Shifting Transformation Learning for Out-of-Distribution Detection

Sina Mohseni
smohseni@nvidia.com
Arash Vahdat
avahdat@nvidia.com
Jay Yadawa
jyadawa@nvidia.com

NVIDIA
2788 San Tomas Expy,
Santa Clara, CA

Abstract

Detecting out-of-distribution (OOD) samples plays a key role in open-world and safety-critical applications such as autonomous systems and healthcare. Recently, self-supervised representation learning techniques have shown to be effective in improving OOD detection. However, one major issue with such approaches is the choice of shifting transformations and pretext tasks which depends on the in-domain distribution. In this paper, we propose a simple framework that selects optimal shifting transformations and pretext tasks and modulating their effect on representation learning without requiring any OOD training samples. In extensive experiments, we show that our simple framework outperforms state-of-the-art OOD detection models on several image datasets. We also characterize the criteria for a desirable OOD detector for real-world applications and demonstrate the efficacy of our proposed technique against state-of-the-art OOD detection techniques.

1 Introduction

Despite advances in representation learning and their generalization to unseen samples, learning algorithms are bounded to perform well on source distribution and vulnerable to out-of-distribution (OOD) or outlier samples. For example, it has been shown that the piece-wise linear decision boundaries in deep neural network (DNN) with ReLU activation are prone to OOD samples as they can assign arbitrary high confidence values to samples away from the training distribution [13]. Recent work on machine learning trustworthiness and safety have shown that OOD detection plays a key role in open-world and safety-critical applications such as autonomous systems [26] and healthcare [31]. However, OOD detection in high dimensional domains like image data is a challenging task and often requires great computational resource [7].

The recent surge in self-supervised learning techniques shows that learning pretext tasks can result in better semantic understanding of data by learning invariant representations [6] and can increase model performance in different setups [8]. Self-supervised learning has also been shown effective in OOD detection. For example, Golan and El-Yaniv [9] and Hendrycks

et al. [18] show that simple geometric transformations improve OOD detection performance, and Tack et al. [37] leverage shifting data transformations and contrastive learning for OOD detection. However, these works manually design the transformations and pretext tasks.

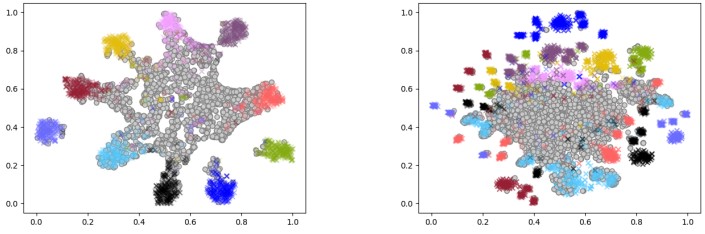
Inspired by the recent works, we study the impact of representation learning on OOD detection when training a model on artificially transformed datasets. We observe that training on a diverse set of dataset transformations jointly, termed as *shifting transformation learning* here, further improves the model's ability to distinguish in-domain samples from outliers. However, we also empirically observe that the choice of effective data transformations for OOD detection depends on the in-domain training set. That is to say, the set of transformations effective for one in-domain dataset may not be effective for another dataset.

To address this problem, we make the following contributions in this paper: **(i)** We propose a simple framework for selecting and modulating effects of training set transformations (shifted views of the in-domain training set) to improve OOD detection. We demonstrate that the optimally selected transformations result in better representations for both main classification and OOD detection compared to data augmentation-based approaches. **(ii)** We propose an ensemble score for OOD detection that leverages multiple transformations trained with a shared encoder. In particular, our technique achieves new state-of-the-art results in OOD detection on multi-class classification by improving averaged area under the receiver operating characteristics (AUROC) +1.3% for CIFAR-10, +4.37% for CIFAR-100, and +1.02% for ImageNet-30 datasets. **(iii)** To the best of our knowledge, this paper is the first to introduce criteria for ideal OOD detection and to analyze a diverse range of techniques along with these criteria. Albeit the simplicity, we show that our proposed approach outperforms the state-of-the-art techniques on robustness and generalization criteria.

2 Related Work

Here, we review OOD detection methods related to this work:

Distance-based Detection: Distance-based methods use different distance measures between the unknown test sample and source training set in the representation space. These techniques involve preprocessing or test-time sampling of the source domain distribution to measure their averaged distance to the novel input sample. The popular distance measures include Mahalanobis distance [23, 35], cosine similarity [37, 38] and others semantic similarity metrics [30]. These techniques usually work well with unlabeled data in unsupervised and one-class classification setups. For example, Ruff et al. [32] present a deep learning one-class classification approach to minimize the representation hypersphere for source distribution and calculate the detection score as the distance of the outlier sample to the center of the hypersphere. Recently, Mukhoti et al. [28] proposed using distance measures for model features to better disentangle model uncertainty from dataset uncertainty. Distance-based methods can benefit from ensemble measurements over input augmentations [37] or transformations [2], network layers [23, 34], or source domain sub-distributions [29] to improve detection results. For instance, Tack et al. [37] present a detection score based on combining representation norm with cosine similarity between the outlier samples and their nearest training samples for one-class classification problem. They also show that OOD detection can be improved with ensembling over random augmentations, which carries a higher computational cost.



