

# Supplementary Materials: Improving Open-World Classification with Disentangled Foreground and Background Features

Anonymous Authors

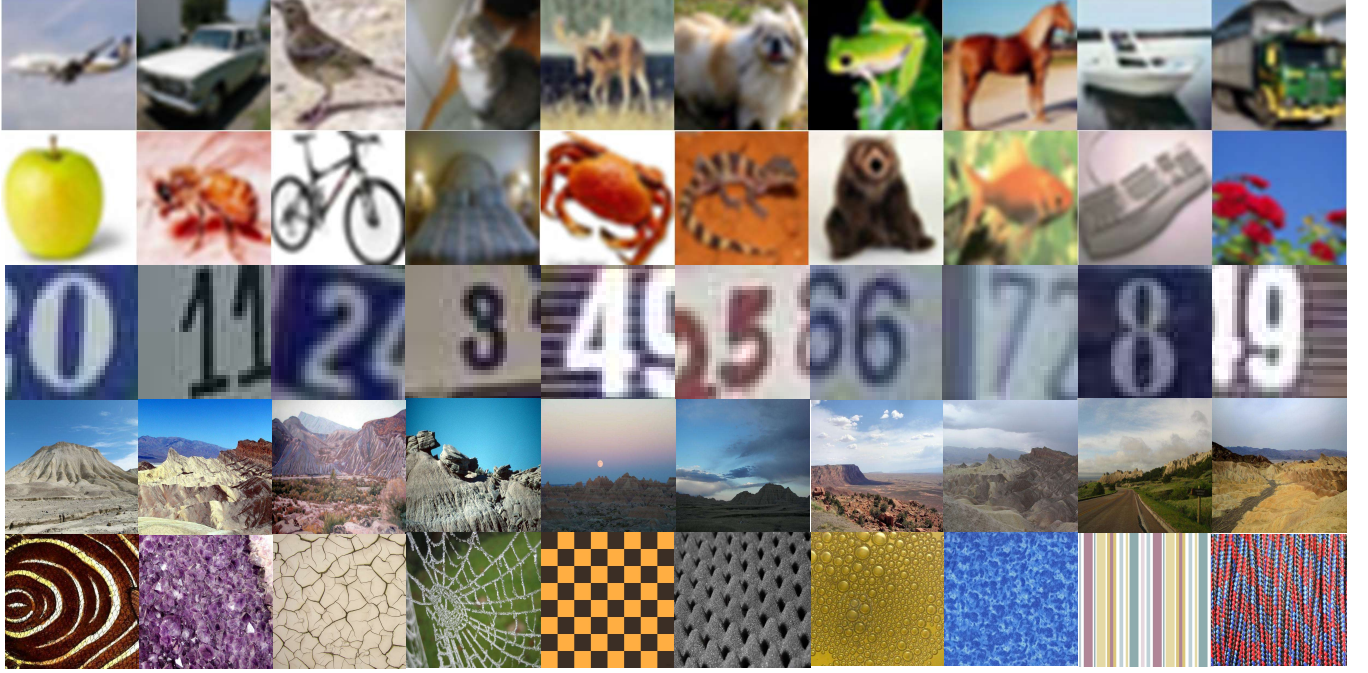


Figure 1: Example images of the five datasets we used. The rows from the top to the bottom are images from CIFAR10 [8], CIFAR100 [8], SVHN [12], Places365 [18], and Textures [2], respectively.

## A DATASETS AND WEAK SUPERVISION LABELS

### A.1 Dataset Details

**CIFAR10/CIFAR100** [8] are two subsets sampled from Tiny Image [14], respectively. Both of them consist of 60,000 32x32 images. CIFAR10 are labelled with 10 mutually exclusive classes, while CIFAR100 contains 100 classes grouped into 20 super-classes. Since they are both subsets of Tiny Image, we consider them as more challenging near-OOD detection problems when they are used as OOD data for each other.

**SVHN** [12] is a digit classification dataset cropped from house number plate pictures. It includes 600,000 32 x 32 images of printed digits (from 0 to 9). SVHN contains strong foreground OOD features and strong background OOD features owing to the significant semantic differences and scenario differences between SVHN and CIFAR10/100.

