Hyperbolic Contrastive Learning for Visual Representations beyond Objects
Supplemental Material

A. Experiment setups

In this section, we provide additional details of our experiments.

Unsupervised object proposals. When pretraining on uncurated datasets, acquiring ground truth object bounding boxes using human annotations can be expensive. However, automatically generating unsupervised region proposal is well studied. We use Selective Search as the unsupervised proposal generation method. Following ORL [10] we first generate the proposals using selective search. Then we filter the proposals with 96 pixels as the minimal scale, maximum IOU of 0.5 and aspect ratio between 1/3 to 3. For every image we generate maximum of 100 proposals and randomly select any image as the object image.

OpenImages dataset. We use the full OpenImages dataset which have bounding box annotations ($\sim$ 1.9 million images). We also use a subset proposed in [9]. This is a subset created from the OpenImages dataset where each image has at least 2 classes present and each class has at least 900 instances. This subset is a balanced subset of OpenImages with an average of 12 object present in an image, making it a good proxy for real-world multi-object images.

Object and Scene image augmentations. We find that small objects are always detrimental to performance. Therefore, when sampling object bounding boxes, we drop bounding boxes with size $\texttt{width} \times \texttt{height} \leq 56 \times 56$. Further, when sampling objects for the Euclidean branch, if the size of a bounding box $\texttt{width} \times \texttt{height} \leq 256 \times 256$, we slightly expand it to either 256×256 or the maximal size allowed by the original image size. We also apply a small jittering to the `width` and `height` to include different contexts around the objects. Next, we apply random cropping and resizing with the same scale $(0.2, 1.)$ as in MoCo [6]. When sampling objects for the hyperbolic branch, we do not apply jittering and random cropping, but only filter the small boxes and resize to $\leq 224 \times 224$. To crop the scene images, we sample another 1 to 5 bounding boxes and merge with the selected object bounding box.

Model details of pre-training. For the optimizer setups and augmentation recipes, we follow the standard protocol described in MoCo-v2 [2]. We find that a base learning rate of 0.3 works better when pre-training on COCO and OpenImage datasets as compared to 0.03. We adopt the linear learning rate scaling receipt that $lr = 0.3 \times \text{BatchSize}/256$ [5] and batch size of 128 by default on 4 NVIDIA p6000 gpus. To ensure fair comparison, we also pre-train the baselines with a learning rate of 0.3. We train our models on COCO and the subset of OpenImage datasets for 200 epochs and full OpenImage dataset for 75 epochs. We also note that calculating hyperbolic loss itself takes nearly the same time as a normal contrastive loss. The only overhead in pre-training is one additional forward pass to get scene representations. In our setting, MoCo takes 0.616 sec/iter while HCL takes 0.757 sec/iter. For the hyperparameters of our hyperbolic objective, we use $r = 4.5$, $\lambda = 0.1$, and $\varepsilon = 1e^{-5}$ as our default setting.

B. Additional experimental results

B.1. Robustness under Corruption.

We calculate the mCE error as in Hendrycks et al. [7]. We compare our HCL model trained on OpenImages and lineval on ImageNet dataset with the baseline model without using HCL loss. We see an improvement of 1.9 mCE over the baseline model, demonstrating that our HCL model learns more robust representations as compared to the vanilla MoCo.

B.2. Fine-grained class classification

Method	Cars [8]	DTD [3]	Food [1]
HCL/$\mathcal{L}_{\text{hyp}}$	31.92	68.46	58.66
HCL	32.02	68.19	58.79

Table 1. Fine grained classification results.