(a) (b)

Figure 1: The t-SNE visualization of the penultimate layer features in a ResNet-18 network trained on CIFAR-10 using (a) supervised learning with cross-entropy loss and (b) our method with shifting transformation learning. OOD samples [41] are presented in gray.

Classification-based Detection: These OOD detection techniques avoid costly distance-based and uncertainty estimation techniques (e.g., Gal and Ghahramani [7]) by seeking effective representation learning to encode normality together with the main classification task. Various detection scores have been proposed including maximum softmax probability [15], maximum logit scores [17], prediction entropy [27], and KL-divergence score [18]. To improve the detection performance, [19, 22] proposed a combination of temperature scaling and adversarial perturbation of input samples to calibrate the model to increase the gap between softmax confidence for the inlier and outlier samples. Another line of research proposed using auxiliary unlabeled and disjoint OOD training set to improve OOD detection for efficient OOD detection without architectural changes [16, 27].

Recent work on self-supervised learning shows that adopting pretext tasks results in learning more invariant representations and better semantic understanding of data [6] and which significantly improves OOD detection [9]. Hendrycks et al. [18] extended self-supervised techniques with a combination of geometric transformation prediction tasks. Self-supervised contrastive training [3] is also shown to be effective to leverage from multiple random transformations to learn in-domain invariances, resulting in better OOD detection [35, 37, 40].

3 Method

In this paper, we propose a framework for training with shifting transformations to increase a network's sensitivity to outlier features and to improve its OOD detection performance. Intuitively, we simultaneously train a base encoder on *multiple shifting transformations* of the training set using auxiliary self-supervised objectives (for unlabeled datasets) and fully-supervised objectives (for labeled datasets).

3.1 Shifting Transformation Learning

Our transformation learning technique trains a multi-tasked network using self-supervised and fully-supervised training objectives. We consider a set of geometric (translation, rotation) and non-geometric (blurring, sharpening, color jittering, Gaussian noise, cutout) shifting transformations and we train the network with dedicated loss functions for each transformation. For

the self-supervised transformation learning, given an unlabeled training set of $\mathcal{S} = \{(x_i)\}_{i=1}^{M}$, we denote the set of domain invariant transformations T_n by $\mathcal{T} = \{T_n\}_{n=1}^{N}$. We generate a self-labeled training set $\mathcal{S}_{T_n} = \{(T_n(x_i), \hat{y}_i)\}_{i=1}^{M}$ for each self-supervised transformation T_n where $\hat{y}_i$ are the transformation labels. For example, we consider the image rotation task with four levels of $\{0°, 90°, 180°, 270°\}$ self-labeled rotations and $\hat{y}_i \in \{0,1,2,3\}$ in this case. The self-supervised loss $\mathcal{L}_{\text{ssl}}$ is the weighted average of loss across all transformations in $\mathcal{T}$:

$$\mathcal{L}_{\text{ssl}}(\lambda, \theta) = \frac{1}{N} \sum_{n=1}^{N} \lambda_n \sum_{(T_n(x_i), \hat{y}_i) \in S_{T_n}} \ell(f_\theta^{(n)}(T_n(x_i)), \hat{y}_i), \tag{1}$$

where $f_\theta^{(n)}$ is a classification network with parameters θ for the n^{th} task, $\lambda = \{\lambda_n\}_{n=1}^{N}$ are transformation weights, and ℓ is the multi-class cross-entropy loss. The objective above trains the network with self-supervised labels only. When class labels are available, given the labeled training set of $\mathcal{S} = \{(x_i, y_i)\}_{i=1}^{M}$, we generate transformed copies of the original training sets $\mathcal{S}'_{T_n} = \{(T_n(x_i), y_i)\}_{i=1}^{M}$ where training samples retain their original class labels. The supervised loss $\mathcal{L}_{\text{sup}}$ is defined by:

$$\mathcal{L}_{\text{sup}}(\lambda', \theta) = \frac{1}{N} \sum_{n=1}^{N} \lambda'_n \sum_{(T_n(x_i), y_i) \in S'_{T_n}} \ell(f_\theta^{\prime(n)}(T_n(x_i)), y_i), \tag{2}$$

which measures the classification loss for transformed copies of the data with $\lambda' = \{\lambda'_n\}_{n=1}^{N}$ as transformation coefficients in $\mathcal{L}_{\text{sup}}$. In labeled setup, we combine $\mathcal{L}_{\text{ssl}}$ and $\mathcal{L}_{\text{sup}}$ with the main supervised learning loss $\mathcal{L}_{\text{main}}$ (e.g., the cross-entropy loss for classifying the in-domain training set):

$$\mathcal{L}_{\text{total}}(\lambda, \lambda', \theta) = \mathcal{L}_{\text{main}}(\theta) + \mathcal{L}_{\text{ssl}}(\lambda, \theta) + \mathcal{L}_{\text{sup}}(\lambda', \theta) \tag{3}$$

In all unlabeled detection setups, we define $\mathcal{L}_{\text{total.}} := \mathcal{L}_{\text{ssl}}$ and discard the main classification task. In the rest of the paper, for the ease of notation, we use λ to refer to all the coefficients $\{\lambda, \lambda'\}$, and we drop λ, θ when it is clear from the context.