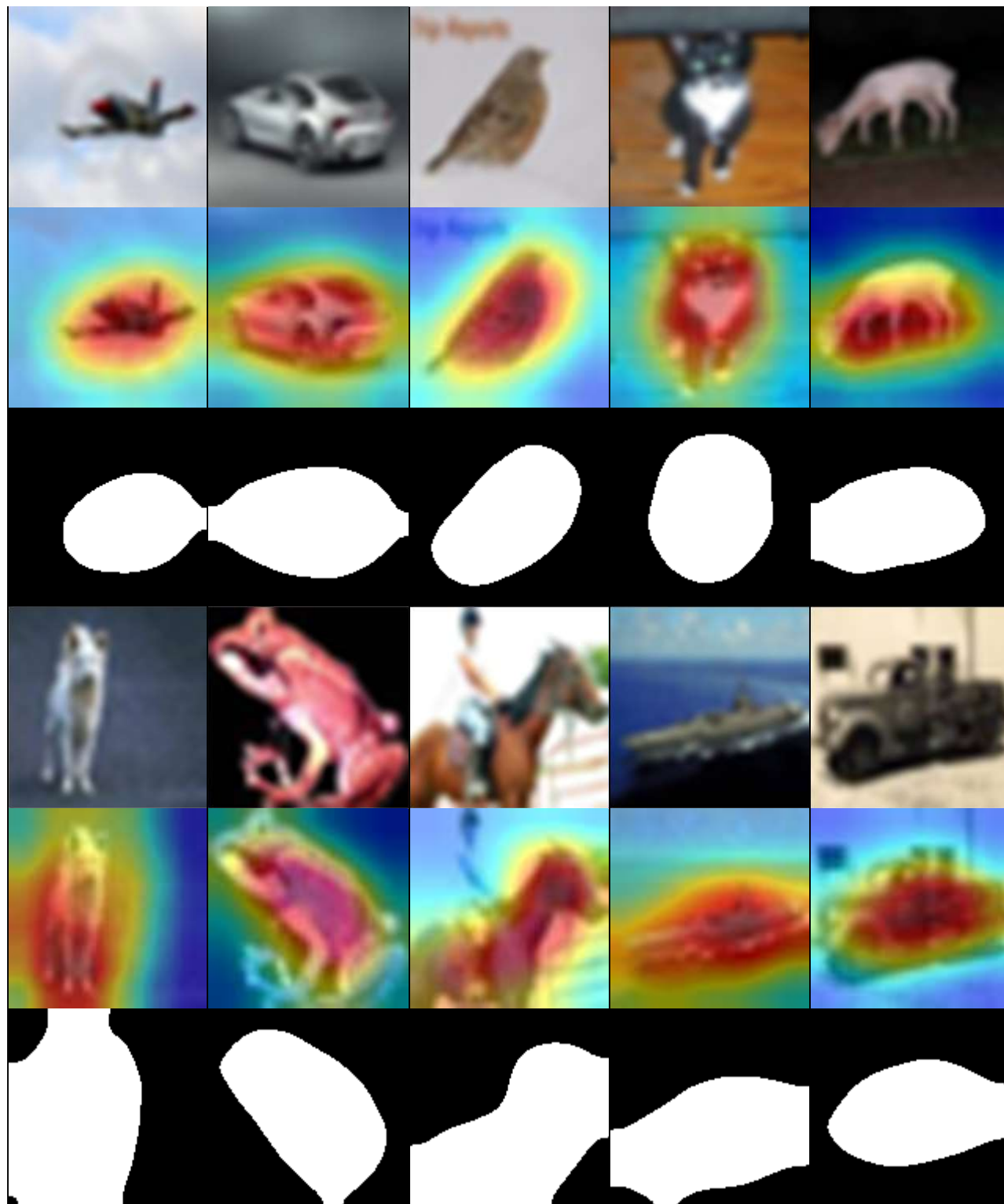
**Places365** [18] is a large-scale scene classification dataset. It has 10 million images comprising 434 scene classes. Similar to SVHN, Places365 also contains strong foreground OOD features and background OOD features against CIFAR10/CIFAR100. Following [6], we use a 10,000 images subset of Places365 as OOD data.

