Exposing Outlier Exposure: What Can Be Learned From Few, One, and Zero Outlier Images

Anonymous authors
Paper under double-blind review

Abstract

Due to the intractability of characterizing *everything* that looks unlike the normal data, anomaly detection (AD) is traditionally treated as an unsupervised problem utilizing only normal samples. However, it has recently been found that unsupervised image AD can be drastically improved through the utilization of huge corpora of random images to represent anomalousness; a technique which is known as *Outlier Exposure*. In this paper we show that specialized AD learning methods are unnecessary for state-of-the-art performance, and furthermore one can achieve strong performance with just a small collection of Outlier Exposure data, contradicting common assumptions in the field of AD. We find that standard classifiers and semi-supervised one-class methods trained to discern between normal samples and relatively few random natural images are able to outperform the current state of the art on an established AD benchmark with ImageNet. Further experiments reveal that even *one* well-chosen outlier sample is sufficient to achieve decent performance on this benchmark (79.3% AUC). We investigate this phenomenon and find that one-class methods are more robust to the choice of training outliers, indicating that there are scenarios where these are still more useful than standard classifiers. Lastly, no training samples are necessary when one uses the representations learned by CLIP, a recent foundation model, which achieves state-of-the-art AD results on CIFAR-10 and ImageNet in a zero-shot setting.

1 Introduction

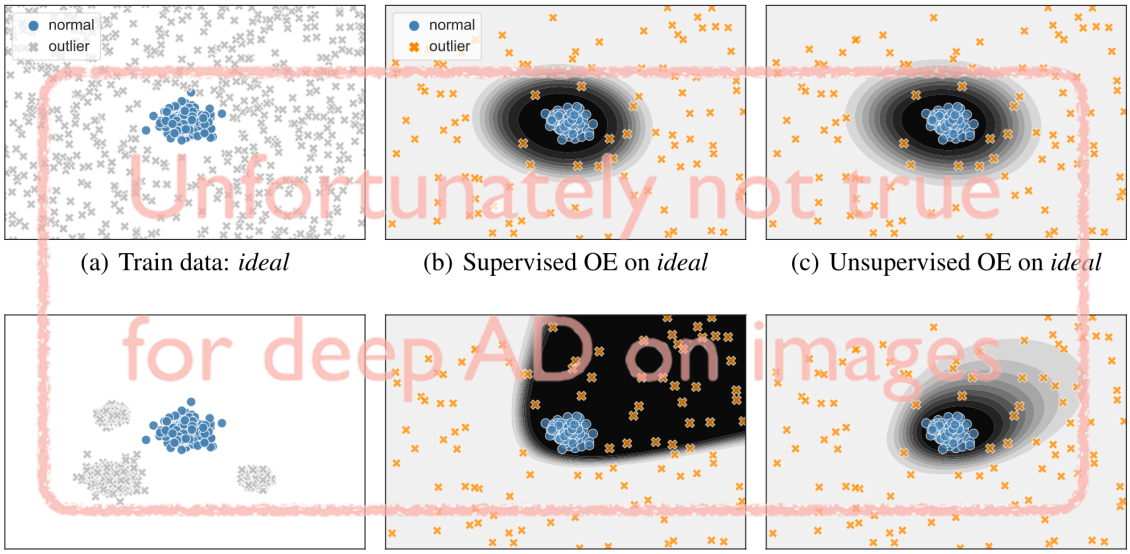
Anomaly detection (AD) (Chandola et al., 2009) is the task of determining whether a sample is anomalous compared to a corpus of data. Recently there has been a great interest in developing novel deep methods for AD (Ruff et al., 2021; Pang et al., 2021). Most prior work performs AD in an *unsupervised* way utilizing only an unlabeled corpus of mostly normal data (Golan & El-Yaniv, 2018; Hendrycks et al., 2019b; Bergman et al., 2020; Tack et al., 2020). While AD can be interpreted as a classification problem of "normal vs. anomalous," it is classically treated as an unsupervised problem due to the rather tricky issue of finding or constructing a dataset that captures *everything different* from the normal dataset.

