Do We Really Need to Learn Representations from In-domain Data for Outlier Detection?

Zhisheng Xiao [1] **Qing Yan** [2] **Yali Amit** [2]

Abstract

Unsupervised outlier detection, which predicts if a test sample is an outlier or not using only the information from unlabelled inlier data, is an important but challenging task. Recently, methods based on the two-stage framework achieve state-of-the-art performance on this task. The framework leverages self-supervised representation learning algorithms to train a feature extractor on inlier data, and applies a simple outlier detector in the feature space. In this paper, we explore the possibility of avoiding the high cost of training a distinct representation for each outlier detection task, and instead using a single pre-trained network as the universal feature extractor regardless of the source of in-domain data. In particular, we replace the task-specific feature extractor by one network pre-trained on ImageNet with a self-supervised loss. In experiments, we demonstrate competitive or better performance on a variety of outlier detection benchmarks compared with previous two-stage methods, suggesting that learning representations from in-domain data may be unnecessary for outlier detection.

1. Introduction

Detecting outlier samples with only unlabeled data from the training distribution is challenging. Previous approaches based on reconstruction (Pidhorskyi et al., 2018; Zong et al., 2018) or density estimation (Du & Mordatch, 2019; Ren et al., 2019; Xiao et al., 2020b) do not obtain comparable performance with classifier based outlier detectors. More importantly, it is observed that density estimation with some probabilistic generative models may assign higher likelihoods to outliers than in-distribution data (Nalisnick et al., 2018), suggesting that there might be fundamental issues with this approach (Le Lan & Dinh, 2020). An alternative framework for unsupervised outlier detection can be summarized as a two-stage procedure, where in the first stage a neural network is used to extract a high-level representation of data and, in the second stage, an outlier detector is applied on the representation space. The key component of this framework is a good feature extractor that ensures the features of in-distribution data are clustered together, while keeping the features of outliers away from the cluster. Previous two-stage outlier detection methods require training the feature extractor on in-distribution data, either with a classification loss (if labels are available) (Ahuja et al., 2019; Lee et al., 2018b) or a self-supervised loss (Sehwag et al., 2021; Tack et al., 2020; Sohn et al., 2021).

In particular, the best feature extractors used in (Sehwag et al., 2021; Tack et al., 2020; Sohn et al., 2021) are trained with a contrastive loss (Chen et al., 2020a; He et al., 2020; Caron et al., 2020), which has led to tremendous progress in representation learning. However, training such representations can be difficult, as it may require large batch size (Chen et al., 2020a), long training epochs, and carefully designed data augmentation and training scheme (Chen et al., 2020a; He et al., 2020; Chen et al., 2020b; Tian et al., 2020; Xiao et al., 2020a). Note that in practice, outlier detection is often a *side task*, where outliers are filtered out before entering the networks designed for the main task. Therefore, the high cost of training feature extractors for outlier detection is undesirable.

Considering the fact that pre-trained feature extractors on some reference datasets (such as ImageNet in the image domain) are easily accessible,[1] and they are shown to be effective in transferring knowledge to other datasets (Marcelino, 2018; Grill et al., 2020; Lu et al., 2021; Azizi et al., 2021), we ask a natural but, to our knowledge, unexplored question: can we directly use feature extractors trained on a reference dataset without any further training to detect outliers for any source of in-distribution data? In this paper, we give an

[*]Equal contribution [1]Computational and Applied Mathematics, The University of Chicago, Chicago, USA [2]Department of Statistics, The University of Chicago, Chicago, USA. Correspondence to: Zhisheng Xiao <zxiao@uchicago.edu>, Qing Yan <yanq@uchicago.edu>.

Presented at the ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning.

[1]For example, pre-trained weights obtained from supervised training and a variety of self-supervised training algorithms on ImageNet are publicly available at https://github.com/facebookresearch/vissl/blob/master/MODEL_ZOO.md

affirmative answer to this question. Specifically, we focus on outlier detection in the image domain, and show that self-supervised feature extractors pre-trained on ImageNet lead to comparable or even better performances than state-of-the-art two-stage outlier detectors, which require expensive training to obtain representations of in-distribution data. In other words, we obtain an universal outlier detector without training on particular in-distribution data. This means that, given the public availability of pre-trained feature extractors, we can detect outlier samples *for free*.

2. Backgrounds and Our Approach

In this section, we provide a detailed review of the two-stage outlier detection framework, and position our approach in that framework.

2.1. Two-stage Outlier Detection

First Stage: In the first stage, in-distribution samples $\{\mathbf{x}_i\}_{i=1}^N$ are mapped to the representation space by a feature extractor network f, and the resulting representations $\mathbf{z}_i = f(\mathbf{x}_i)$ are collected. When there is no label available, previous methods train self-supervised feature extractors on in-distribution data (Sehwag et al., 2021; Sohn et al., 2021; Tack et al., 2020). In particular, it is observed in their experiments that feature extractors trained by contrastive loss lead to better results than those trained by handcrafted self-supervised tasks.