**Textures** [2] contains 5,640 texture images in the wild. Since texture images do not contain specific objects and backgrounds, we consider Textures has a significant difference in both foreground and background distributions against CIFAR10/CIFAR100.

To provide intuitive understanding of the foreground and background difference between ID and OOD datasets, we present 10 example images for each dataset in Fig. 1.

### A.2 Examples of the Weak Supervision (Pseudo-masks) for ID Samples

Fig. 2 shows the class activation mappings generated utilizing the pre-trained  $K$ -class classification network, and the pseudo-masks then are used to train the dense prediction network. As can be seen in the figure, the class activation mapping can generally well localize the foreground information in the image, *i.e.*, the foreground objects. Although, the pseudo-masks generated by the class activation mapping cannot segment the complex contours perfectly, they can only segment the foreground and background with a fairly good quality, which can provide sufficient supervision for supporting the learning of the background features, as shown by the results in the main text.



**Figure 2: Examples of ID samples, CAMs, and pseudo-masks for CIFAR10. For each group of examples (three rows per group), the images on the top are original image, the middle is its class activate mapping visualisation, and the bottom ones are its pseudo-mask.**

**Table 1: Ablation results. ‘BG’ is our method that uses only the background OOD score, while ‘Vanilla X’ means the use of original foreground OOD scoring function in the method X. Best results in each group are highlighted.**

Methods	In:CIFAR10					In:CIFAR100				
	CIFAR100	SVHN	Places365	Textures	Average	CIFAR10	SVHN	Places365	Textures	Average
	FPR95↓ / AUROC↑					FPR95↓ / AUROC↑				
BG	38.16/91.42	2.60/99.30	4.40/98.99	0.04/99.99	11.30/97.43	89.24/67.98	26.61/94.54	26.55/94.34	0.53/99.88	35.73/89.18
Vanilla MSP [4]	33.44/89.01	17.40/95.72	22.47/92.93	8.55/97.66	20.46/93.83	64.25/81.52	49.50/88.92	72.10/76.18	46.24/89.33	58.02/83.99
MSP-DFB	23.75/94.29	2.55/98.94	5.05/98.49	0.02/99.90	7.84/97.90	58.76/84.67	50.75/89.27	67.82/85.20	28.21/95.86	51.38/88.75
Vanilla ODIN [10]	34.62/87.83	16.13/95.66	22.15/92.43	7.45/97.86	20.09/93.45	59.67/82.39	38.11/91.32	69.80/75.39	37.38/91.10	51.24/85.05
ODIN-DFB	22.15/95.50	4.27/99.19	8.08/98.66	0.34/99.92	8.71/98.32	55.92/87.31	32.79/90.60	55.34/81.56	10.78/97.40	38.71/89.22
Vanilla Energy [11]	41.98/84.25	19.73/94.46	25.42/90.74	8.72/97.45	23.96/91.73	64.34/80.48	36.76/91.38	74.75/72.14	39.17/90.37	53.75/83.59
Energy-DFB	19.90/94.98	3.10/99.28	6.96/98.60	0.53/99.87	7.62/98.19	54.02/88.12	24.78/93.39	48.87/85.72	7.11/98.41	33.70/91.41
Vanilla ViM [15]	15.25/96.92	1.27/99.47	2.74/99.32	0.11/99.93	4.84/98.91	59.13/85.72	10.23/97.90	49.38/87.23	2.45/99.47	30.30/92.58
ViM-DFB	13.49/97.08	0.41/99.85	0.72/99.85	0.00/100.00	3.65/99.20	60.88/85.74	7.58/98.40	20.93/96.06	0.16/99.96	22.39/95.04

## B IMPLEMENTATION DETAILS

We establish all experiments based on the Google BiT-M [7] model. This model is a variant of ResNetv2 architecture [3] and is pre-trained on ImageNet-21K. We use the official release checkpoint of BiT-M-R50x1 as our pre-training parameters.