One often has, in addition to normal data, access to some data which is known to be anomalous. Hendrycks et al. (2019a) noted that, for an image AD problem, one has access to a virtually limitless amount of random natural images from the internet that are presumably not normal. They term the utilization of such auxiliary data *Outlier Exposure* (OE). Many top-performing AD methods on standard image AD benchmarks utilize tens of thousands of OE samples combined with self-supervised learning (Hendrycks et al., 2019b) or transfer learning (Reiss et al., 2021; Deecke et al., 2021) to achieve state-of-the-art detection performance.

For clarity, we here delineate three basic approaches to anomaly detection:

- *Unsupervised*: These are methods trained on unlabeled data that is assumed to be mostly normal. This is the classic and most common approach to AD.
- *Unsupervised OE*: These are adaptations of unsupervised methods that incorporate auxiliary OE data that is not normal. Elsewhere this is also called "semi-supervised" AD (Görnitz et al., 2013; Ruff et al., 2020).
- *Supervised OE*: This simply indicates standard classification methods trained to discern between normal data and an auxiliary OE dataset that is not normal.

Using unsupervised OE rather than supervised OE to discern between the normal data and OE samples seems intuitive since the presented anomalies likely do not completely characterize "anomalousness." Figure 1 illustrates this classic intuition and highlights the differences between these approaches on a 2D toy dataset. The benefits of unsupervised OE when incorporating (a few specific) known anomalies has also been observed (Tax, 2001; Görnitz et al., 2013; Ruff et al., 2020).



(a) Train data: *ideal* (b) Supervised OE on *ideal* (c) Unsupervised OE on *ideal*

(d) Train data: *skewed* (e) Supervised OE on *skewed* (f) Unsupervised OE on *skewed*

Unfortunately not true for deep AD on images

Figure 1: This figure visualizes a deceptively reasonable intuition that one might have for deep AD. It shows the decision boundaries of a supervised OE method and an unsupervised OE method on two toy datasets: *ideal* (a–c) and *skewed* (d–f). The skewed scenario occurs naturally when there are not enough OE samples to cover the ambient space, the data dimensionality is high, or the OE data is clustered. Unsupervised OE (c + f) learns a compact decision region for the normal class that generalizes well in both scenarios. A supervised OE approach (b + e), on the other hand, learns a decision region that generalizes well in the *ideal* case where the outlier data is fully representative, but not in the *skewed* case. While this intuition is true for shallow AD settings, like in this 2D toy example (a–f), our results suggest that a deep approach removes the aforementioned phenomena so supervised OE performs remarkably well.

In this paper we present surprising experimental results that challenge the assumption that deep AD on images needs an unsupervised approach (with or without OE). Using the same OE setup as Hendrycks et al. (2019b), which is common in the literature, we find that a standard classifier outperforms current state-of-the-art AD methods on the one vs. rest AD benchmark with CIFAR-10 and ImageNet. The one vs. rest benchmark has been recommended as a standard evaluation protocol to validate AD methods (Emmott et al., 2013) and is used as a litmus test in virtually all deep AD papers published at top-tier venues; see e.g. (Ruff et al., 2018; Deecke et al., 2018; Golan & El-Yaniv, 2018; Akcay et al., 2018; Hendrycks et al., 2019b; Abati et al., 2019; Perera et al., 2019; Wang et al., 2019a; Ruff et al., 2020; Bergman & Hoshen, 2020; Kim et al., 2020; Liznerski et al., 2021; Deecke et al., 2021). Further challenging common assumptions, we find that OE does *not* seem to require huge amounts of data to represent "anomalousness." A classifier requires only 256 random OE samples to outperform the state of the art on ImageNet and only *one* well-chosen OE sample to score reasonably compared to unsupervised methods and classical AD approaches. We also investigate transfer learning approaches to AD that have recently improved AD on images (Reiss et al., 2021; Deecke et al., 2021). Using CLIP (Radford et al., 2021), a recent foundation model, we find that it is possible to set a state of the art on CIFAR-10 and ImageNet *without any additional training data*. While transfer learning and standard classification work well, we still show advantages of unsupervised OE over supervised OE, particularly regarding robustness.