Second Stage: In the second stage, after collecting the set of features $\{\mathbf{z}_i\}_{i=1}^N$, a test sample $\mathbf{x}_{\text{test}}$ is mapped to the feature space using f to obtain $\mathbf{z}_{\text{test}}$, and a simple outlier detector is used to compute the detection score $s(\mathbf{x}_{\text{test}})$ in the feature space by treating $\{\mathbf{z}_i\}_{i=1}^N$ as in-distribution data. The motivation is that it is much easier to quantify distance between features, which are relatively low-dimensional vectors, than original data with high dimensionality and complicated structure. A variety of simple outlier detectors have been used in the second stage, including non-parametric detectors, such as One-class SVM (Schölkopf et al., 1999) and Kernel Density Estimation (KDE), and parametric detectors Mahalanobis distance (Sehwag et al., 2021).

2.2. Outlier Detection with Pre-trained Representation

Representation Learning Algorithm. We replace the feature extractors trained on different in-distribution data in previous two-stage methods by a single feature extractor pre-trained on ImageNet. Some previous work such as (Sohn et al., 2021) include extracting features by a classifier trained on ImageNet as a baseline, and they show that ImageNet classifiers lead to worse results than their proposed representation learning methods, highlighting the importance of learning representations from in-domain distributions. However, we believe the main reason is that representations obtained from classifiers mainly keep label-related information on ImageNet, which may not be useful for detecting outliers for other sources of data. In contrast, self-supervised representation learning develops a richer understanding of semantics, and absence of such semantics in outlier samples can cause them to lie far away in the feature space. Therefore, we propose to use pre-trained self-supervised representations on ImageNet for outlier detection.

Outlier detector in the feature space. We choose to build a parametric OOD detector in the feature space. Parametric models have huge advantage in computation during testing, as non-parametric models such as KDE and nearest neighbors require pair-wise computation with each element in the training set. Following (Sehwag et al., 2021; Lee et al., 2018b), we first partition the training features into several components by fitting a Gaussian mixture model, and the minimum Mahalanobis distance is used to detect outliers. We empirically find a tied covariance matrix as in (Lee et al., 2018b) leads to better performances.

3. Related Work

Unsupervised Outlier Detection: Most previous approaches can be categorized as based on either density (Ren et al., 2019; Nalisnick et al., 2019; Serrà et al., 2019; Choi et al., 2018; Xiao et al., 2020b; Havtorn et al., 2021), reconstruction (Zong et al., 2018; Pidhorskyi et al., 2018; Denouden et al., 2018; Perera & Patel, 2019) or feature distance (Sehwag et al., 2021; Tack et al., 2020; Sohn et al., 2021). Note that both density- or reconstruction-based methods are largely outperformed by classifier based outlier detectors (Liang et al., 2018; Lakshminarayanan et al., 2017; Lee et al., 2018a). We have reviewed the feature distance framework in Section 2.1. These methods obtain the best results in unsupervised outlier detection.

Other Outlier Detection Methods with Self-supervised Learning: Some other outlier detection methods have a self-supervised learning component, but they are not based on feature distance. For example, (Hendrycks et al., 2019; Winkens et al., 2020; Liu & Abbeel, 2020) use self-supervised loss in conjunction with supervised cross-entropy loss to improve OOD detection. (Bergman & Hoshen, 2020; Golan & El-Yaniv, 2018) train a network to predict certain geometric transformations, and detect outliers by the prediction accuracy.

Feature Distance with ImageNet Pre-trained Networks: An important motivation of our work is the success of using ImageNet pre-trained features to define a proper distance for images from other sources. For instance, features obtained from ImageNet classifiers are used to define scores such as IS and FID (Heusel et al., 2017; Salimans et al., 2016;

Sajjadi et al., 2018) that measure the distance between sets of arbitrary images, and they have been widely used for assessing the sample quality of generative models. Recently, (Morozov et al., 2021) propose to compute the FID score based on self-supervised representations, and show that the resulting score better aligns with human perceptual quality.[2]

4. Results

In this section, we evaluate our proposed method on a variety of outlier detection tasks. In particular, we study OOD detection, where the in-distribution samples come from a certain dataset and outliers come from other datasets, and one-class anomaly detection, where in a multi-class dataset, images from one class are given as inlier and those from remaining classes are given as outlier. Unlike many previous works which only consider outlier detection on low-resolution images (typically 32×32), we will evaluated our method on both small and large images. For reporting main results, we use the ResNet-50 network trained by **SimCLRv2** (Chen et al., 2020b) on ImageNet as the feature extractor. Throughout the paper, we use AUROC as the evaluation metric.

4.1. OOD Detection on Unlabeled Multi-class Datasets

We largely follow (Tack et al., 2020) to choose the datasets for benchmarking our method on the OOD detection task. In Table 1, we present results of OOD detection with CIFAR-10 as in-distribution data. We compare our methods against previous unsupervised OOD detectors, either based on deep generative models or self-supervised feature extractors trained on in-distribution data. We observe that our method significantly outperforms prior generative model based methods, and is competitive with SOTA self-supervised methods. In addition, from the last two lines, we observe that the feature extractor trained with self-supervised loss on ImageNet performs significantly better than the feature extractor trained with classification loss on ImageNet.