Instead of training a separate network $f_\theta^{(n)}$ or $f_\theta^{\prime(n)}$ for each task, all the auxiliary tasks and the main classification task share a feature extraction network and each only introduces an additional 2-layer fully-connected head for each task. Training is done in a multi-task fashion in which the network is simultaneously trained for the main classification (if applicable) and all weighted auxiliary tasks using standard cross-entropy loss.

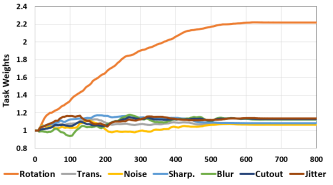
Visualization: To illustrate the impact of our training loss functions on OOD detection, Figure 1 shows t-SNE visualization [39] of the CIFAR-10 examples obtained from ResNet-18 [12] trained with the cross-entropy loss (left) compared to our multitask transformation learning (right). The visualization shows using shifted in-domain samples during training increases in-domain features' distribution, resulting in improved separation between out-domain (in gray) and in-domain samples (in colors) at the test time. Note that the proposed approach does not need additional OOD training samples, unlike previous work [16, 27].

3.2 Learning to Select Optimal Transformations

Previous work on self-supervised learning used ad-hoc heuristics for choosing data transformations for the training set [9, 18, 37]. However, the optimal choice of effective transformations depends on the source distribution and heuristic approaches cannot scale up to diverse training distributions when there are many potential transformations. To illustrate this, we train a

Rot.	92.18	92.55	92.25	90.65	91.47	91.43	92.48
Trans.	92.55	90.16	90.65	90.62	90.54	89.95	90.57
Noise	92.25	90.65	90.08	90.24	90.41	89.99	90.35
Jitter	90.65	90.62	90.24	89.63	89.84	90.07	90.23
Sharp.	91.47	90.54	90.41	89.84	89.58	89.40	90.14
Cutout	91.43	89.95	89.99	90.07	89.40	89.93	89.92
Blur	92.48	90.57	90.35	90.23	90.14	89.92	90.50
	Rot.	Trans.	Noise	Jitter	Sharp.	Cutout	Blur

(a)



(b)

Figure 2: Our studies show that the optimal transformation set $\mathcal{T}$ and their weights λ depend on the in-domain training set. **a)** An ablation study to measure effects of individual and paired transformations on OOD detection performance. **b)** Optimizing transformation weights (λ) for auxiliary self-supervised tasks for each training set. Experiments are done in multi-class classification setup on different training sets.

ResNet-18 [12] with one or two self-supervised transformations that are selected from a pool of seven transformations. Here, we use the training objective in Eq. 1 with equal weights for all transformations. The OOD detection results are reported in Figure 2 (a) with CIFAR-10 dataset as in-distribution and CIFAR-100 as OOD test sets. The heatmap visualization presents a clear view of how different transformations (and the combinations of two) have a different impact on the OOD detection performance depending on the source distribution. Figure 3 in Appendix B.1 presents additional results with CIFAR-100 and ImageNet-30 [18] datasets as training sets. For example, although rotation is the most effective transformation on CIFAR-10 and ImageNet-30, it is among the least effective ones for CIFAR-100. On the other hand, sharpening and color jittering are among the most effective transformations for CIFAR-100, but they perform worse on CIFAR-10.

To tackle the problem of selecting optimal transformations, we propose a simple two-step transformation selection framework presented in Alg. 1. Our approach relies on Bayesian optimization to first select an effective transformation set $\mathcal{T}$. It then uses meta-learning to learn λ for OOD detection, as discussed next.

Optimizing Transformations Set $\mathcal{T}$: We use Bayesian optimization to identify effective transformations for each in-domain training set as the first step shown in Alg. 1. Here, we assume that transformation weights λ are equal to one and we only search for effective transformations set from a pool of available transformations. Due to the small $\mathcal{T}$ search space (i.e., 2^n for n transformations), we use a low-cost Bayesian optimization [1] with Tree-Parzen estimators to find the optimum self-supervised task set. The Bayesian optimization objective seeks to minimize the main classification loss $\mathcal{L}_{\text{main}}$ on D^{in}_{val}, the validation set for the in-domain training data.