The complete code to reproduce our results is included in the supplementary material.

In conclusion, the primary message of this paper is neither that we propose yet another state-of-the-art method nor that one of our investigated methodologies is of general superiority, but that there is a surprisingly strong performance of off-the-shelf classifiers, transfer learning, and just a few OE samples–contradicting widespread common assumptions for deep AD on well-established AD benchmarks. Through this work, we want to encourage rethinking how previous AD results extend to deep learning.

2 Related work

Here we briefly review recent developments in deep AD, including self-supervision, transfer learning, and outlier exposure. We further clarify the differences between out-of-distribution detection and AD and discuss non-natural image AD benchmarks.

Deep anomaly detection While there exist many shallow methods for AD, it has been observed that these methods perform poorly on high-dimensional data (Huang & LeCun, 2006; Kriegel et al., 2008; Erfani et al., 2015; 2016). To address this, deep approaches to AD that scale well with higher dimensions have been proposed (Ruff et al., 2021; Pang et al., 2021). The most common approaches to deep AD employ autoencoders trained on normal data, where samples not reconstructed well at test time are deemed anomalous (Hawkins et al., 2002; Sakurada & Yairi, 2014; Chen et al., 2017; Zhou & Paffenroth, 2017; Nguyen et al., 2019; Kim et al., 2020). Deep generative models detect anomalies via a variety of methods (Schlegl et al., 2017; Deecke et al., 2018; Zenati et al., 2018; Schlegl et al., 2019), yet their effectiveness has been called into question (Nalisnick et al., 2019).

A recent avenue of research uses self-supervision for deep AD on images (Gidaris et al., 2018; Golan & El-Yaniv, 2018; Wang et al., 2019b; Hendrycks et al., 2019b; Tack et al., 2020; Sohn et al., 2021). One of the best-performing AD methods is the self-supervised approach from Tack et al. (2020), which combines Hendrycks et al. (2019b)'s AD method with contrastive representation learning (Chen et al., 2020). Tack et al. (2020) train their network on transformed normal data so that it maps similar transformations of a sample close together, while sufficiently distorting transformations and other samples are mapped away from it. The network then has to classify each sample's type of transformation as in (Hendrycks et al., 2019b). For a test sample, both the trained network's certainty on predicting correct transformations and the similarity of the sample with its nearest neighbor in feature space determine its anomaly score: the larger the certainty and similarity, the smaller the anomaly score.

More recently, transfer learning-based approaches to AD (Bergman et al., 2020; Reiss et al., 2021; Deecke et al., 2021) that fine-tune supervised classifiers trained on ImageNet have shown to outperform self-supervised methods such as Tack et al. (2020) on common benchmarks. To the best of our knowledge, Reiss et al. (2021) is the best performing AD method on CIFAR-10 that does not use OE (Hendrycks et al., 2019a). Their method fine-tunes a ResNet pre-trained on ImageNet on a deep one-class loss (Ruff et al., 2018) and applies continual learning to avoid a feature collapse. Since they use ImageNet pre-trained models, they do not validate their method on the ImageNet one vs. rest benchmark.

The state of the art on the CIFAR-10 benchmark, however, takes advantage of OE (Hendrycks et al., 2019a), which follows the idea of using a large unstructured corpus of images as "auxiliary anomalies" during training. For example, Hendrycks et al. (2019b) use OE to improve their self-supervised method by training the network to predict the uniform distribution for all transforms on OE samples, while leaving training on normal samples unchanged. Reiss et al. (2021) and Deecke et al. (2021) combined transfer learning and OE, which yields the currently best performing methods.