Next, we study OOD detection on higher resolution images, with different inlier-outlier pairs. In Table 2, we compare the results of our proposed method with the baseline where the feature extractor is pre-trained by the classification task on ImageNet. We find that our method is effective in all inlier-outlier pairs, and obtain nearly perfect results in many instances. Similar to the observation in CIFAR-10 experiments, the pre-trained self-supervised representation largely outperforms the pre-trained classifiers, especially on difficult tasks such as Dogs vs. Pets and Caltech-256 vs Places.

[2]Strictly speaking, the resulting score should not be called FID, as the network is no longer the InceptionV3 (Szegedy et al., 2016), but we still use the name for convenience.

4.2. Anomaly Detection on Unlabeled One-class Datasets

For anomaly detection tasks, we follow (Golan & El-Yaniv, 2018) to choose four image datasets in our experiments: CIFAR-10 (10 classes) and CIFAR-100 (20 super-classes) (Krizhevsky et al., 2009), Fashion-MNIST (10 classes) (Xiao et al., 2017) and Cats-vs-Dogs (2 classes) (Elson et al., 2007). We employ a one-vs-all evaluation scheme in each experiment, where the final metric is the mean AUCs over all in-lier classes. We report the above metric in Table 3, and compare our method with various methods based on self-supervised learning. Note that one-class anomaly detection is a more difficult task than OOD detection, as we have less training data and the outliers share similar texture with the inliers. Nevertheless, our proposed method is highly effective, achieving the best performances on two out of four tasks, while being slightly behind the best methods on the other two tasks. In particular, our method largely outperforms previous ones on Cat-vs-Dog, which is the only one-class anomaly detection task for high-resolution images.

The one-class anomaly detection tasks on CIFAR-10 and CIFAR-100 are more commonly studied in previous work, and we present more results on these two tasks, including the per-class AUROCs and confusion matrices in Appendix B.

4.3. Ablation Study

We perform an ablation study on the components of our method introduced in Section 2.2. Throughout the ablation study, we conduct experiments on the CIFAR-100 one-class anomaly detection task. We report the results in Appendix A.

5. Conclusion and Discussion

In this paper, we study the effectiveness of using representations pre-trained on ImageNet to detect outliers from various sources. Through extensive experiments, we show that by leveraging a single publicly available pre-trained feature extractor with self-supervised loss on ImageNet, we can achieve competitive performance on various outlier detection tasks. While previous work highlighted the importance of learning representations from in-domain data, our study suggests that the actual benefits of domain-specific training may be marginal. Our method has important practical implications, as it is easy to use and requires no training of the representation space. In addition, our results can serve as an important baseline for future studies on outlier detection.

Table 1. AUROC (%) of various unsupervised OOD detection methods with CIFAR-10 as in-distribution. LSUN(F) and ImageNet(F) correspond to the fixed version of LSUN and ImageNet introduced in (Tack et al., 2020), where they fixed the resize issue of resized LSUN and ImageNet datasets produced by broken image resize operations that contain artificial noise.

		SVHN	LSUN	ImageNet	LSUN (F)	ImageNet (F)	CIFAR-100
Generative Models	Glow	8.3	-	66.3	-	-	58.2
	EBM (Du & Mordatch, 2019)	63.0	-	-	-	-	50.0
	VAEBM (Xiao et al., 2021)	83.0	-	-	-	-	62.0
	Input Complexity (Serrà et al., 2019)	95.0	-	71.6	-	-	73.6
	Likelihood Ratio (Ren et al., 2019)	91.2	-	-	-	-	-
	Likelihood Regret (Xiao et al., 2020b)	87.5	69.1	-	-	-	-
Self-supervised Training	Rot + Trans (Hendrycks et al., 2019)	97.8	89.2	90.5	81.6	86.7	79.0
	GOAD (Bergman & Hoshen, 2020)	96.3	89.3	91.8	78.8	83.3	77.2
	CSI (Tack et al., 2020)	**99.8**	97.5	97.6	90.3	**93.3**	89.2
	SSD (Sehwag et al., 2021)	99.6	-	-	-	-	**90.6**
ImageNet Pre-trained	Supervised	86.3	76.9	80.3	69.9	76.8	70.4
	Self-supervised (ours)	98.3	**98.5**	**98.6**	**92.9**	91.8	81.7

Table 2. AUROC (%) on various high resolution image datasets. For each grid in the table, **Top:** using features extracted by a ResNet-50 pre-trained with SimCLRv2, and **Bottom:** using features extracted by a ResNet-50 pre-trained with classification task.