In Table 1 we show results on fine-grained classification datasets. We can see that on fine-grained classification our model provides little performance improvement. This could be due to the fact that all classes in these datasets have very

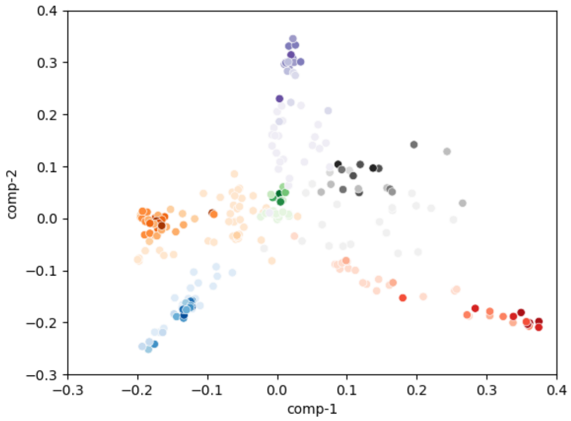


Figure 1. PCA visualization from randomly sampled images from COCO validation set. The various colors denote distinct scenes, and the denser dots signify a higher concentration of objects within the area. The data strongly suggest that regions with a greater number of objects generally exhibit higher magnitudes.

similar scene contexts, and hence the hyperbolic objective does not help very much.

B.3. Visualization of the representation space:

To show that the norm of the scenes is larger the objects; we randomly sample a few images with multiple objects from the COCO validation set. For each scene, we compute representations for its objects and regions and apply PCA to obtain the visualization in the figure above. Each color represents a different scene and the darker dots indicate more objects in the region. It is clearly shown that regions with more objects tend to have larger norms.

B.4. More ImageNet Examples

More visualization results on ImageNet are presented in Figures 2 and 3.

C. Additional ablation studies

In this section, we provide more ablation experiments on hyperbolic linear evaluation, model architecture, and the radius of Poincare ball. All the models are trained on the OpenImages dataset and evaluated on the ImageNet-100 (IN-100) or ImageNet-1k (IN-1k) the top-1 accuracy reported.

Radius of the Poincare ball. In Table 2 we show results by varying the radius of Poincaré ball. The hyperbolic objective improves the performance over all the tested radius. We find that a too small radius may lead to a smaller improvement due to the stronger regularization.

Configuration of the encoder head. In our experiments, the Euclidean and hyperbolic branches share the weights in both the backbone and the head of the encoders. We also try using a separate head for the hyperbolic branch. As shown in Table 3, this leads to a more stable training when larger learning rate is applied. However, we did not see any improvements brought by this modification.

c	1	0.5	0.1	0.05	0.01
IN-1k Acc.	58.08	58.31	58.29	58.51	58.49

Table 2. Results by varying the radius r of Poincaré ball. $c = \frac{1}{r^2}$.

Head	λ	IN-100 Acc.
N/A	0	77.36
shared	0.1	79.08
	0.5	0
splitted	0.1	77.88
	0.5	77.58

Table 3. Different configurations of head in the the Euclidean and hyperbolic branches.

SGD		Adam	
lr	IN-100	lr	IN-100
0.1	63.82	0.001	67.64
0.2	64.22	0.0005	70.32
0.3	1	0.0001	72.58
0.4	1	0.00005	70.5

Table 4. Results of hyperbolic linear evaluation with different optimizers and learning rates.

Hyperbolic linear evaluation. Apart from the common linear evaluation in the Euclidean space, we show the hyperbolic linear evaluation results with different optimizers and learning rates in Table 4. The idea is to test if the representations are more linearly separable in the hyperbolic space. We follow the same setting of hyperbolic softmax regression [4] and train a single hyperbolic linear layer. However, we find the optimization with SGD can easily cause overflow. By contrast, Adam is much more stable with appropriate learning rates.