Out-of-distribution detection and anomaly detection A field of research related to AD is out-of-distribution (OOD) detection, where the aim is to detect anomalous samples that do not belong to any of the given classes of a multi-class classification task (Lee et al., 2018). One can always apply an AD method to OOD by using it separately from the classifier, treating all training samples of all classes as normal, ignoring the available class labels. However, AD methods typically (and expectedly) perform worse than specialized OOD methods that take advantage of the in-distribution labels and confidence scores of a trained classifier (Liang et al., 2018; Tack et al., 2020; Hendrycks et al., 2022). Such methods define the anomaly score to be

large when the maximum of the softmax outputs (Liang et al., 2018) or logits (Hendrycks et al., 2022) is small, i.e. when the classifier is uncertain about the classification of a sample. Conversely to AD methods, which are applicable to OOD problems, OOD methods cannot be applied to AD setups due to the absence of in-distribution class labels. Note that the type of auxiliary supervision via OE we utilize in this paper hence differs from the kind of supervision applied in OOD, as we do not discriminate between different classes of normality but only between normal samples and auxiliary outliers.

Non-natural image AD Recently there has been increasing attention on image AD on "non-natural" images (e.g. medical images or technical images from manufacturing), where anomalies tend to be more subtle. For example, the MVTec-AD dataset consists of photos from manufacturing with, for instance, screws being normal and defective screws being anomalous (Bergmann et al., 2019). In this paper we instead focus on the common and well-established one vs. rest benchmarks with natural images, aiming to detect images of natural classes that are semantically different from the normal class. For other types of image data, random natural images from the web are likely not informative as OE. However, one can see that both transfer learning and OE *can* work well in other settings as many state-of-the-art methods on MVTec-AD rely on one of these. For instance, see Liznerski et al. (2021); Schlüter et al. (2021); Li et al. (2021) that employ OE in the form of synthetically generated anomalies and Defard et al. (2021); Gudovskiy et al. (2022); Roth et al. (2022) that use transfer learning-based methods.

3 Methods

In this section, we introduce the methods that we will use for our experimental evaluation. We first motivate why AD typically follows an unsupervised approach and is not viewed as a binary classification problem. Afterwards, we introduce deep one-class classification as well as CLIP (Radford et al., 2021) for zero-shot AD.

3.1 AD as a classification problem

Traditionally AD is understood as the problem of estimating the support (or level sets of the support) of the normal data-generating distribution. This is known as *density level set estimation* (Polonik, 1995; Tsybakov, 1997). This follows the assumption that normal data is concentrated whereas anomalies are not concentrated (Schölkopf & Smola, 2002). Steinwart et al. (2005) remark that density level set estimation can also be interpreted as a binary classification problem between the normal and an anomalous distribution. Many classic AD methods (e.g., kernel density estimation or one-class SVMs) implicitly assume the anomalies to follow a uniform distribution, i.e. they make an uninformative prior assumption on the anomalous distribution (Steinwart et al., 2005). These methods, as well as a binary classifier trained to discriminate between normal samples and uniform noise, are in fact asymptotically consistent density level set estimators (Steinwart et al., 2005; Vert & Vert, 2006). Practically, however, it is preferable to estimate the level set directly rather than classifying against uniform noise. Such a classification approach is particularly ineffective and inefficient in high dimensions since it would require massive amounts of noise samples to properly fill the sample space. As we show in our experiments, however, we find that this intuition does not seem to extend to deep anomaly detection on images.

3.2 Deep one-class classification

Deep one-class classification (Ruff et al., 2018) was introduced as a deep learning extension of the one-class classification approach to anomaly detection (Schölkopf et al., 2001; Tax, 2001). Deep SVDD (Ruff et al., 2018) is trained to map normal samples close to a center $\boldsymbol{c}$ in feature space, thereby following the concentration assumption (Schölkopf & Smola, 2002) mentioned above. For a neural network ϕ_θ with parameters θ, the Deep SVDD objective is given by $\min_\theta \frac{1}{n}\sum_{i=1}^{n} \|\phi_\theta(\boldsymbol{x}_i) - \boldsymbol{c}\|^2$. Ruff et al. (2020) proposed an extension of Deep SVDD that incorporates known anomalies, called *Deep Semi-supervised Anomaly Detection* (Deep SAD). Deep SAD trains a network to concentrate normal data near the center $\boldsymbol{c}$, while mapping anomalous samples away from that center. Hence, this follows an unsupervised OE approach to AD. Here, we present a principled modification of Deep SAD based on the cross-entropy loss, which we call *hypersphere classification*