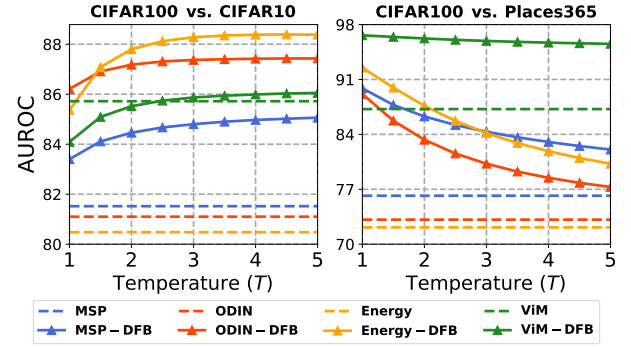
### B.1 Main Results

**In-distribution Classification Pretrain Details** We follow the BiTHyperRule [7] setting to train the baseline classification network on the in-distribution dataset (CIFAR10/CIFAR100) with pre-trained weights. The classification network was trained using 20k steps with a batch size of 128. SGD is used as parameter optimization with an initial learning rate of 0.003 and a momentum of 0.9. We used the STEP learning rate decay strategy, which decays the learning rate by a factor of 10 at 30%, 60%, and 90% of the training steps. Moreover, we used a learning rate warm-up in the first 500 steps of training. All images were resized to 160x160 and randomly cropped to 128x128. Finally, we used MixUp [16] with  $\alpha = 0.1$  to combine image samples in training.

The *post hoc* out-of-distribution detection comparison methods in the experiments are based on the classification models trained from this step.

**Generation of Pseudo-masks** Based on the well-trained classification network, we use CAM (Class Activation Mapping) [17] to generate pseudo-mask labels for the in-distribution dataset. Consistent with [1], we use the ensemble of multi-scale images to generate accurate pseudo-mask labels. Specifically, an input image is converted to a set of 8 images through 4 different scales {0.5, 1.0, 1.5, 2.0} and horizontal flips. After computing the mean values of the CAMs at all scales, the final CAM is smoothed by a Gaussian filter and converted to pseudo-mask labels based on an empirical threshold. In practice, we normalize the filtered CAMs and use 0.5 as the threshold.

**Dense Prediction Training Details** Finally, we use a modified Dense-BiT architecture to retrain the new dense prediction model with BiT-M-R50x1 checkpoints on the in-distribution dataset and pseudo-mask labels. We replace the MixUp augmentation from the training with randomly scaling (from 0.5 to 2.0) and randomly horizontally flipping augmentation. All images are resized to 128x128 during training and testing. Besides, the rest of the training strategy and hyperparameters are the consistent as the classification network.



**Figure 3: AUROC results of DFB using varying  $T$  settings in the other two OOD dataset benchmark with CIFAR100 as the ID data.**

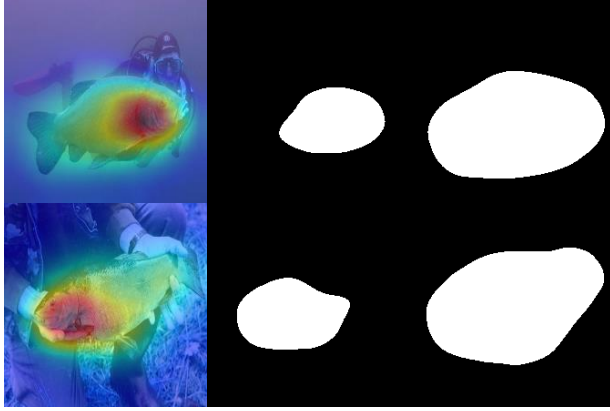
### B.2 Experiments on Large-scale Semantic Space

We implement DBF on the high-resolution large-scale dataset ImageNet-1k following the steps introduced in Sec. B.1 and evaluate its OOD detection performance in large-scale semantic spaces that have a large number of ID classes. There are two main differences in the implementation of large-scale experiments:

**Stable Mask Generation for High-resolution Images** We observe that since high-resolution images have richer appearance information, the network pays more attention to the most discriminative parts of the foreground object (e.g., the head of the fish). This leads to CAM assigning the highest class activation to the most discriminative parts and assigning the lower class activation to the rest part of the object, which degrades the quality of the resulting masks and does not completely segment the entire foreground object. Given that the average class activation of the whole object is still significantly higher than that of the background, we propose to use the mean value of the class activation of the whole image as the threshold for segmentation. As shown in Fig. 4, this trick significantly improves the pseudo-mask generated for high-resolution images.

