Enhancing Out-of-Distribution Detection with Extended Logit Normalization Supplementary Material

Yifan Ding[1] Xixi Liu[2] Jonas Unger[1]
Gabriel Eilertsen[1]
[1]Linköping University [2]Imperial College London
[1]{yifan.ding, jonus.unger, gabriel.eilertsen}@liu.se [2]x.liu2@imperial.ac.uk

1. Proof of Proposition 3.1

Using singular value decomposition, we express the weight matrix as

$$\mathbf{W}^\top \mathbf{z} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^\top \mathbf{z},$$

where $\mathbf{U}$ and $\mathbf{V}$ are orthonormal matrices, and $\mathbf{\Sigma}$ is a diagonal matrix containing the singular values $\sigma_1, \ldots, \sigma_c$. Since orthonormal transformations preserve norms, it follows that

$$\|\mathbf{W}^\top \mathbf{z}\| = \|\mathbf{\Sigma}\mathbf{V}^\top \mathbf{z}\|.$$

Expanding this expression, we obtain

$$\|\mathbf{W}^\top \mathbf{z}\| = \sqrt{\sum_{i=1}^{d} \sigma_i^2 (z_i')^2},$$

where $\mathbf{z}' = \mathbf{V}^\top \mathbf{z}$ represents the transformed feature vector. Applying the extremal singular values, we establish the following bound:

$$\sigma_{\min}\|\mathbf{z}\| \leq \|\mathbf{W}^\top \mathbf{z}\| \leq \sigma_{\max}\|\mathbf{z}\|.$$

Including the bias term $\mathbf{b}$, we obtain

$$\sigma_{\min}\|\mathbf{z}\| - \|\mathbf{b}\| \leq \|\mathbf{f}\| \leq \sigma_{\max}\|\mathbf{z}\| + \|\mathbf{b}\|.$$

If the singular values cluster around their mean $\bar{\sigma}$, we approximate

$$\|\mathbf{f}\| \approx \bar{\sigma}\|\mathbf{z}\| + \eta,$$

where η represents noise due to $\mathbf{b}$ and singular value variations. In the special case where $\|\mathbf{b}\| = 0$ and $\mathbf{W}$ is isotropic (i.e., all singular values are identical), strict proportionality holds:

$$\|\mathbf{f}\| = \bar{\sigma}\|\mathbf{z}\|.$$

Thus, Proposition 3.1 is proved.

2. Proof of Proposition 3.2

The decision boundary between class $z_{\max}$ and class i is defined as

$$H_{i\ z_{\max}} = \{\mathbf{z} \mid (\mathbf{w}_{z_{\max}} - \mathbf{w}_i)^\top \mathbf{z} + (b_{z_{\max}} - b_i) = 0\}, \quad \forall i \neq z_{\max}.$$

If $m \geq c-1$, there exists at least one solution, ensuring that $\mathcal{D}_{\min}(\mathbf{z}) = 0$. Their intersection forms an affine subspace of dimension

$$\dim\left(\bigcap_{i \neq z_{\max}} H_{i\ z_{\max}}\right) = m - c + 1.$$

If $m < c-1$, the linear system is overdetermined and admits no exact solution. Instead, we solve the least-squares problem

$$\mathbf{A}\mathbf{z} = \mathbf{b},$$

where $\mathbf{A}$ is a $(c-1) \times m$ matrix whose rows are given by $(\mathbf{w}_{z_{\max}} - \mathbf{w}_i)^\top$, $\mathbf{b}$ is a $(c-1)$-dimensional vector with entries $b_{z_{\max}} - b_i$, and $\mathbf{z}$ is the unknown m-dimensional vector. The approximate solution is given by

$$\mathbf{z}^* = (\mathbf{A}^\top \mathbf{A})^{-1} \mathbf{A}^\top \mathbf{b}.$$

Assume $\mathbf{A}^\top \mathbf{A}$ (independent decision boundaries) is an invertible $m \times m$ matrix. Then the solution space of $\mathbf{z}^*$ is zero-dimensional, and

$$\mathcal{D}_{\min}(\mathbf{z}) = \|\mathbf{A}\mathbf{z}^* - \mathbf{b}\|_2 > 0.$$

Thus, Proposition 3.2 is proved.