C.1. Downstream performances by varying objects:

We divide the COCO validation set into two splits based on whether the image has more than 5 objects or not. We report the object detection and semantic segmentation results of MoCo-v2 and HCL on each split in the table below. We first note that images with more objects pose additional difficulty for these tasks. We also find that our method

	AP^{b}	$\mathrm{AP}^{\mathrm{b}}_{50}$	$\mathrm{AP}^{\mathrm{b}}_{75}$	AP^{m}	$\mathrm{AP}^{\mathrm{m}}_{50}$	$\mathrm{AP}^{\mathrm{m}}_{75}$
MoCo-v2	38.5	58.1	42.1	34.8	55.3	37.3
$\leq$ 5 objects / image	49.3	69.6	54.8	42.1	58.6	42.5
$>$ 5 objects / image	34.6	52.6	37.8	34.5	54.1	34.4
HCL	40.6	61.1	44.5	37.0	58.3	39.7
$\leq$ 5 objects / image	51.7	72.8	57.6	44.6	61.9	45.2
$>$ 5 objects / image	37.7	56.6	41.2	37.0	58.2	39.7

Table 5. Object detection and sematic segmentation results of ORL, ORL with proposal boxes, and HCL pre-trained on COCO.

Description	AP	AP_{50}	AP_{75}	AP_s	AP_l	AP_m
MoCo-v2	39.8	59.8	43.6	36.1	56.9	38.7
HCL(Ours)	**40.5**	**60.8**	**43.8**	**36.5**	**57.6**	**39.3**
MoCo-v2	57.0	82.2	63.4	-	-	-
HCL(Ours)	**58.2**	**83.1**	**64.5**	-	-	-

Table 6. Object detection results on COCO (top 2 rows) and VOC (bottom 2 rows). All SSL models have been pre-trained on ImageNet for 200 epochs and then fine-tuned on COCO and VOC.

generally improves more on the images with more objects, e.g. 3.1 vs. 2.4 on object detection AP.

C.2. Removing exponential mapping from hyperbolic loss.

To evaluate the importance of $\mathcal{L}_{hpy}$, we directly use the inner product between unnormalized scene and object representations for a contrastive objective. As shown in Table 7, we find that this leads to a performance drop on the ImageNet-100 linear evaluation from 69.95 to 53.48. This completes the picture that for modeling scene representations: hyperbolic space $>$ spherical space $>$ unconstrained flat space.

MoCo-v2	HCL w/o exp	HCL w/o $\mathcal{L}_{hpy}$	HCL
69.95	53.48	73.79	75.40

Table 7. Linear evaluation top-1 accuracy on ImageNet-100. Models are pre-trained on the OpenImage dataset.

C.3. Results on ImageNet

We also show results by pre-training on the ImageNet dataset as well. Even though ImageNet is not primarily a multi-object dataset, we still see some gains by using our hyperbolic contrastive learning method.

References

[1] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101 – mining discriminative components with random forests. In *ECCV*, 2014. 1

[2] Xinlei Chen, Haoqi Fan, Ross Girshick, and Kaiming He. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020. 1

[3] M. Cimpoi, S. Maji, I. Kokkinos, S. Mohamed, , and A. Vedaldi. Describing textures in the wild. In *CVPR*, 2014. 1

[4] Octavian Ganea, Gary Bécigneul, and Thomas Hofmann. Hyperbolic neural networks. *NeurIPS*, 2018. 2

[5] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He. Accurate, large minibatch sgd: Training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 1