**High-resolution OOD Datasets** In order to comprehensively evaluate the performance of DBF under high-resolution images, we add two high-resolution datasets as OOD datasets:

- **ImageNet-O** consists of images from classes that are not found in the ImageNet-1k dataset. It is adversarially filtered to fool the classifier and used to evaluate the robustness of the classifier to out-of-distribution data.



**Figure 4: Examples of pseudo-masks generated for ImageNet1k using different strategies. (Left) CAM (Middle) Pseudo-masks generated using fixed threshold, and (Right) Pseudo-masks generated using mean class activation as the threshold.**

- SUN contains 130,519 high-definition scene images from 397 categories. Following [6], we select 50 nature-related categories that do not overlap with ImageNet-1k, and randomly sample 10,000 images as the OOD dataset.

## C ADDITIONAL EXPERIMENT RESULTS

### C.1 Detailed Analysis of OOD Scoring Methods

We report the detailed results of background OOD scoring combined with all four post-hoc foreground OOD scoring methods in Tab. 1, including the use of background OOD scores  $S_b$  only (BG), foreground OOD scores  $S_h$  (Vanilla), and the full DBF model. Consistent with the results in the main text, using only the background OOD scores in DBF can yield significantly improved performance on OOD benchmarks with significant background differences from the in-distribution image. Moreover, although BG underperforms in the foreground-feature-dependent CIFAR100 vs. CIFAR10 benchmark, it still obtains competitive results in CIFAR10 vs. CIFAR100, which also relies on foreground features. This difference is caused by the number of ID categories, where more ID categories make the ID background richer and more challenging to detect OOD samples by using background features only. Holistically, the complete DBF achieves the best performance on most benchmarks, demonstrating the need to synthesize foreground and background OOD scores.

### C.2 Additional Results w.r.t. the Temperature Hyperparameter

Fig. 3 supplements the results of DBF on the remaining two benchmarks using various temperatures. Due to the SVHN, Textures and Places365 benchmarks having significant background distribution differences, all methods' performance gradually decreases as temperature increases. Note that stronger foreground OOD scoring methods (e.g., ViM) can significantly mitigate the decreasing performance trend. We choose  $T = 2.5$  as a trade-off between foreground and background scores in our experiment.

**Table 2: Comparison of DBF and outlier exposure (OE). <sup>†</sup> indicates that the results are taken from the original paper, and other methods share the same architecture. Reported results are averaged over the results on the four OOD datasets.**

Methods	ID: CIFAR10		ID: CIFAR100	
	FPR95↓	AUROC↑	FPR95↓	AUROC↑
Baseline	33.44	89.01	64.25	81.52
OE [5]	20.16	93.74	57.68	83.98
OE <sup>†</sup> [5]	15.57	96.40	52.30	83.47
DFB	<b>7.87</b>	<b>98.04</b>	<b>42.29</b>	<b>91.03</b>

**Table 3: AUROC results using CAM and Grad-CAM for mask generation.**

ID: CIFAR100	CIFAR10	SVHN	Places365	Textures	Average
CAM	86.26	92.92	87.14	97.91	91.10
Grad-CAM	84.54	92.83	88.78	97.22	90.84

**Table 4: AUROC results for DBF under various  $\theta$  in mask generation.**

ID: CIFAR100	CIFAR10	SVHN	Places365	Textures	Average
$\theta = 0.3$	85.33	91.19	86.57	97.15	90.06
$\theta = 0.5$	86.46	92.92	87.14	97.91	91.10
$\theta = 0.7$	86.34	91.84	87.65	97.97	90.95