Inlier \ Outlier	ImageNet-30	CUB	Dogs	Flowers	Pets	Places	Caltech
ImageNet-30	-	99.8 74.5	99.5 95.4	98.1 85.1	99.7 92.5	80.0 75.4	89.2 80.9
Dogs	99.9 95.5	99.9 96.5	-	99.9 98.7	94.5 72.4	99.7 94.4	99.2 95.2
Places	99.4 89.4	99.8 87.8	99.0 95.7	97.4 86.4	99.8 93.6	-	93.4 84.1
Caltech	95.1 83.5	98.9 63.8	96.0 89.6	85.6 60.7	99.2 91.5	74.3 61.9	-

Table 3. Mean AUROC (%) for one-class classification AUCs averaged over outlier classes and over 5 runs. We omit the standard deviations as they are small.

		CIFAR-10	CIFAR-100	f-MNIST	Cat-vs-Dog
Self-supervised Training	Rot Prediction(Sohn et al., 2021)	91.3	84.1	**95.8**	86.4
	Contrastive(Sohn et al., 2021)	89.0	82.4	93.6	87.7
	Contrastive+DA(Sohn et al., 2021)	92.5	86.5	94.8	89.6
	Geometric Trans(Golan & El-Yaniv, 2018)	86.0	78.7	93.5	88.8
	InvAE(Fei et al., 2020)	86.6	78.8	93.9	-
	Rot + Trans (Hendrycks et al., 2019)	90.1	-	-	-
	GOAD (Bergman & Hoshen, 2020)	88.2	-	94.1	-
	CSI (Tack et al., 2020)	**94.3**	89.6	-	-
	SSD (Sehwag et al., 2021)	90.0	-	-	-
ImageNet Pre-trained	Supervised	86.2	87.1	90.2	89.4
	Self-supervised (ours)	93.8	**92.6**	94.4	**94.8**

References

Ahuja, N. A., Ndiour, I., Kalyanpur, T., and Tickoo, O. Probabilistic modeling of deep features for out-of-distribution and adversarial detection. *arXiv preprint arXiv:1909.11786*, 2019.

Azizi, S., Mustafa, B., Ryan, F., Beaver, Z., Freyberg, J., Deaton, J., Loh, A., Karthikesalingam, A., Kornblith, S., Chen, T., et al. Big self-supervised models advance medical image classification. *arXiv preprint arXiv:2101.05224*, 2021.

Bergman, L. and Hoshen, Y. Classification-based anomaly detection for general data. *arXiv preprint arXiv:2005.02359*, 2020.

Caron, M., Misra, I., Mairal, J., Goyal, P., Bojanowski, P., and Joulin, A. Unsupervised learning of visual features by contrasting cluster assignments. *arXiv preprint arXiv:2006.09882*, 2020.

Chalapathy, R., Menon, A. K., and Chawla, S. Anomaly detection using one-class neural networks. *arXiv preprint arXiv:1802.06360*, 2018.

Chen, T., Kornblith, S., Norouzi, M., and Hinton, G. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020a.

Chen, T., Kornblith, S., Swersky, K., Norouzi, M., and Hinton, G. Big self-supervised models are strong semi-supervised learners. *arXiv preprint arXiv:2006.10029*, 2020b.

Chen, X., Fan, H., Girshick, R., and He, K. Improved baselines with momentum contrastive learning. *arXiv preprint arXiv:2003.04297*, 2020c.

Choi, H., Jang, E., and Alemi, A. A. Waic, but why? generative ensembles for robust anomaly detection. *arXiv preprint arXiv:1810.01392*, 2018.

Denouden, T., Salay, R., Czarnecki, K., Abdelzad, V., Phan, B., and Vernekar, S. Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance. *arXiv preprint arXiv:1812.02765*, 2018.

Du, Y. and Mordatch, I. Implicit generation and generalization in energy-based models. *ArXiv*, abs/1903.08689, 2019.

Elson, J., Douceur, J. R., Howell, J., and Saul, J. Asirra: a captcha that exploits interest-aligned manual image categorization. In *ACM Conference on Computer and Communications Security*, volume 7, pp. 366–374, 2007.

Fei, Y., Huang, C., Jinkun, C., Li, M., Zhang, Y., and Lu, C. Attribute restoration framework for anomaly detection. *IEEE Transactions on Multimedia*, 2020.

Golan, I. and El-Yaniv, R. Deep anomaly detection using geometric transformations. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, pp. 9781–9791, 2018.

Goyal, S., Raghunathan, A., Jain, M., Simhadri, H. V., and Jain, P. Drocc: Deep robust one-class classification. In *International Conference on Machine Learning*, pp. 3711–3721. PMLR, 2020.

Grill, J.-B., Strub, F., Altché, F., Tallec, C., Richemond, P., Buchatskaya, E., Doersch, C., Pires, B. A., Guo, Z., Azar, M. G., et al. koray kavukcuoglu, remi munos, and michal valko. bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 2020.

Havtorn, J. D., Frellsen, J., Hauberg, S., and Maaløe, L. Hierarchical vaes know what they don't know. *arXiv preprint arXiv:2102.08248*, 2021.