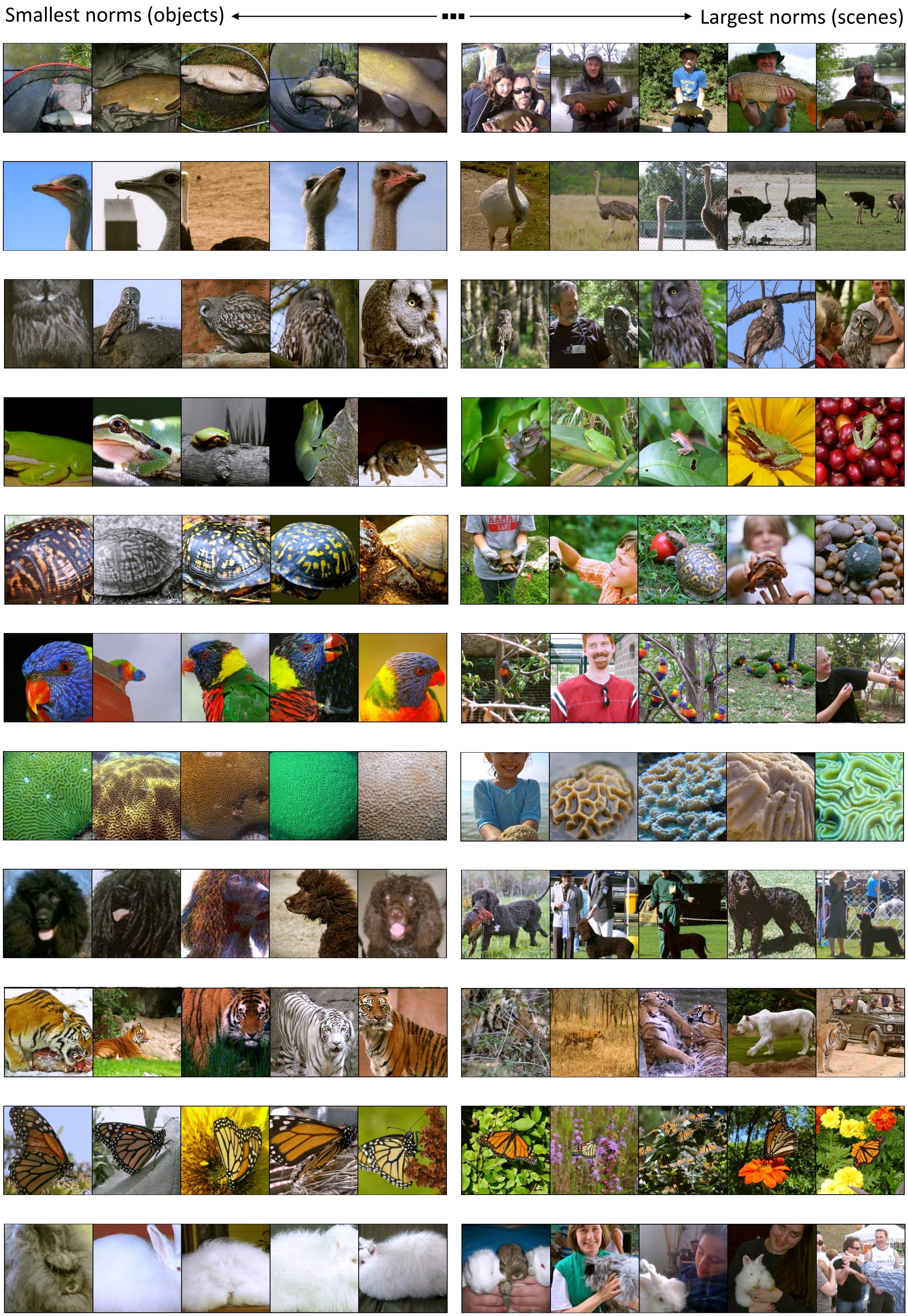
[6] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *CVPR*, 2020. 1