3. Computation Overhead Analysis

Our method introduces an additional pairwise weight-distance computation with theoretical complexity $\mathcal{O}(C^2 D)$; the corresponding PyTorch implementation is shown in Fig. 1. Although this appears costly, the overhead is negligible in practice. The key reason is that the pairwise operation is applied only to the final linear classifier rather than the backbone. Even for ImageNet-1K, where $C = 1000$ and a typical classifier dimension is $D \approx 2048$, the resulting parameter tensor ($\sim$2M weights) is extremely small compared to the feature maps processed by modern CNN backbones. The entire pairwise norm computation is executed as a single dense, fully vectorized GPU kernel, requires no gradient backpropagation, and adds only a few milliseconds per iteration. Consequently, the overall training cost continues to be dominated by the forward–backward computation of the backbone network.

Empirically, we observe almost no additional wall-time when training on ImageNet-1K. For example, with ResNet-18 on ImageNet-1K, the extra classifier operation amounts to $C^2 D = 1000^2 \times 512 \approx 5.12 \times 10^8$ FLOPs (≈ 0.512 GFLOPs), compared to the ≈ 5.4 GFLOPs required for a forward–backward pass through the backbone—an overhead of only about 9–10%. On smaller-scale datasets, the overhead becomes even more negligible. For ResNet-18 on CIFAR-100 ($C = 100, D = 512$), the additional work is only $100^2 \times 512 = 5.12 \times 10^6$ FLOPs (≈ 0.005 GFLOPs), which is less than 1% of the ≈ 0.9 GFLOPs consumed by the forward–backward pass. Across all settings we tested, the end-to-end training time remains nearly identical to standard cross-entropy training despite the asymptotically higher complexity.

4. Extra Experiments

To further assess the generality and practical utility of the proposed ELogitNorm, we conduct additional experiments on both small-scale and large-scale OOD benchmarks. The results are summarized in Tables 1 and 2. Across all post-hoc detection methods evaluated, including MSP [1], ReAct [4], KNN [5], GEN [3], fDBD [2], and SCALE [7], their ELogitNorm-enhanced variants (denoted with "$*$") consistently outperform their standard counterparts on both near-OOD and far-OOD datasets.

Per-dataset results on CIFAR-100. Table 1 reports the per-dataset OOD performance when the in-distribution (ID) dataset is CIFAR-100 and the encoder is ResNet18. ELogitNorm provides substantial improvements in far-OOD scenarios (e.g., MNIST, SVHN, Textures, and Places365). In particular, MSP*, ReAct*, and GEN*exhibit clear gains in AUROC and considerable reductions in FPR, demonstrating that the distance-aware normalization effectively sharpens confidence transitions around decision boundaries. These improvements occur without any architectural modifications or additional training-time regularizers, confirming the plug-in nature of our method.

Per-dataset results on ImageNet-200. Table 2 presents analogous results when the ID dataset is ImageNet-200. Here the label space is significantly larger and the OOD shifts are more challenging. ELogitNorm again consistently boosts performance for all tested post-hoc detectors. For example, KNN*achieves strong performance on both iNaturalist and OpenImage-O, and SCALE*attains state-of-the-art far-OOD detection on ImageNet-200 benchmarks such as Textures and

```python
import torch
import torch.nn.functional as F

def elogitnorm_loss(logits, fc_weight, target):
    """
    ELogitNorm loss corresponding to the equations in Appendix.
    logits: (N, C)
    fc_weight: (C, D)
    target: (N,)
    """

    # Pairwise classifier-weight differences
    w_diff = fc_weight.unsqueeze(1) - fc_weight.unsqueeze(0)
    denom = torch.norm(w_diff, dim=2)
    denom.fill_diagonal_(1.0)

    # Maximum logit and predicted class
    values, nn_idx = logits.max(dim=1)

    # Logit gaps
    gaps = (logits - values.unsqueeze(1)).abs()

    # Instance-wise scaling factor
    scale = (gaps / denom[nn_idx]).mean(dim=1, keepdim=True)

    # ELogitNorm objective
    scaled_logits = logits / scale
    loss = F.cross_entropy(scaled_logits, target)

    return loss
```

Figure 1. PyTorch-style implementation of the proposed ELogitNorm training objective corresponding to Eqns. (7)–(8).

Table 1. *Per-dataset performance of OOD detection methods and their ELogitNorm-enhanced variants (denoted with $*$).* The image encoder is ResNet18. The ID dataset is **CIFAR-100**.