He, K., Fan, H., Wu, Y., Xie, S., and Girshick, R. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9729–9738, 2020.

Hendrycks, D., Mazeika, M., Kadavath, S., and Song, D. Using self-supervised learning can improve model robustness and uncertainty. *arXiv preprint arXiv:1906.12340*, 2019.

Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., and Hochreiter, S. Gans trained by a two time-scale update rule converge to a local nash equilibrium. *arXiv preprint arXiv:1706.08500*, 2017.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Lakshminarayanan, B., Pritzel, A., and Blundell, C. Simple and scalable predictive uncertainty estimation using deep ensembles. In *NIPS*, 2017.

Le Lan, C. and Dinh, L. Perfect density models cannot guarantee anomaly detection. In *"I Can't Believe It's Not Better!"NeurIPS 2020 workshop*, 2020.

Lee, K., Lee, H., Lee, K., and Shin, J. Training confidence-calibrated classifiers for detecting out-of-distribution samples. *ArXiv*, abs/1711.09325, 2018a.

Lee, K., Lee, K., Lee, H., and Shin, J. A simple unified framework for detecting out-of-distribution samples and adversarial attacks. In *NeurIPS*, 2018b.

Liang, S., Li, Y., and Srikant, R. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv: Learning*, 2018.

Liu, H. and Abbeel, P. Hybrid discriminative-generative training via contrastive learning. *arXiv preprint arXiv:2007.09070*, 2020.

Lu, Y., Jha, A., and Huo, Y. Contrastive learning meets transfer learning: A case study in medical image analysis. *arXiv preprint arXiv:2103.03166*, 2021.

Marcelino, P. Transfer learning from pre-trained models. *Towards Data Science*, 2018.

Morozov, S., Voynov, A., and Babenko, A. On self-supervised image representations for {gan} evaluation. In *International Conference on Learning Representations*, 2021.

Nalisnick, E., Matsukawa, A., Teh, Y. W., Gorur, D., and Lakshminarayanan, B. Do deep generative models know what they don't know? *arXiv preprint arXiv:1810.09136*, 2018.

Nalisnick, E., Matsukawa, A., Teh, Y. W., and Lakshminarayanan, B. Detecting out-of-distribution inputs to deep generative models using a test for typicality. *arXiv preprint arXiv:1906.02994*, 5:5, 2019.

Perera, P. and Patel, V. M. Deep transfer learning for multiple class novelty detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11544–11552, 2019.

Perera, P., Nallapati, R., and Xiang, B. Ocgan: One-class novelty detection using gans with constrained latent representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2898–2906, 2019.

Pidhorskyi, S., Almohsen, R., Adjeroh, D. A., and Doretto, G. Generative probabilistic novelty detection with adversarial autoencoders. *arXiv preprint arXiv:1807.02588*, 2018.

Ren, J., Liu, P. J., Fertig, E., Snoek, J., Poplin, R., DePristo, M. A., Dillon, J. V., and Lakshminarayanan, B. Likelihood ratios for out-of-distribution detection. In *NeurIPS*, 2019.

Ruff, L., Vandermeulen, R. A., Görnitz, N., Binder, A., Müller, E., Müller, K.-R., and Kloft, M. Deep semi-supervised anomaly detection, 2019.

Sajjadi, M. S., Bachem, O., Lucic, M., Bousquet, O., and Gelly, S. Assessing generative models via precision and recall. *arXiv preprint arXiv:1806.00035*, 2018.

Salimans, T., Goodfellow, I., Zaremba, W., Cheung, V., Radford, A., and Chen, X. Improved techniques for training gans. *arXiv preprint arXiv:1606.03498*, 2016.

Schlegl, T., Seeböck, P., Waldstein, S. M., Schmidt-Erfurth, U., and Langs, G. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International conference on information processing in medical imaging*, pp. 146–157. Springer, 2017.

Schölkopf, B., Williamson, R. C., Smola, A. J., Shawe-Taylor, J., Platt, J. C., et al. Support vector method for novelty detection. In *NIPS*, volume 12, pp. 582–588. Citeseer, 1999.

Sehwag, V., Chiang, M., and Mittal, P. Ssd: A unified framework for self-supervised outlier detection. In *International Conference on Learning Representations*, 2021.

Serrà, J., Álvarez, D., Gómez, V., Slizovskaia, O., Núñez, J. F., and Luque, J. Input complexity and out-of-distribution detection with likelihood-based generative models. *arXiv preprint arXiv:1909.11480*, 2019.

Sohn, K., Li, C.-L., Yoon, J., Jin, M., and Pfister, T. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., and Wojna, Z. Rethinking the inception architecture for computer vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2818–2826, 2016.

Tack, J., Mo, S., Jeong, J., and Shin, J. Csi: Novelty detection via contrastive learning on distributionally shifted instances. *arXiv preprint arXiv:2007.08176*, 2020.

Tian, Y., Sun, C., Poole, B., Krishnan, D., Schmid, C., and Isola, P. What makes for good views for contrastive learning. *arXiv preprint arXiv:2005.10243*, 2020.