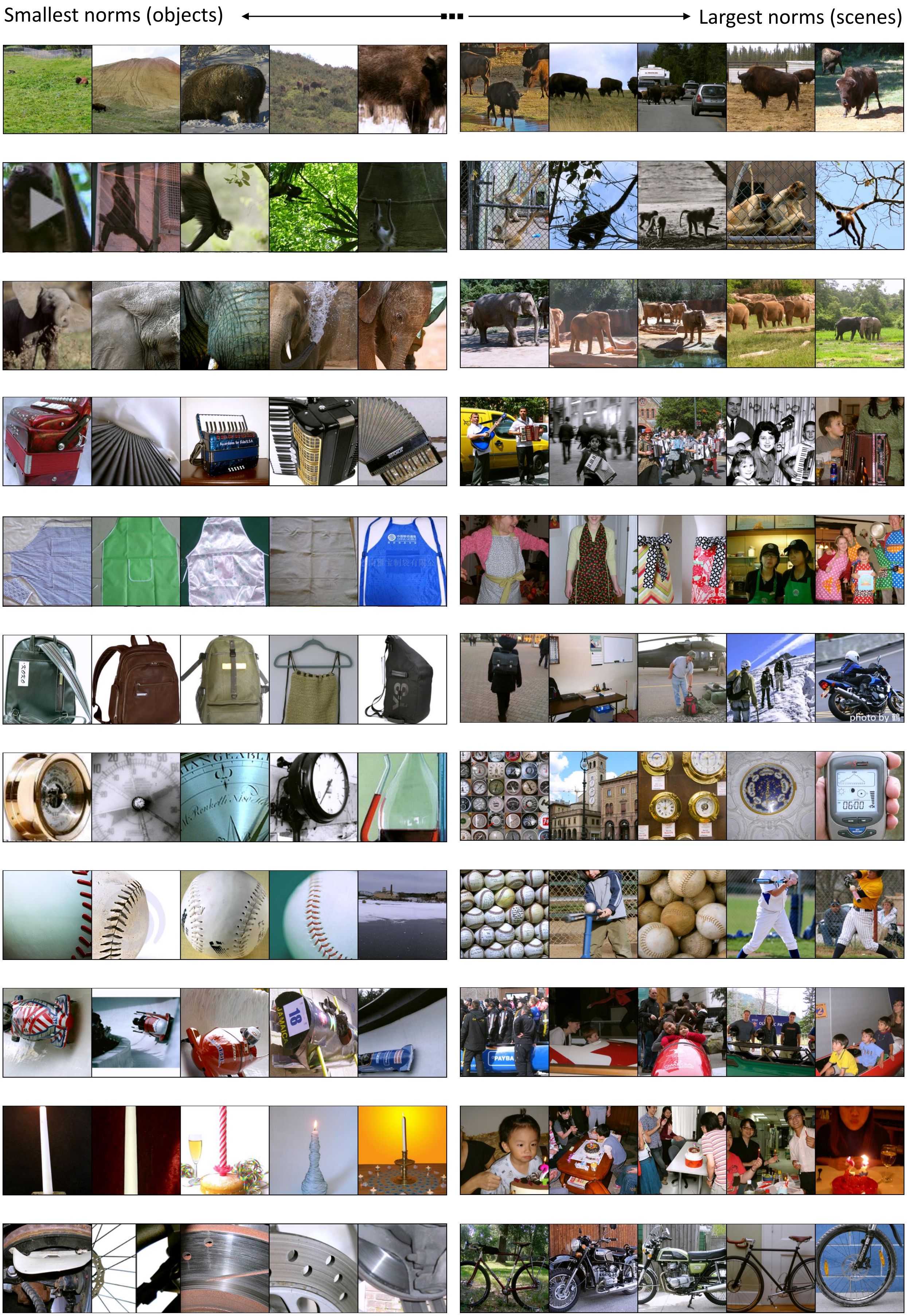
[7] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *ICLR*, 2019. 1

[8] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *3DRR*, 2013. 1

[9] Shlok Kumar Mishra, Anshul B. Shah, Ankan Bansal, Abhyuday N. Jagannatha, Abhishek Sharma, David Jacobs, and Dilip Krishnan. Object-aware cropping for self-supervised learning. *ArXiv*, abs/2112.00319, 2021. 1

[10] Jiahao Xie, Xiaohang Zhan, Ziwei Liu, Yew Soon Ong, and Chen Change Loy. Unsupervised object-level representation learning from scene images. In *NeurIPS*, 2021. 1


Smallest norms (objects)
Largest norms (scenes)


Smallest norms (objects)
Largest norms (scenes)