OOD Method	CIFAR-10		TIN		Near-OOD		MNIST		SVHN		Textures		Places365		Far-OOD	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
MSP [1]	78.47	58.91	82.07	50.70	80.27	54.80	76.08	57.23	78.42	59.07	77.32	61.88	79.22	56.62	77.76	58.70
MSP*	76.00	69.15	83.37	49.81	79.68	59.48	90.23	30.88	88.56	37.73	78.16	64.33	81.07	54.51	84.51	46.86
ReAct [4]	78.65	61.30	82.88	51.47	80.77	56.39	78.37	56.04	83.01	50.41	80.15	55.04	80.03	55.30	80.39	54.20
ReAct*	73.80	74.76	82.27	50.96	78.04	62.86	91.86	27.45	90.19	33.13	77.87	59.37	80.25	56.38	85.04	44.08
KNN [5]	77.02	72.80	83.34	49.65	80.18	61.22	82.36	48.58	84.15	51.75	83.66	53.56	79.43	60.70	82.40	53.65
KNN*	77.11	65.94	83.14	50.26	80.12	58.10	85.93	44.80	83.89	56.63	82.78	55.01	78.74	60.48	82.84	54.23
GEN [3]	79.38	58.87	83.25	49.98	81.31	54.42	78.29	53.92	81.41	55.45	78.74	61.23	80.28	56.25	79.68	56.71
GEN*	75.20	69.42	82.75	50.18	78.98	59.80	91.50	29.31	89.29	36.51	76.88	64.73	80.74	54.71	84.60	46.32
fDBD [2]	78.35	63.89	83.97	47.89	81.16	55.89	79.05	51.35	80.48	53.80	81.18	53.65	79.85	57.16	80.14	53.99
fDBD*	75.60	73.73	83.32	50.20	79.46	61.96	90.23	31.84	88.78	38.70	82.08	50.91	80.52	56.44	85.40	44.47
SCALE [7]	79.26	59.11	82.71	52.24	80.99	55.68	80.27	51.64	84.45	49.27	80.50	58.45	80.47	56.98	81.42	54.09
SCALE*	74.99	69.91	82.73	50.36	78.86	60.14	91.72	28.60	89.68	35.58	77.46	64.02	80.82	54.56	84.92	45.69

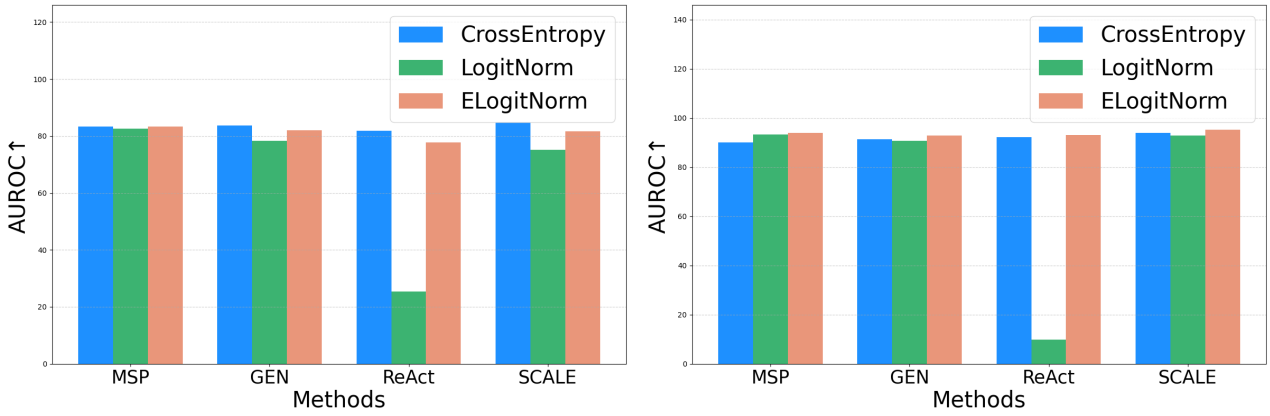
OpenImage-O. These results indicate that ELogitNorm scales favorably with the number of classes and benefits detectors that rely on logits, features, or activation statistics alike.

Max-logit visualization. Fig 3 visualizes the maximal logit over a 2D feature space for Cross-Entropy, LogitNorm, and ELogitNorm. Under this controlled setting, ELogitNorm produces smoother and more structured max-logit landscapes compared to the baselines. This geometric refinement translates into clearer decision regions and better separation between high- and low-confidence areas, which in turn supports more reliable OOD detection.