Optimizing Transformations Weights λ: Next, we optimize λ coefficients for the selected transformation from the previous step to improve the effect of shifting transformation on representation learning. This step is important because the λ coefficients modulate the impact of different transformations in the training objective in Eq. 1 and Eq. 2. Here, we assume that λ is a "meta-parameter" and we use a differentiable hyperparameter optimization algorithm [25] for optimizing it as the second step shown in Alg. 1. Our optimization algorithm follows a bi-level optimization setting. The inner training updates train network parameters θ using $\mathcal{L}_{\text{total}}$ on D^{in}_{train} for K steps. Given the current state of parameters θ, we update λ in the outer loop such that $\mathcal{L}_{\text{main}}(\theta)$ is minimized on D^{in}_{val}. Note that the gradient of $\mathcal{L}_{\text{main}}(\theta)$ w.r.t. λ is defined only through the gradient updates in the inner loop. Thus, the λ updates in the

Algorithm 1 Transformations $\mathcal{T}$ and λ Optimization

Input: Available transformation set $\mathcal{T}$, learning rate α, β, inner steps K
Output: Optimal $\mathcal{T}_{opt}$ and λ_{opt} sets
Step 1: Transformations Selection

while not converged **do**
  Sample a new $\mathcal{T}$ set with $\lambda = 1$.
  Train a classifier with $\mathcal{L}_{\text{total}}$ loss.
  Calculate $\mathcal{L}_{\text{main}}$ on D^{in}_{val} as fitness measure.
  Update the acquisition function.
end while

Step 2: λ Weights Optimization

Initialize with $\lambda = 1$.
while not converged **do**
  for K steps **do**
    $\theta = \theta - \alpha \nabla_\theta \mathcal{L}_{\text{total}}(\lambda, \theta)$ on D^{in}_{train}
  end for
  $\lambda = \lambda - \beta \nabla_\lambda \mathcal{L}_{\text{main}}(\theta)$ on D^{in}_{val}
end while

outer loop require backpropagating through the gradients updates in the inner loop which can be done easily using differentiable optimizers [11]. We use $K = 1$ step for the inner-loop optimization with SGD when updating θ and we use Adam [21] to update λ with small learning rate β, set to 0.01. Figure 2 (b) presents λ values during optimization from a study on three training sets.

Because the choice of effective shifting transformations depends on the in-domain distribution, our optimization framework avoids the need for D^{out}_{test} samples and only relies on in-domain validation loss as a proxy for representation learning. Our ablation studies show that multi-task training of shifting transformations with this objective function is an effective proxy for selecting optimal transformations for both OOD detection and in-domain generalization.

3.3 OOD Detection Scores

In multi-class detection, we consider two ways for computing the detection score: (i) since all *supervised heads* are trained on the same task, we get the λ weighted sum of the softmax predictions from the main task and all auxiliary supervised transformation heads to compute an *ensemble score*. (ii) Alternatively, to reduce the test-time computational complexity, a faster detection score can be computed using only the main classification head. Given softmax scores obtained from either (i) or (ii), in all experiments we use KL-divergence loss between the softmax scores and uniform distribution (similar to [18]) as the OOD detection score:

$$score = \sum_{n=1}^{N} KL[U \| f^{(n)}(T_n(x))] \tag{4}$$

where U represents a uniform distribution, $f^{(n)}(T_n(x))$ is the prediction probability from each head. In unlabeled and one-class detection with only self-supervised heads, we first get the KL-divergence between each auxiliary head and its self-labeled targets, then calculate the final ensemble score using λ weighted sum of these scores from all auxiliary heads.

4 Experiments and Results

We run our main experiments on ResNet-18 [12] network to have a fair comparison with state-of-the-art. We used 7 different image transformations to define self-labeled prediction tasks in our proposed multi-task training. We used 4 levels of distortions in each semantic-preserving transformation including rotation of $\{0°, 90°, 180°, 270°\}$ degrees, translation in combinations of ± 30 % horizontal and vertical steps, Gaussian noise with standard deviations of {0, 0.3, 0.5, 0.8}, Gaussian blur with sigmas of {0, 0.3, 0.5, 0.8}, rectangular cutout with sizes of {0, 0.3, 0.5, 0.8}, sharpening via image blending with convolution-based edges with alphas of {0, 0.3, 0.6, 1.0}, and color distortion with jittered brightness, contrast, saturation, and hue by rates of {0, 0.4, 0.6, 0.8}.

In all experiments, both transformations set $\mathcal{T}$ and their training weights λ are optimized using the proposed framework with the final $(\mathcal{T}, \lambda)$ sets as follows: for the CIFAR-10 dataset {(Jitter, 0.8044), (Rotation, 0.6758), (Sharpening, 0.6601)}; for the CIFAR-100 dataset: {(blur, 0.4974), (Jitter, 0.2612), (Translate, 0.3424), (Sharpening, 0.4579)}; and for the ImageNet-30: {(Noise, 0.5748), (Rotation, 0.3606), (Sharpening, 0.5088)}.

Unless mentioned otherwise, our main evaluation results are based on the ensembled score from available auxiliary heads.