Wang, S., Zeng, Y., Liu, X., Zhu, E., Yin, J., Xu, C., and Kloft, M. Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. In *NeurIPS*, pp. 5960–5973, 2019.

Wang, Z., Chen, J., and Hoi, S. C. Deep learning for image super-resolution: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 2020.

Winkens, J., Bunel, R., Roy, A. G., Stanforth, R., Natarajan, V., Ledsam, J. R., MacWilliams, P., Kohli, P., Karthikesalingam, A., Kohl, S., et al. Contrastive training for improved out-of-distribution detection. *arXiv preprint arXiv:2007.05566*, 2020.

Xiao, H., Rasul, K., and Vollgraf, R. Fashion-mnist: a novel image dataset for benchmarking machine learning algorithms. *arXiv preprint arXiv:1708.07747*, 2017.

Xiao, T., Wang, X., Efros, A. A., and Darrell, T. What should not be contrastive in contrastive learning. *arXiv preprint arXiv:2008.05659*, 2020a.

Xiao, Z., Yan, Q., and Amit, Y. Likelihood regret: An out-of-distribution detection score for variational auto-encoder. *Advances in Neural Information Processing Systems*, 33, 2020b.

Xiao, Z., Kreis, K., Kautz, J., and Vahdat, A. Vaebm: A symbiosis between variational autoencoders and energy-based models. In *International Conference on Learning Representations*, 2021.

Zbontar, J., Jing, L., Misra, I., LeCun, Y., and Deny, S. Barlow twins: Self-supervised learning via redundancy reduction. *arXiv preprint arXiv:2103.03230*, 2021.

Zong, B., Song, Q., Min, M. R., Cheng, W., Lumezanu, C., Cho, D., and Chen, H. Deep autoencoding gaussian mixture model for unsupervised anomaly detection. In *International Conference on Learning Representations*, 2018.

A. Ablation Study

In this section, We perform an ablation study on the components of our method introduced in Section 2.2. Throughout this section, we conduct experiments on the CIFAR-100 one-class anomaly detection task.

Representation Learning Algorithm. The core component of our method is the feature extractor trained on ImageNet with self-supervised loss. In Table 4, we compare several state-of-the-art self-supervised representation learning algorithms while keeping other factors fixed. We observe that all algorithms can be effectively applied to the anomaly detection task. In particular, the algorithms based on the contrastive loss (SimCLR and MoCo) obtain better performances than algorithms with alternative losses, although the latter may obtain better classification accuracy on ImageNet (under the linear evaluation protocol). Note that previous two-stage outlier detection algorithms, such as (Tack et al., 2020; Sehwag et al., 2021; Sohn et al., 2021) also train representations with contrastive losses. Therefore, probably the contrastive loss introduces a good inductive bias for outlier detection, as it explicitly pushes dissimilar images away. Given its popularity in previous work on outlier detection and the its strong empirical performance in our study, SimCLRv2 is chosen as the pre-train algorithm for all experiments.

Network Structure. Recent advances in self-supervised representation learning mainly use ResNet-50 as the backbone structure, while wider and deeper variants also exist. We fix the SimCLRv2 learning algorithm, and explore the effect of feature extractor's capacity on outlier detection in Table 5. We observe that deeper networks lead to slightly improved performance. We also note that there is a performance drop on wider ResNet-50, and the reason is that the wider network doubles the dimension of the feature space (4096 vs. 2048), making the feature space outlier detection much harder (note that for each class, we only have 2500 inlier samples). Considering the computational cost in test time, we choose to use the basic ResNet-50 structure throughout the paper.

Feature Space Outlier Detector. We compare different feature space outlier detectors in Table 6, including non-parametric (OC-SVM with RBF kernel and KDE with Gaussian kernel) and parametric (Mahalanobis distance) detectors. For non-parametric detectors, we do a grid search on the hyper-parameters (such as the kernel coefficient for OC-SVM and the bandwidth for KDE) and report the best results. For Mahalanobis distance, we try either a single component or 4 components obtained from K-means clustering. We do not consider more components because of the small in-lier data size. We also compare using a tied covariance matrix vs. component conditional covariance matrices. From Table 6, we observe that parametric detec-

Table 4. Ablation study on representation learning algorithms for the pre-trained feature extractor with ResNet-50 structure. Top1 Acc is the top 1 accuracy for linear evaluation on ImageNet.

Algorithm	AUROC	Top1 Acc
SimCLRv2 (Chen et al., 2020b)	**92.6**	74.6
MoCov2 (Chen et al., 2020c)	92.1	71.1
SwAV (Caron et al., 2020)	91.2	**75.3**
BYOL (Grill et al., 2020)	90.7	74.3
Barlow Twins (Zbontar et al., 2021)	89.5	73.2

Table 5. Ablation study on the network structure for the feature extractor trained with SimCLRv2. Results of linear evaluation accuracy are reported in (Chen et al., 2020b).