Training stability analysis. Fig 4 compares the training dynamics of ResNet18 on CIFAR-10 and CIFAR-100 using Cross-Entropy, LogitNorm [6], and **ELogitNorm** (Ours). Across both datasets, ELogitNorm exhibits stable convergence behavior that closely matches the baselines, without introducing additional oscillations or slowdowns. On CIFAR-10, all three methods

Table 2. *Per-dataset performance of OOD detection methods and their ELogitNorm-enhanced variants (denoted with $*$).* The image encoder is ResNet18. The ID dataset is **ImageNet-200**.

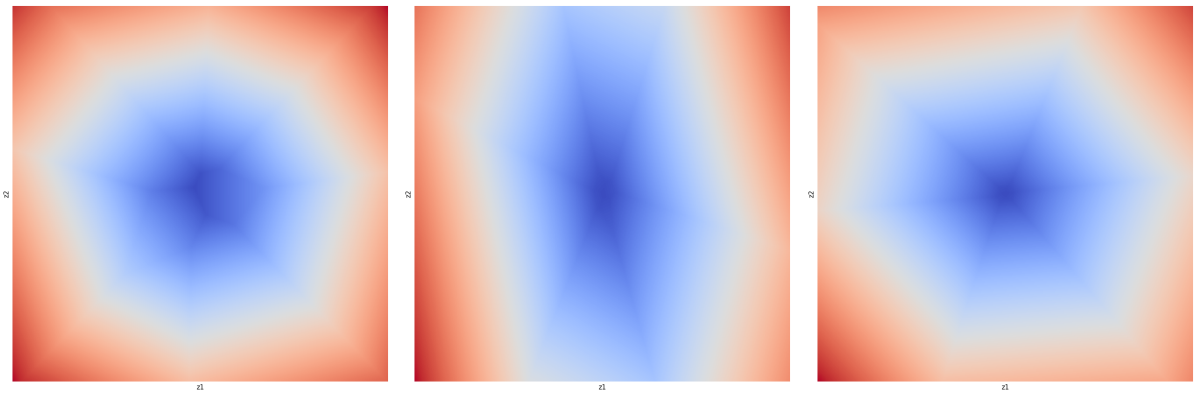
OOD Method	**SSB-hard**		**NINCO**		**Near-OOD**		**iNaturalist**		**Textures**		**OpenImage-O**		**Far-OOD**	
	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR	AUROC	FPR
MSP [1]	80.38	66.00	86.29	43.65	83.34	54.82	92.80	26.48	88.36	44.58	89.24	35.23	90.13	35.43
MSP*	79.00	66.01	87.12	43.96	83.06	54.99	96.50	14.75	92.56	31.81	91.70	28.68	93.58	25.08
ReAct [4]	78.97	71.51	84.76	53.47	81.87	62.49	93.65	22.97	92.86	29.67	90.40	32.86	92.31	28.50
ReAct*	73.24	75.78	83.87	54.66	78.56	65.22	96.27	16.89	92.91	32.55	90.10	35.97	93.10	28.47
KNN [5]	77.03	73.71	86.10	46.64	81.57	60.18	93.99	24.46	95.29	24.45	90.19	32.90	93.16	27.27
KNN*	78.18	69.64	87.27	43.66	82.73	56.65	97.17	12.86	97.00	16.79	94.06	24.47	96.08	18.04
GEN [3]	80.75	66.79	86.60	43.61	83.68	55.20	93.70	22.03	90.25	42.01	90.13	32.25	91.36	32.10
GEN*	77.43	66.30	86.03	46.60	81.73	56.45	96.26	16.39	91.62	34.79	90.56	31.68	92.81	27.62
fDBD [2]	78.62	72.42	87.78	42.11	83.20	57.27	96.84	14.19	93.95	26.37	92.56	26.59	94.45	22.38
fDBD*	78.85	72.39	87.81	42.39	83.33	57.39	96.80	14.03	93.94	26.36	92.61	26.31	94.45	22.23
SCALE [7]	82.08	67.39	87.60	47.20	84.84	57.29	95.79	18.41	94.11	29.75	92.04	31.21	93.98	26.46
SCALE*	75.73	71.33	85.95	48.86	80.84	60.10	97.09	13.64	95.85	20.70	92.68	28.88	95.21	21.08



(a) ImageNet-200, near-OOD

(b) ImageNet-200, far-OOD

Figure 2. *Average near-OOD and far-OOD performance with four post-hoc methods MSP [1], ReAct [4], GEN [3] and SCALE [7].* ResNet18 models are trained with Cross-Entropy, LogitNorm [6], and **ELogitNorm** (Ours), respectively. ID data is ImageNet-200.