(HSC). We find that this modification improves performance over Deep SAD and use it in our experiments as a prototypical representative of the unsupervised OE approach to AD.

Let $\mathcal{D} = \{(\boldsymbol{x}_1, y_1), \ldots, (\boldsymbol{x}_n, y_n)\}$ be a dataset with $\boldsymbol{x}_i \in \mathbb{R}^d$ and $y \in \{0, 1\}$ where $y = 1$ denotes normal and $y = 0$ anomalous instances. Let $\phi_\theta : \mathbb{R}^d \to \mathbb{R}^r$ be a neural network and $l : \mathbb{R}^r \to [0, 1]$ a function that maps the output to a probabilistic score. Then, the cross-entropy loss is given by

$$-\frac{1}{n}\sum_{i=1}^{n} y_i \log l(\phi_\theta(\boldsymbol{x}_i)) + (1-y_i)\log\left(1-l(\phi_\theta(\boldsymbol{x}_i))\right). \tag{1}$$

For a standard binary classifier, l is a linear layer followed by a sigmoid, and the decision region of the mapped samples $\phi_\theta(\boldsymbol{x}_1), \ldots, \phi_\theta(\boldsymbol{x}_n)$ is a half-space S. In this case, the preimage of S, $\phi_\theta^{-1}(S)$, is not guaranteed to be compact. To encourage the preimage of our normal decision region to be compact, we choose l to be a radial basis function: $l(\boldsymbol{z}) = \exp\left(-\|\boldsymbol{z}\|^2\right)$. In this case, (1) becomes

$$\frac{1}{n}\sum_{i=1}^{n} y_i \left\|\phi_\theta(\boldsymbol{x}_i)\right\|^2 - (1-y_i)\log\left(1-\exp\left(-\left\|\phi_\theta(\boldsymbol{x}_i)\right\|^2\right)\right).$$

If there are no anomalies, the HSC loss simplifies to $\frac{1}{n}\sum_{i=1}^{n}\left\|\phi_\theta(\boldsymbol{x}_i)\right\|^2$. For $\boldsymbol{c} = 0$, we thus recover Deep SVDD as a special case. Similar to Deep SVDD/SAD, we define our anomaly score as $s(\boldsymbol{x}) := \left\|\phi_\theta(\boldsymbol{x})\right\|^2$. Motivated by robust statistics (Hampel et al., 2005; Huber & Ronchetti, 2009), we also considered replacing l with radial functions that replace the squared-norm with more robust alternatives. Here, we found the pseudo-Huber loss (Charbonnier et al., 1997) to consistently yield the best results. We refer to Appendix E for a detailed analysis.

3.3 Contrastive language-image pre-training

To challenge the assumption that transfer learning approaches require OE for state-of-the-art detection performance, we consider a zero-shot approach to AD using the features of the contrastive language-image pre-training (CLIP) model (Radford et al., 2021). CLIP is trained on a massive dataset of 400 million (image, text) pairs with an objective to align corresponding pairs in feature space while keeping other pairs apart. Let $(\boldsymbol{x}_u, \boldsymbol{x}_v)$ denote an (image, text) pair, $\boldsymbol{u} = f_u(\boldsymbol{x}_u)$ and $\boldsymbol{v} = f_v(\boldsymbol{x}_v)$ the corresponding representations obtained by networks f_u and f_v, and consequently $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ the representations of the i-th data pair. CLIP uses the following losses: the text-to-image loss $l_i^{(v\to u)}$ and the image-to-text loss $l_i^{(u\to v)}$