Network	AUROC	Top1 Acc
ResNet-50	92.6	74.6
ResNet-50 2×	87.2	**77.7**
ResNet-101	93.3	76.3
ResNet-152	**94.0**	77.2

Table 6. Ablation study on the feature space outlier detector. Non-parametric and parametric detectors are separated by the bar. **MD** stands for Mahalanobis distance, and **tied** means using a tied covariance matrix for all components.

Decision score	Component	AUROC
OC-SVM	-	90.3
KDE	-	88.7
MD	1	91.8
MD	4	86.7
MD (tied)	4	**92.6**

Table 7. Ablation study on the input pre-processing.

Up-sampling	Perturbation	AUROC
Nearest Neighbor	✗	80.7
Bilinear	✗	90.4
Cubic	✗	92.6
Lanczos	✗	92.8

tors outperforms non-parametric detectors. This suggests that the representations extracted by ImageNet pre-trained network have a compact structure, which can be well approximated by simple parametric models. Interestingly, we find that for Mahalanobis distance, a single tied covariance matrix outperforms a separate covariance matrix for each cluster. Considering its good performances and computational efficiency, we use Mahalanobis distance with a tied covariance matrix as the feature outlier detector in the main experiments.

Input Pre-processing. We use a single feature extractor for all outlier detection tasks, where images can vary significantly in sizes. We need to pre-process the data to fit the input size of feature extractors trained on ImageNet. In particular, for small image datasets such as CIFAR-10 and CIFAR-100 with resolution 32×32, we need to up-sample the images by a factor of 8 (followed by a center crop to size 224×224). We study the effect of different up-sampling methods in Table 7, where we observe that more sophisticated up-samplers such as bi-cubic interpolation and Lanczos resampling lead to significantly better results than simple nearest neighbor up-sampling. We use bi-cubic interpolation throughout the main results for a balance of simplicity and effectiveness. We believe that using learning based up-sampler (Wang et al., 2020) will lead to improved performances, however, since our goal is to avoid any training on inlier data, we do not explore that direction.

B. Additional One-class Anomaly Detection Results on CIFAR-10 and CIFAR-100

The one-class anomaly detection tasks on CIFAR-10 and CIFAR-100 are studied extensively in previous work. Due to the space limit, in this section we compare our method with previous method in details. In Table 8 we compare our method with other methods (based on generative models, one-class classifiers and representation learning) on one-class anomaly detection task on CIFAR-10 for each of the 10 classes. Our method is competitive with CSI, the current SOTA. Each of our method and CIS obtains the best result on 5 out of 10 classes, while on average our method is only slightly worse (93.8 vs. 94.3).

In Table 9, we present the confusion matrix of AUROC of our method on one-class anomaly detection task on CIFAR-10, where bold denotes the hard pairs (AUROC less than 80%). The results align with the human intuition that 'car' is confused to 'truck', 'cat' is confused to 'dog', and 'deer' is confused to 'horse'.

Table 10 compares our method with various methods on one-class anomaly detection task on CIFAR-100 (super-class), for each of the 20 super-classes. Our method outperforms the prior methods for most of the classes and obtains the best average result.

Table 8. Comparison of our method with other detectors for one class anomaly detection task on CIFAR-10. Results of other methods are presented in (Sehwag et al., 2021).

	Airplane	Automobile	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Average
Randomly Initialized network	77.4	44.1	62.4	44.1	62.1	49.6	59.8	48.0	73.8	53.7	57.5
VAE	70.0	38.6	67.9	53.5	74.8	52.3	68.7	49.3	69.6	38.6	58.3
PixelCNN	53.1	99.5	47.6	51.7	73.9	54.2	59.2	78.9	34.0	66.2	61.8
OCSVM (Schölkopf et al., 1999)	63.0	44.0	64.9	48.7	73.5	50.0	72.5	53.3	64.9	50.8	58.5
AnoGAN (Schlegl et al., 2017)	67.1	54.7	52.9	54.5	65.1	60.3	58.5	62.5	75.8	66.5	61.8
DSVDD (Ruff et al., 2019)	61.7	65.9	50.8	59.1	60.9	65.7	67.7	67.3	75.9	73.1	64.8
OCGAN (Perera et al., 2019)	75.7	53.1	64.0	62.0	72.3	62.0	72.3	57.5	82.0	55.4	65.6
RCAE (Chalapathy et al., 2018)	72.0	63.1	71.7	60.6	72.8	64.0	64.9	63.6	74.7	74.5	68.2
DROCC (Goyal et al., 2020)	81.7	76.7	66.7	67.1	73.6	74.4	74.4	71.4	80.0	76.2	74.2
Deep-SAD (Ruff et al., 2019)	–	–	–	–	–	–	–	–	–	–	77.9
E3Outlier (Wang et al., 2019)	79.4	95.3	75.4	73.9	84.1	87.9	85.0	93.4	92.3	89.7	85.6
Geom Trans (Golan & El-Yaniv, 2018)	74.7	95.7	78.1	72.4	87.8	87.8	83.4	95.5	93.3	91.3	86.0
InvAE (Fei et al., 2020)	78.5	89.8	86.1	77.4	90.5	84.5	89.2	92.9	92.0	85.5	86.6
GOAD (Bergman & Hoshen, 2020)	77.2	96.7	83.3	77.7	87.8	87.8	90.0	96.1	93.8	92.0	88.2
CSI (Tack et al., 2020)	89.9	**99.9**	**93.1**	86.4	**93.9**	93.2	95.1	**98.7**	**97.9**	95.5	**94.3**
SSD (Sehwag et al., 2021)	82.7	98.5	84.2	84.5	84.8	90.9	91.7	95.2	92.9	94.4	90.0
Supervised pre-train	84.5	96.1	77.3	78.9	84.8	82.3	90.7	88.6	85.3	94.5	86.2
Ours	**94.8**	96.4	88.3	**87.6**	92.7	**94.2**	**96.4**	94.3	96.1	**97.0**	93.8