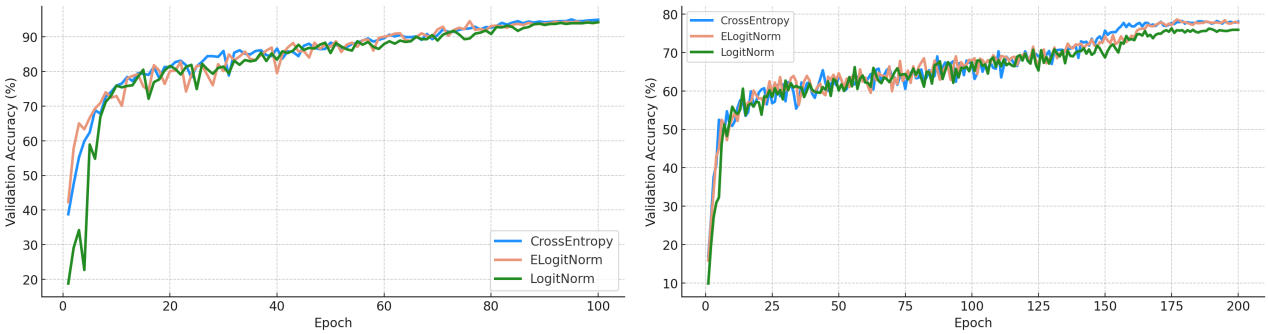


(a) Cross-Entropy

(b) LogitNorm

(c) ELogitNorm

Figure 3. Max-logit map on a 2D feature space, with the same setting as in Fig. 2(b). A ResNet18 is trained on CIFAR-10 and the feature is set to $\mathbf{z} \in \mathbb{R}^2$ before the penultimate layer.



(a) CIFAR-10 (b) CIFAR-100

Figure 4. *Training stability.* ResNet18 models are trained on CIFAR-10 and CIFAR-100 with Cross-Entropy, LogitNorm [6], and **ELogitNorm** (Ours), respectively.

converge rapidly to similar final validation accuracy, with ELogitNorm showing smooth optimization despite the adaptive scaling applied to logits. On CIFAR-100, where optimization is typically more sensitive, ELogitNorm again tracks the trajectories of Cross-Entropy and LogitNorm, demonstrating reliable and consistent learning throughout the entire 200-epoch schedule. These results confirm that the instance-wise distance-aware normalization does not adversely affect the optimization landscape. Instead, ELogitNorm preserves the training characteristics of standard cross-entropy while offering significant improvements in OOD robustness, validating that the proposed normalization is both effective and practical to deploy in real training pipelines.

References

[1] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. In *ICLR*, 2017. 2, 3, 4

[2] Litian Liu and Yao Qin. Fast decision boundary based out-of-distribution detector. *arXiv preprint arXiv:2312.11536*, 2023. 2, 3, 4

[3] Xixi Liu, Yaroslava Lochman, and Christopher Zach. Gen: Pushing the limits of softmax-based out-of-distribution detection. In *CVPR*, 2023. 2, 3, 4

[4] Yiyou Sun, Chuan Guo, and Yixuan Li. React: Out-of-distribution detection with rectified activations. In *NeurIPS*, 2021. 2, 3, 4

[5] Yiyou Sun, Yifei Ming, Xiaojin Zhu, and Yixuan Li. Out-of-distribution detection with deep nearest neighbors. In *ICML*, 2022. 2, 3, 4

[6] Hongxin Wei, Renchunzi Xie, Hao Cheng, Lei Feng, Bo An, and Yixuan Li. Mitigating neural network overconfidence with logit normalization. In *ICML*, 2022. 3, 4, 5

[7] Kai Xu, Rongyu Chen, Gianni Franchi, and Angela Yao. Scaling for training time and post-hoc out-of-distribution detection enhancement. In *ICLR*, 2024. 2, 3, 4