$$l_i^{(v\to u)} = -\log\frac{\exp(\langle \boldsymbol{v}_i, \boldsymbol{u}_i\rangle\, e^\tau)}{\sum_{k=1}^{N}\exp(\langle \boldsymbol{v}_i, \boldsymbol{u}_k\rangle\, e^\tau)}, \quad l_i^{(u\to v)} = -\log\frac{\exp(\langle \boldsymbol{u}_i, \boldsymbol{v}_i\rangle\, e^\tau)}{\sum_{k=1}^{N}\exp(\langle \boldsymbol{u}_i, \boldsymbol{v}_k\rangle\, e^\tau)}, \tag{2}$$

where $\langle\cdot,\cdot\rangle$ denotes the cosine similarity and τ is a temperature parameter. CLIP's final objective is

$$\min_{f_u, f_v, \tau} \frac{1}{N}\sum_{i=1}^{N}\left(l_i^{(v\to u)} + l_i^{(u\to v)}\right)/2. \tag{3}$$

Radford et al. (2021) report that, without any fine-tuning on the downstream task, CLIP is able to outperform a fully supervised linear classifier with ResNet-50 features on several classification benchmarks, including ImageNet. For this, they use the names of the dataset classes as potential text candidates and predict the class whose text has the largest alignment with a given image. For out-of-distribution detection, Fort et al. (2021) explored using CLIP by taking the in-distribution and out-of-distribution text labels as candidates. We use CLIP in a similar way to perform zero-shot AD, where we use the text pair $(v_1, v_2) =$ ("a photo of a {NORMAL_CLASS}", "a photo of something"). For a test image $\boldsymbol{x}$, we compute its anomaly score as

$$s(\boldsymbol{x}) = \frac{\exp(\langle f_u(\boldsymbol{x}), f_v(v_2)\rangle \cdot 100)}{\sum_{k=1}^{2}\exp(\langle f_u(\boldsymbol{x}), f_v(v_k)\rangle \cdot 100)}. \tag{4}$$

Fine-tuning CLIP for AD with OE also is straightforward. We simply minimize the score (4) for normal samples and maximize it for OE samples. Since this corresponds to a binary cross-entropy loss, this is an instance of supervised OE, which we term "BCE with CLIP" (or just BCE-CL).

On the legitimacy of transfer learning for AD The use of transfer learning improved the performance of deep AD approaches significantly. Yet, it seems at least questionable whether the use of pre-trained models is experimentally sound since there may be a semantic overlap between the pre-training data and the anomalies seen at test time. While it is technically true that the AD model is still unsupervised and does not exploit knowledge of test samples, the reported performance on typical image AD benchmarks might be spurious as the model may not generalize well to other data. Radford et al. (2021), however, have investigated the overlap of data used for pre-training CLIP, where they remove all the data that overlaps with the downstream tasks and observe only an insignificant drop in performance on average. This suggests that our experiments and results based on CLIP reasonably explore generalization performance. We still want to raise awareness for this somewhat problematic trend in deep AD, however, for which we propose future research below.

4 Experimental setups

Before we present our results, we explain the experimental setup. In particular, we introduce the common one vs. rest benchmark, the CIFAR-10 and ImageNet datasets, and the state-of-the-art AD methods we consider in our experiments.

