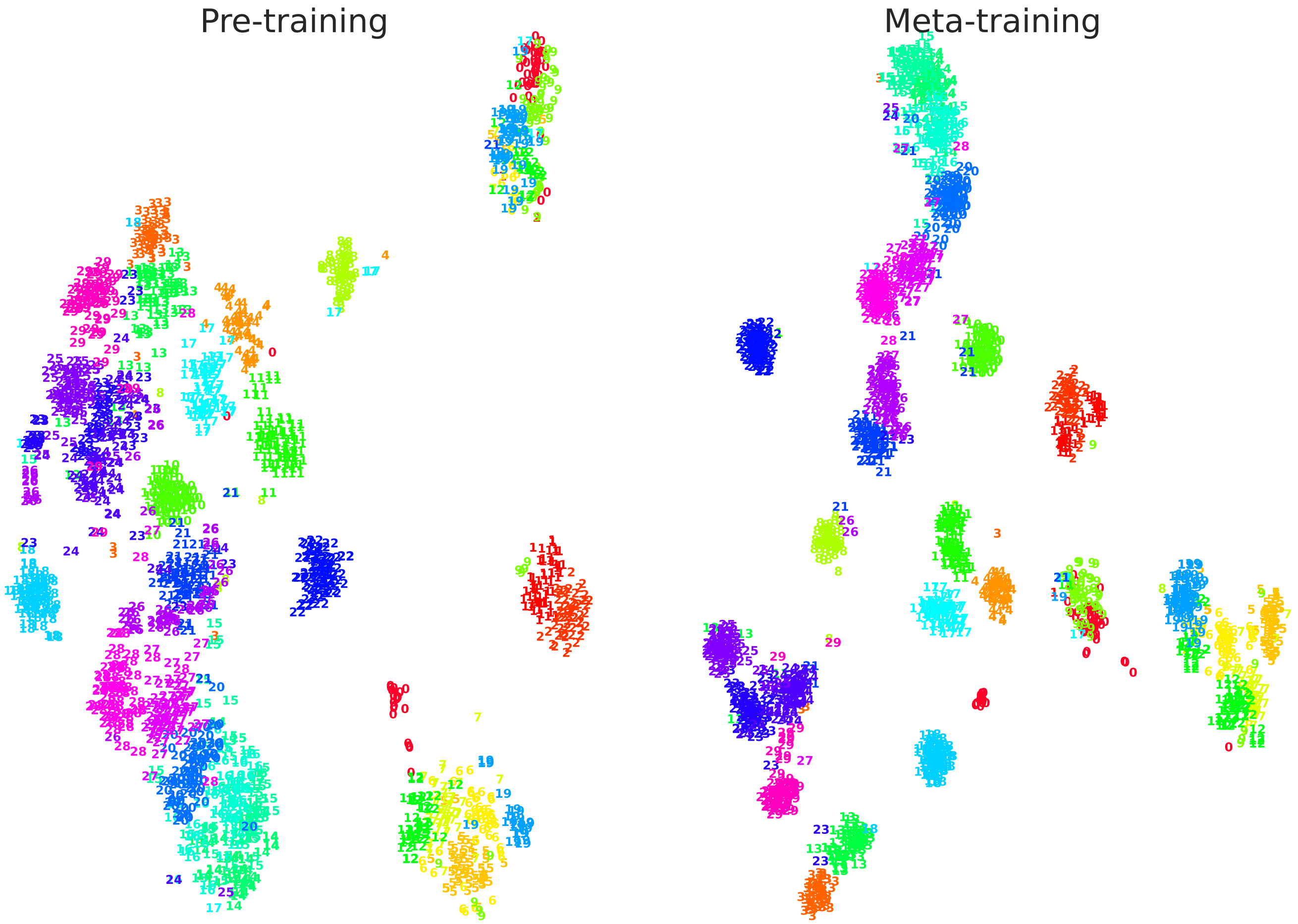


Figure 2. **Aircraft domain**

a very good initialization to ProtoNet so that it can refine the clusters to be much tighter. While the situation would be quite different if we were training the ProtoNet from scratch, which are confirmed by the no-pre-training results in Table 3. This can be explained in the sense of K-means clustering, where a good initialization is always desired.