Table 9. Confusion matrix of AUROC (%) values of our method on one-class CIFAR-10. The row and column indicates the in-distribution and OOD class, respectively, and the final column indicates the mean value. Bold denotes the values under 80%, which implies the hard pair.

	Plane	Car	Bird	Cat	Deer	Dog	Frog	Horse	Ship	Truck	Mean
Plane	-	94.5	95.7	98.9	97.4	99.3	99.1	96.8	81.1	91.4	95.0
Car	97.2	-	99.8	99.7	99.6	99.8	99.8	99.6	96.0	**76.5**	96.5
Bird	87.8	99.5	-	90.5	**63.5**	90.5	87.0	**79.7**	98.0	99.2	88.4
Cat	95.4	98.5	91.1	-	81.7	**58.4**	82.9	83.1	98.1	97.9	87.4
Deer	96.6	99.7	90.2	93.8	-	93.5	92.3	**69.2**	98.9	99.3	92.6
Dog	99.3	99.7	97.5	81.9	90.4	-	96.1	85.3	99.7	99.6	94.3
Frog	98.7	99.7	94.5	93.3	89.0	95.8	-	97.6	99.7	99.8	96.4
Horse	96.9	99.2	96.4	95.1	**75.7**	91.9	98.3	-	99.2	98.2	94.5
Ship	86.2	92.5	99.2	99.4	98.9	99.6	99.7	99.0	-	90.0	96.0
Truck	96.4	**79.5**	99.9	99.8	99.9	99.9	99.9	99.3	96.5	-	96.9

Table 10. AUROC (%) values of various OOD detection methods trained on one-class CIFAR-100 (super-class). Each row indicates the results of the selected super-class, and the final row indicates the mean value. The values of other methods are from the reference.

	OC-SVM	DAGMM	DSEBM	ADGAN	Geom Trans	Rot+Trans	GOAD	CSI	Ours
0	68.4	43.4	64.0	63.1	74.7	79.6	73.9	86.3	**91.6**
1	63.6	49.5	47.9	64.9	68.5	73.3	69.2	84.8	**88.3**
2	52.0	66.1	53.7	41.3	74.0	71.3	67.6	88.9	**95.0**
3	64.7	52.6	48.4	50.0	81.0	73.9	71.8	85.7	**96.7**
4	58.2	56.9	59.7	40.6	78.4	79.7	72.7	93.7	**94.8**
5	54.9	52.4	46.6	42.8	59.1	72.6	67.0	81.9	**95.1**
6	57.2	55.0	51.7	51.1	81.8	85.1	80.0	91.8	**96.5**
7	62.9	52.8	54.8	55.4	65.0	66.8	59.1	83.9	**90.0**
8	65.6	53.2	66.7	59.2	85.5	86.0	79.5	91.6	**93.0**
9	74.1	42.5	71.2	62.7	90.6	87.3	83.7	**95.0**	92.5
10	84.1	52.7	78.3	79.8	87.6	88.6	84.0	94.0	**95.2**
11	58.0	46.4	62.7	53.7	83.9	77.1	68.7	90.1	**91.6**
12	68.5	42.7	66.8	58.9	83.2	84.6	75.1	**90.3**	**90.3**
13	64.6	45.4	52.6	57.4	58.0	62.1	56.6	81.5	**85.5**
14	51.2	57.2	44.0	39.4	92.1	88.0	83.8	94.4	**96.7**
15	62.8	48.8	56.8	55.6	68.3	71.9	66.9	85.6	**86.5**
16	66.6	54.4	63.1	63.3	73.5	75.6	67.5	83.0	**88.6**
17	73.7	36.4	73.0	66.7	93.8	93.5	91.6	**97.5**	95.6
18	52.8	52.4	57.7	44.3	90.7	91.5	88.0	**95.9**	95.3
19	58.4	50.3	55.5	53.0	85.0	88.1	82.6	95.2	**93.1**
Mean	63.1	50.6	58.8	55.2	78.7	79.8	74.5	89.6	**92.6**