One vs. rest benchmark The one vs. rest evaluation protocol is a ubiquitous benchmark in the deep AD literature (Ruff et al., 2018; Golan & El-Yaniv, 2018; Hendrycks et al., 2019a;b; Ruff et al., 2020; Sohn et al., 2021; Deecke et al., 2021; Liznerski et al., 2021; Reiss et al., 2021). This benchmark constructs AD settings from classification datasets (e.g., CIFAR-10) by considering the "one" class (e.g., "airplane") as being normal and the "rest" classes (e.g., "automobile", "bird", ...) as being anomalous at test time. In each experiment, we train a model using only the training set of the normal class and samples from an OE set that are not contained in the anomaly classes of the benchmark. We use the same OE auxiliary datasets as suggested in previous works (Hendrycks et al., 2019a;b). To evaluate detection performance, we use the common Area Under the ROC curve (AUC) on the one vs. rest test sets. This is repeated over classes and multiple random seeds. For CIFAR-10, we consider all ten classes as our one vs. rest classes. For OE we use 80 Million Tiny Images (80MTI) (Torralba et al., 2008) with CIFAR-10 and CIFAR-100 images removed. This follows the experimental setup in Hendrycks et al. (2019b). For ImageNet (Deng et al., 2009), we use a subset of 30 classes as the one vs. rest classes. These are the same classes used in Hendrycks et al. (2019b). For OE we use ImageNet-22K with ImageNet-1K removed, again following Hendrycks et al. (2019b).

End-to-end methods We present results from end-to-end methods (without transfer learning) including all methods that achieve state-of-the-art performance on the one vs. rest benchmarks.

- Unsupervised: Shorthands for unsupervised methods are DSVDD (Ruff et al., 2018), GT (Golan & El-Yaniv, 2018), GT+ (Hendrycks et al., 2019b), and CSI (Tack et al., 2020).
- Unsupervised OE: We implement HSC from Section 3.2 and DSAD (Ruff et al., 2020) as unsupervised OE methods and also report the results from the unsupervised OE variant of GT+ (Hendrycks et al., 2019b).
- Supervised OE: BCE denotes a standard binary cross-entropy classifier. We also implement the Focal loss classifier with $\gamma = 2$ (Lin et al., 2017), a BCE variant for imbalanced classes that was also presented in Hendrycks et al. (2019b). Results from Hendrycks et al. (2019b) are marked with an asterisk as Focal*.

Transfer learning-based methods We consider the following transfer learning-based methods.

- Unsupervised: We implement a zero-shot anomaly detector using CLIP's feature space as described in Section 3.3 and use DN2 (Bergman et al., 2020) and PANDA (Reiss et al., 2021) as shorthands respectively for these unsupervised methods from the literature.
- Supervised OE: We consider a fine-tuned version of CLIP with a binary cross-entropy classifier, denoted as BCE-CL. ADIP (Deecke et al., 2021) and the supervised OE variant of PANDA (Reiss et al., 2021) are also included.

We provide network architecture and optimization details in Appendix G and investigate the impact of γ for the Focal loss in Appendix F. We report the mean AUC performance over all classes and seeds in the main paper. Individual results per class and method are in Appendix I. Each experiment is averaged over 10 seeds.

5 On the usefulness of samples in deep AD

Traditionally, AD methods utilize an unsupervised approach due to the assumption that it is impossible to characterize everything that is not normal. As mentioned above, later works introduced the idea of including a large collection of random images (OE) during training that serve as auxiliary examples of anomalousness to improve unsupervised AD methods (Hendrycks et al., 2019a). Models are trained with these OE samples using a loss that is essentially inverting a given unsupervised AD loss. We here look into whether an unsupervised OE approach, instead of a straightforward supervised OE approach using standard binary cross-entropy, is really necessary.

5.1 Supervised OE achieves state-of-the-art results

The first work on OE (Hendrycks et al., 2019a) applied an unsupervised OE method to two experimental setups: CIFAR-10 with 80MTI as OE and ImageNet-1K with ImageNet-22K as OE. Here we consider the same experimental setups with the basic HSC and BCE classifiers. The results are shown in Table 1. In this section we only consider end-to-end methods (i.e., no transfer learning).

Table 1: Mean AUC detection performance in % for end-to-end methods on the CIFAR-10 and ImageNet-1K one vs. rest benchmarks, using either 80MTI or ImageNet-22K (with 1K classes removed) as OE (* indicates results from the literature).