Pre-training

Meta-training

Figure 3. **CUB domain**

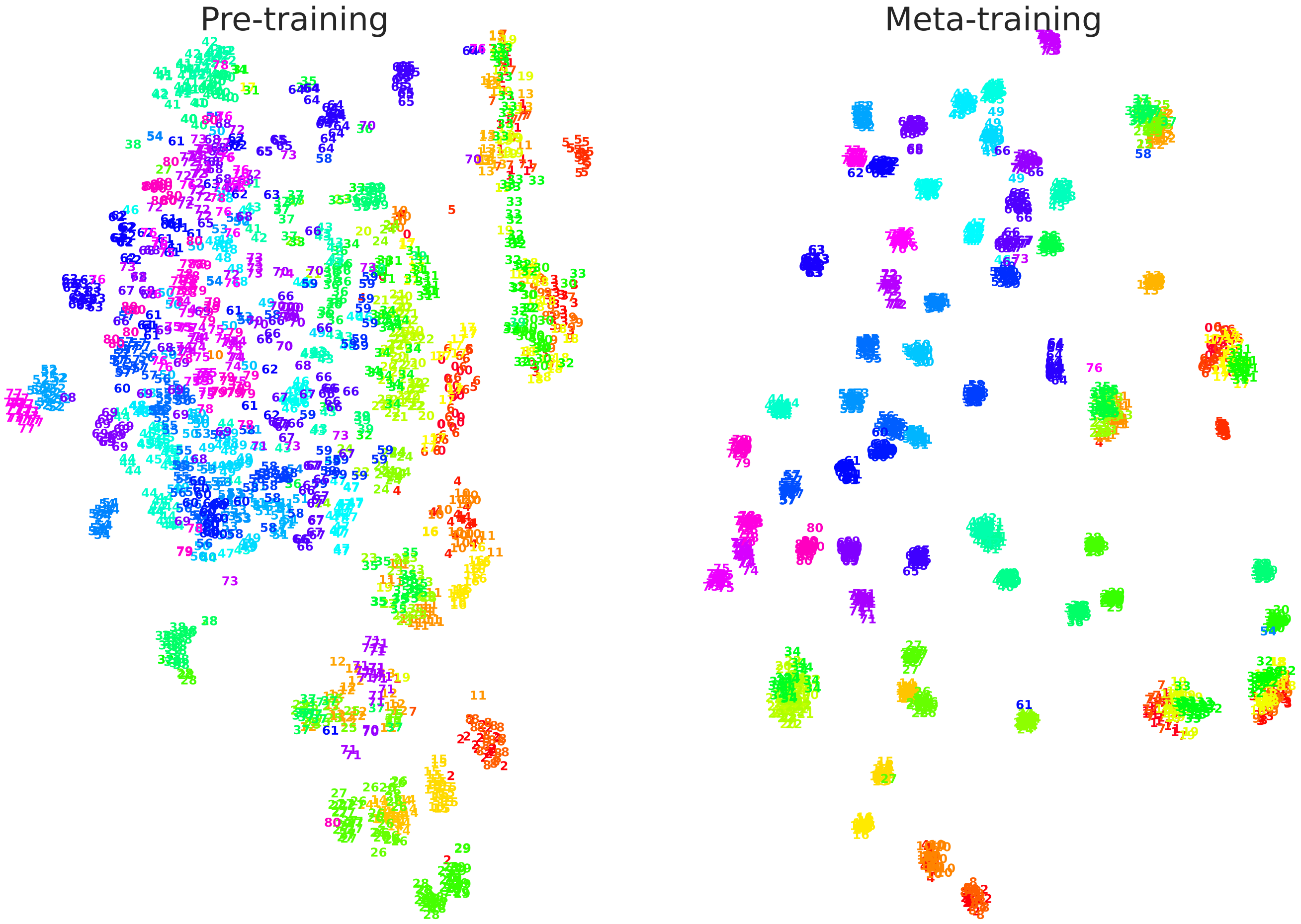


Figure 4. **Omniglot domain**

References

[1] Hangbo Bao, Li Dong, and Furu Wei. Beit: Bert pre-training of image transformers. In *ICLR*, 2022. 1

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

[3] 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. 1