Ablations Studies: A set of ablations studies are presented in Appendix B.1 that examine and quantify the (i) effectiveness of the proposed transformation optimization, (ii) advantage of ensemble detection score, and (iii) OOD detection gain against data augmentation techniques. Our ablation studies indicate (i) the significant effect of transformation set $\mathcal{T}$ selection and their training weights λ optimization, (ii) considerable improvement in detection performance with our ensembled detection score (+1.84% in CIFAR-10, +5.04% in CIFAR-100, and +4.09% in ImageNet-30 datasets), and (iii) substantial OOD detection gain over RandAugment [5] and AutoAugment [4] augmentation techniques in CIFAR-10 (+2.95%) and CIFAR-100 (+5.67%) datasets.

4.1 Comparison to State-of-the-Arts

4.1.1 Multi-class Classification

Table 1 presents our main evaluation results for multi-class classification training with Eq. 4 on CIFAR-10, CIFAR-100, and ImageNet-30 [18] datasets each with six disjoint D_{test}^{out} sets with details provided in Appendix A. We compare our technique with the full supervised Baseline [15] and current state-of-the-art methods including self-supervised learning (Geometric) [18], supervised contrastive learning (SupSimCLR) [20] and SSD [35], and contrasting shifted instances (CSI) [37] and with its ensembled version (CSI-ens). All techniques are trained on ResNet-18 network with an equal training budget, and all except SSD+ use their softmax prediction as OOD detection score in multi-class classification. We compared the impact of both $\mathcal{L}_{\text{ssl}}$ and $\mathcal{L}_{\text{sup}} + \mathcal{L}_{\text{ssl}}$ training loss functions on OOD performance (in addition to $\mathcal{L}_{\text{main}}$ for the main classification task which is used by all the techniques in Table 1). Results show our approach outperforms previous works with a large margin with both $\mathcal{L}_{\text{sup}}$ and $\mathcal{L}_{\text{ssl}}$ training objectives. The averaged standard deviation for detection AUROC over six test sets from 5 runs of our techniques shows 0.13% for CIFAR-10, 0.33% for CIFAR-100, and 0.18% for ImageNet-30.

Moreover, Table 1 shows that training on the optimized transformation set $\mathcal{T}$ and λ weights only using an in-domain validation set consistently outperforms the previous work

D_{train}^{in}	D_{test}^{out}	Detection AUROC						
		Baseline	Geometric	SupSimCLR	SSD+	CSI (ens)	Ours ($\mathcal{L}_{\text{ssl}}$)	Ours ($\mathcal{L}_{\text{ssl}} + \mathcal{L}_{\text{sup}}$)
CIFAR-10	SVHN	92.89	97.96	97.22	93.80	96.11 (97.38)	**99.92**	96.60
	Texture	87.69	96.25	94.21	94.05	95.92 (97.18)	**97.61**	96.91
	Places365	88.34	92.57	91.11	91.77	92.21 (93.11)	93.72	**98.73**
	TinyImageNet	87.44	92.06	92.10	90.28	91.33 (92.49)	92.99	**93.57**
	LSUN	89.87	93.57	92.13	94.40	92.91 (94.02)	**95.03**	94.12
	CIFAR-100	87.62	91.91	88.36	90.40	90.60 (92.06)	93.24	**94.07**
	Average	88.98	94.05	92.52	92.45	93.18 (94.37)	95.42	**95.67**
CIFAR-100	SVHN	79.18	83.62	81.55	83.60	79.22 (87.38)	87.11	**90.64**
	Texture	75.28	82.39	76.83	81.35	78.33 (78.31)	**85.47**	77.99
	Places365	76.07	74.57	75.37	79.16	77.15 (78.1)	77.87	**92.62**
	TinyImageNet	78.53	77.56	80.77	76.29	80.07 (**82.41**)	80.66	79.25
	LSUN	73.73	71.86	73.50	63.77	74.89 (**75.22**)	74.32	74.01
	CIFAR-10	78.26	74.73	73.28	73.94	75.98 (78.44)	79.25	**91.56**
	Average	76.84	77.46	76.88	76.35	77.61 (79.98)	80.78	**84.35**
ImageNet-30	Flowers 101	87.70	92.13	93.81	96.47	95.43 (96.18)	94.19	**97.18**
	CUB-200	85.26	90.58	89.19	**96.57**	93.32 (94.15)	93.34	96.44
	Dogs	90.30	93.25	95.16	95.23	96.43 (**97.64**)	93.63	97.07
	Food	78.93	85.09	83.61	85.48	88.48 (89.04)	82.51	**96.49**
	Pets	92.88	95.28	96.38	96.24	97.35 (**98.49**)	94.82	96.37
	Texture	86.98	92.16	98.70	94.86	97.63 (**98.54**)	93.99	96.56
	Average	87.01	91.42	92.81	94.14	94.77 (95.67)	92.08	**96.69**

Table 1: Comparison of OOD detection results (AUROC %) with the supervised Baseline, state-of-the-art self-supervised [18], contrastive learning [20, 35, 37] and our technique with multi-task self-supervised ($\mathcal{L}_{\text{ssl}}$) and hybrid ($\mathcal{L}_{\text{ssl}} + \mathcal{L}_{\text{sup}}$) transformation learning tasks.

when testing on diverse D_{test}^{out} sets. This observation highlights the dependency of the optimal set of shifting transformations on the in-domain training set as opposed to prior work that manually selected the shifting transformation. In fact, we observe that all prior work based on rotation transformation perform worse than the supervised Baseline on the CIFAR-100 experiment when testing with CIFAR-10 as the D_{test}^{out} with the exception of CSI-ens.