### C.3 DBF vs. Outlier Exposure

OOD features can also be alternatively learned by using the popular outlier exposure (OE) method [5] that uses external outlier data to support the learning. The comparison of DFB and OE is presented in Tab. 2, in which OE<sup>†</sup> uses the large-scale 80M Tiny Image [14] as the outlier dataset, OE is trained using Tiny ImageNet [9] as the outlier data, OE, OE<sup>†</sup> and DBF are all based on the MSP-based OOD scoring function, and Baseline is the original MSP without using outlier data. The results show that the two OE methods can also significantly outperform the Baseline model, but their performance is heavily dependent on the outlier data, e.g., the results of OE and OE<sup>†</sup> differ significantly from each other. By contrast, DFB does not need outlier data and significantly outperforms both OE methods on both benchmarks.

### C.4 Analysis of Pseudo Mask Quality

We investigate the impact of different pseudo-mask qualities on the final performance of DBF during the pseudo-mask generation phase. Experiments show that pseudo mask quality has limited impact on final performance. Particularly, Table 3 compares Grad-CAM[13] generated masks with CAM[17], showing similar performance. Table 4 analyzes the threshold  $\theta$  in mask generation. Results indicate that a mask that roughly distinguishes objects from the background is sufficient for DFB.

## REFERENCES

- [1] Jiwoon Ahn, Sunghyun Cho, and Suha Kwak. 2019. Weakly supervised learning of instance segmentation with inter-pixel relations. In *CVPR*. 2209–2218.
- [2] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. 2014. Describing textures in the wild. In *CVPR*. 3606–3613.
- [3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Identity mappings in deep residual networks. In *ECCV*. Springer, 630–645.
- [4] Dan Hendrycks and Kevin Gimpel. 2017. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. *ICLR* (2017).
- [5] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. 2019. Deep Anomaly Detection with Outlier Exposure. In *ICLR*. <https://openreview.net/forum?id=HyxCxhRcY7>
- [6] Rui Huang and Yixuan Li. 2021. Mos: Towards scaling out-of-distribution detection for large semantic space. In *CVPR*. 8710–8719.
- [7] Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Joan Puigcerver, Jessica Yung, Sylvain Gelly, and Neil Houlsby. 2020. Big transfer (bit): General visual representation learning. In *ECCV*. Springer, 491–507.
- [8] Alex Krizhevsky, Geoffrey Hinton, et al. 2009. Learning multiple layers of features from tiny images. (2009).
- [9] Ya Le and Xuan Yang. 2015. Tiny imagenet visual recognition challenge. *CS 231N* 7, 7 (2015), 3.
- [10] Shiyu Liang, Yixuan Li, and R. Srikant. 2018. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In *ICLR*. <https://openreview.net/forum?id=H1VGkIxRZ>
- [11] Weitang Liu, Xiaoyun Wang, John Owens, and Yixuan Li. 2020. Energy-based out-of-distribution detection. 33 (2020), 21464–21475.
- [12] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng. 2011. Reading digits in natural images with unsupervised feature learning. (2011).
- [13] Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. 2017. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *CVPR*. 618–626.
- [14] Antonio Torralba, Rob Fergus, and William T Freeman. 2008. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE TPAMI* 30, 11 (2008), 1958–1970.
- [15] Haoqi Wang, Zhizhong Li, Litong Feng, and Wayne Zhang. 2022. ViM: Out-Of-Distribution with Virtual-logit Matching. In *CVPR*. 4921–4930.
- [16] Hongyi Zhang, Moustapha Cisse, Yann N. Dauphin, and David Lopez-Paz. 2018. mixup: Beyond Empirical Risk Minimization. In *ICLR*. <https://openreview.net/forum?id=r1Ddp1-Rb>
- [17] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. 2016. Learning deep features for discriminative localization. In *CVPR*. 2921–2929.
- [18] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. 2017. Places: A 10 million image database for scene recognition. *IEEE TPAMI* 40, 6 (2017), 1452–1464.