4.1.2 Unlabeled Detection

Next, we test our technique for multi-class unlabeled and one-class OOD detection trained with the $\mathcal{L}_{\text{ssl}}$ loss (Eq. 1) using our proposed transformation optimization framework.

Table 5-a in Appendix B.2 presents results for unlabeled multi-class detection in which averaged detection AUROC over the six D_{test}^{out} sets is outperforming state-of-the-art methods with a large margin in unlabeled CIFAR-100 (83.95%), unlabeled ImageNet-30 (96.57%) datasets, with the exception of CSI-ens method showing better results in unlabeled CIFAR-10 (89.80%) dataset.

Table 5-b in Appendix B.2 shows detailed one-class classification results for each of the CIFAR-10 classes as D_{train}^{in} and the remaining classes as D_{test}^{out}. Our technique with 90.9% averaged AUROC on CIFAR-10 one-class detection outperforms previous works including DROCC [10], GOAD [2], Geometric, and SSD, with the exception of CSI-ens which requires a far more computationally expensive distance-based detection score.

5 OOD Detection Generalizability

We characterize four main criteria required from a generalizable OOD detection technique, including i) zero-shot OOD training, ii) no hyperparameter dependency, and iii) generalization to various unseen OOD distributions and iv) robustness against test-time perturbations. In this section, we situate our proposed technique against a diverse range of state-of-the-art OOD detection techniques along these requirements. Table 2 presents results for training on CIFAR-10 and testing on six D^{out}_{test} sets used in Table 1. Note that this is not intended to be a ranking of different OOD detection techniques; instead, we aim to review trade-offs and limitations among different detection approaches.

Hyperparameters Dependency: While hyperparameter tuning for the training of in-domain samples is done using a held-out validation set, the hyperparameter disentanglement is a crucial property for OOD detection. Specifically, an ideal detector should not be sensitive to hyperparameters tied to the target outlier distribution. Table 2 divides different techniques w.r.t their dependency on detection hyperparameters into three levels of high, low, and no dependency. Techniques with high dependency like ODIN [24] and Mahalanobis [23] use a validation set of D^{out} for training, resulting in poor performance under unseen or diverse mixture of outlier distributions. Table 2 shows over 3% performance gap between the averaged detection performance on six D^{out}_{test} sets (Column 5) and detection performance under an equal mixture of the same test sets (Column 6) for these two detectors which indicates strong D^{out} hyperparameter dependency.

Techniques with low dependency do not use a subset of D^{out}_{test}, however, they depend on hyperparameters such as the choice of D^{out}_{train} set (e.g., Outlier Exposure [16]), or hand-crafted self-supervised tasks [9], [18], or data augmentation [37] that requires post training D^{out}_{test} for validation. These techniques can suffer significantly in settings in which the new source training set is invariant to previous hand-crafted self-supervised tasks and augmentations, as seen in Figure 2. On the other hand, techniques with no hyperparameters like Gram Matrices [34], SSD [35], and our proposed framework bears no hyperparameter dependency on the choices of in-domain or outlier distribution. Note that many techniques, like ours, use a λ training hyperparameter to balance training between in-domain classification and auxiliary tasks. However, in our case, these hyperparameters are tuned automatically without requiring OOD training samples.

Zero-shot Training: A previous trend in OOD detection techniques considered using a subset of the target D^{out}_{test} for model tuning (e.g., ODIN [24] and Mahalanobis [23]) or using an auxiliary D^{out}_{train} set as a part of model training (e.g., Outlier Exposure [16]). Although these techniques can achieve high detection performance with the right training set, having access to the specific D^{out}_{tune} for tuning or even any D^{out}_{train} for training the detector is not a realistic assumption in practical setups. An efficient proposal to use these techniques is to integrate them into zero-shot techniques as presented by [18, 34] when D^{out}_{train} is available or to benefit from taking semi-supervised or few-shot approaches as done by [33, 35].

Detection Generalizability: Recent work on OOD detection recognized the necessity of diverse D^{out}_{test} sets to evaluate the generalizability of OOD detection techniques [27, 34, 40]. Typically, near-OOD and far-OOD sets are chosen based on the semantic and appearance similarities between the in-domain and outlier distributions and, in some cases, measured by relevant similarity metrics (e.g., confusion log probability [40]). Following the previous

Detection Technique	OOD Detection Criteria			Averaged Detection Performance	Generalizability Tests		
	Hyp.-Para. Dependency	Generalizable	Zero Shot		Mixed Distribution	Far-OOD	Near-OOD
ODIN	High	–	–	91.15	88.10	96.70	85.80
Mahalanobis	High	–	–	95.35	92.24	99.10	88.51
Outlier Exposure	Low	✓	–	96.24	96.88	98.76	93.41
Geometric	Low	✓	✓	94.05	94.29	97.96	91.91
CSI-ens	Low	✓	✓	94.37	94.10	97.38	92.06
SSD	No	✓	✓	92.45	92.70	93.80	90.40
Gram Matrices	No	–	✓	94.17	95.08	99.50	79.01
Ours	No	✓	✓	95.67	95.55	96.60	94.07

Table 2: Review of OOD detection criteria, averaged detection performance, and generalizability to unseen OOD test distributions (AUROC %) for a diverse set of OOD detection techniques. We compare our technique with ODIN [24], Mahalanobis [23], Outlier Exposure [16], Geometric [18], CSI [37], and Gram [34].

works, we chose CIFAR-100 as the near-OOD test distribution and SVHN as the far-OOD test distribution for detectors trained on CIFAR-10. While Table 2 shows high performance on far-OOD for all techniques, Gram Matrices, Mahalanobis, and ODIN show 20.5%, 10.9%, and 10.6% detection performance drop for near-OOD distribution compared to the far-OOD test distribution, respectively. In comparison, our technique shows 2.53% performance gap between far-OOD and near-OOD test distributions.

Detection Robustness: Evaluating the effects of distribution shift on predictive uncertainty have been previously studied in [10, 36] for real-world application. In Appendix C, we investigate the effect of natural perturbations and corruptions proposed in [14] on OOD detection performance. We measure averaged OOD detection results for all 15 image distortions on 5 levels of intensity where both D_{test}^{in} and D_{test}^{out} are treated with the same distortion type and level. Figure 4 in Appendix C presents detailed OOD detection results in which all techniques show more performance drop at the higher levels of perturbation intensity. However, distance-based detectors (Figure 4-a) like Gram and Mahalanobis show significantly less performance drop (4.23% and 5.24% AUROC drop, respectively) compared to classification-based detectors (Figure 4-b) like Outlier Exposure and Geometric with over 14% AUROC drop. Our experiments indicate the advantage of distance-based detection methods in OOD detection under test-time input perturbations.

6 Conclusion

Developing reliable and trustworthy machine learning algorithms for open-world and safety-critical applications poses a great challenge. In this paper, we presented a simple framework for OOD detection that leverages representation learning with shifting data transformations, and we empirically demonstrated its efficacy on several image datasets. We showed that the optimal choice of shifting transformation depends on the in-domain training distribution and we propose a framework to automatically choose the optimal transformations for a given in-domain set without requiring any OOD training samples. Albeit its simplicity, our proposed method outperforms the state-of-the-art OOD detection techniques and exhibits strong generalization to different outlier distributions. A limitation of our work is longer training time and large memory requirement due to the large training batch size. Future work is focused on improving the efficiency and scalability of shifted transformation learning for larger datasets.

References

[1] Takuya Akiba, Shotaro Sano, Toshihiko Yanase, Takeru Ohta, and Masanori Koyama. Optuna: A next-generation hyperparameter optimization framework. In *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pages 2623–2631, 2019.

[2] Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020.

[3] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. *arXiv preprint arXiv:2002.05709*, 2020.

[4] Ekin D Cubuk, Barret Zoph, Dandelion Mane, Vijay Vasudevan, and Quoc V Le. Autoaugment: Learning augmentation strategies from data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 113–123, 2019.

[5] Ekin D Cubuk, Barret Zoph, Jonathon Shlens, and Quoc V Le. Randaugment: Practical automated data augmentation with a reduced search space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 702–703, 2020.

[6] Alexey Dosovitskiy, Jost Tobias Springenberg, Martin Riedmiller, and Thomas Brox. Discriminative unsupervised feature learning with convolutional neural networks. In *Advances in neural information processing systems*, pages 766–774, 2014.

[7] Yarin Gal and Zoubin Ghahramani. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *ICML*, pages 1050–1059, 2016.

[8] Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. *arXiv preprint arXiv:1803.07728*, 2018.

[9] Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *NIPS*, pages 9758–9769, 2018.

[10] Sachin Goyal, Aditi Raghunathan, Moksh Jain, Harsha Vardhan Simhadri, and Prateek Jain. DROCC: Deep robust one-class classification. In Hal Daumé III and Aarti Singh, editors, *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pages 3711–3721. PMLR, 13–18 Jul 2020.

[11] Edward Grefenstette, Brandon Amos, Denis Yarats, Phu Mon Htut, Artem Molchanov, Franziska Meier, Douwe Kiela, Kyunghyun Cho, and Soumith Chintala. Generalized inner loop meta-learning. *arXiv preprint arXiv:1910.01727*, 2019.

[12] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[13] Matthias Hein, Maksym Andriushchenko, and Julian Bitterwolf. Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In *CVPR*, 2019.

[14] Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. In *International Conference on Learning Representations*, 2019.

[15] Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

[16] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2018.

[17] Dan Hendrycks, Steven Basart, Mantas Mazeika, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. A benchmark for anomaly segmentation. *arXiv preprint arXiv:1911.11132*, 2019.

[18] Dan Hendrycks, Mantas Mazeika, Saurav Kadavath, and Dawn Song. Using self-supervised learning can improve model robustness and uncertainty. In *Advances in Neural Information Processing Systems*, pages 15663–15674, 2019.

[19] Yen-Chang Hsu, Yilin Shen, Hongxia Jin, and Zsolt Kira. Generalized odin: Detecting out-of-distribution image without learning from out-of-distribution data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10951–10960, 2020.

[20] Prannay Khosla, Piotr Teterwak, Chen Wang, Aaron Sarna, Yonglong Tian, Phillip Isola, Aaron Maschinot, Ce Liu, and Dilip Krishnan. Supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33, 2020.

[21] Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

[22] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *arXiv preprint arXiv:1711.09325*, 2017.

[23] Kimin Lee, Kibok Lee, Honglak Lee, and Jinwoo Shin. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *Advances in Neural Information Processing Systems*, pages 7167–7177, 2018.

[24] Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

[25] Dougal Maclaurin, David Duvenaud, and Ryan Adams. Gradient-based hyperparameter optimization through reversible learning. In *International conference on machine learning*, pages 2113–2122. PMLR, 2015.

[26] Sina Mohseni, Mandar Pitale, Vasu Singh, and Zhangyang Wang. Practical solutions for machine learning safety in autonomous vehicles. *arXiv preprint arXiv:1912.09630*, 2019.

[27] Sina Mohseni, Mandar Pitale, JBS Yadawa, and Zhangyang Wang. Self-supervised learning for generalizable out-of-distribution detection. In *AAAI Conference on Artificial Intelligence*, 2020.

[28] Jishnu Mukhoti, Andreas Kirsch, Joost van Amersfoort, Philip HS Torr, and Yarin Gal. Deterministic neural networks with appropriate inductive biases capture epistemic and aleatoric uncertainty. *arXiv preprint arXiv:2102.11582*, 2021.

[29] Philipp Oberdiek, Matthias Rottmann, and Gernot A Fink. Detection and retrieval of out-of-distribution objects in semantic segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops*, pages 328–329, 2020.

[30] Nima Rafiee, Rahil Gholamipoor, and Markus Kollmann. Unsupervised anomaly detection from semantic similarity scores. *arXiv preprint arXiv:2012.00461*, 2020.

[31] Jie Ren, Peter J Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark Depristo, Joshua Dillon, and Balaji Lakshminarayanan. Likelihood ratios for out-of-distribution detection. In *Advances in Neural Information Processing Systems*, pages 14707–14718, 2019.

[32] Lukas Ruff, Robert Vandermeulen, Nico Goernitz, Lucas Deecke, Shoaib Ahmed Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *ICML*, pages 4393–4402, 2018.

[33] Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Alexander Binder, Emmanuel Müller, Klaus-Robert Müller, and Marius Kloft. Deep semi-supervised anomaly detection. In *International Conference on Learning Representations*, 2019.

[34] Chandramouli Shama Sastry and Sageev Oore. Detecting out-of-distribution examples with in-distribution examples and gram matrices. *arXiv preprint arXiv:1912.12510*, 2019.

[35] Vikash Sehwag, Mung Chiang, and Prateek Mittal. {SSD}: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*, 2021.

[36] Jasper Snoek, Yaniv Ovadia, Emily Fertig, Balaji Lakshminarayanan, Sebastian Nowozin, D. Sculley, Joshua V. Dillon, Jie Ren, and Zachary Nado. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. In *NeurIPS*, 2019.

[37] Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. Csi: Novelty detection via contrastive learning on distributionally shifted instances. In *34th Conference on Neural Information Processing Systems (NeurIPS) 2020*. Neural Information Processing Systems, 2020.

[38] Engkarat Techapanurak, Masanori Suganuma, and Takayuki Okatani. Hyperparameter-free out-of-distribution detection using cosine similarity. In *Proceedings of the Asian Conference on Computer Vision*, 2020.

[39] Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(11), 2008.

[40] Jim Winkens, Rudy Bunel, Abhijit Guha Roy, Robert Stanforth, Vivek Natarajan, Joseph R Ledsam, Patricia MacWilliams, Pushmeet Kohli, Alan Karthikesalingam, Simon Kohl, et al. Contrastive training for improved out-of-distribution detection. *arXiv preprint arXiv:2007.05566*, 2020.

[41] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *TPAMI*, 40(6):1452–1464, 2017.