	Unsupervised			Unsupervised OE			Supervised OE		
	DSVDD*	GT+*	CSI*	GT+*	DSAD	HSC	Focal*	Focal	BCE
CIFAR-10	64.8	90.1	94.3	95.6	94.5	95.9	87.3	95.8	**96.1**
ImageNet	×	84.8	91.6	85.7	96.7	97.3	56.1	97.5	**97.7**

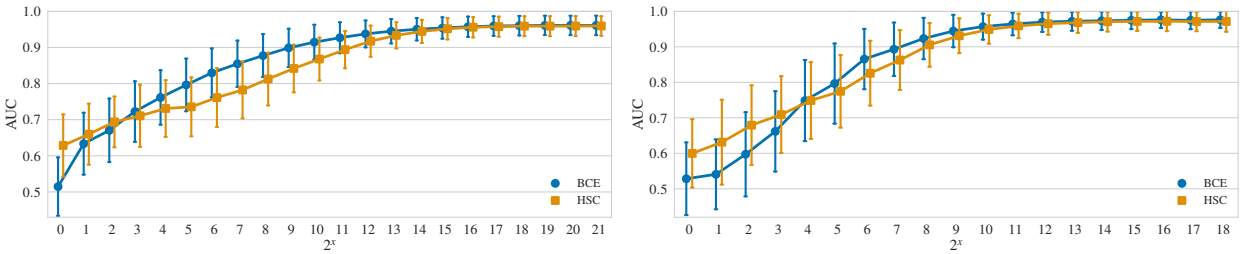
Discussion Surprisingly, we find that the choice of AD method has little impact on performance. On CIFAR-10, all unsupervised OE methods yield a comparable detection performance, while the supervised methods Focal and BCE show state-of-the-art performance, with BCE attaining the overall best mean AUC. On ImageNet, Deep SAD, HSC, Focal, and BCE all outperform the current state of the art (CSI*) by a significant margin. We are unsure as to why the Focal* performs so poorly in Hendrycks et al. (2019b) since their experimental code is not publicly available. Overall, our results show that a vanilla classifier (BCE) using OE outperforms all previous deep AD approaches on these ubiquitous benchmarks.

Our experiments seem to suggest that the inclusion of OE does not just improve AD performance, it also changes the problem into a typical supervised classification problem that does not require a compact decision boundary (cf. Figure 1). This stands in contrast to previous observations in shallow AD (Tax, 2001; Görnitz et al., 2013). While there is an abundance of OE data for the sorts of AD problems we investigate here, we want highlight that this data is likely not very helpful for AD problems where anomalies are more subtle. For example, in the realm of cancer detection, pictures of cats and trucks are not as useful for OE as medical images, which are not so widely available. Nonetheless, one may still have a few anomalous examples to incorporate during training. As one removes OE data, we would expect the behavior of an unsupervised OE approach to transition to the behavior of an unsupervised method that outperforms supervised OE. Our next experiment investigates this transition.

5.2 Few OE samples are sufficient

To investigate the effect of the size of the OE corpus, we perform experiments varying the OE training set size from just *one* sample ($2^0 = 1$) to using the maximal amount of OE data. Our results are in Figure 2. With sufficiently few samples, HSC outperforms BCE, which seems to indicate a regime where the unsupervised OE approach is advantageous. Interestingly, this regime seems to be quite small, with supervised OE needing only 8 samples to outperform unsupervised OE on CIFAR-10 and only ~ 32 samples for such a transition on ImageNet. Remarkably, BCE classification outperforms previous state-of-the-art methods on ImageNet with

only using ~ 256 OE samples. A training set with so few outliers and high-dimensional data represents an instance of skewed data (see Figure 1 and caption), where the common intuition suggests that supervised OE should *not* generalize well. As before, we see that the choice of the specific method seems negligible as soon as sufficient OE data is available.



(a) CIFAR-10 (over 10 classes $\times$ 10 seeds)

(b) ImageNet-1K (over 30 classes $\times$ 5 seeds)

Figure 2: Mean AUC detection performance in % on the CIFAR-10 and ImageNet-1K one vs. rest benchmarks when varying the number of 80MTI and ImageNet-22K OE samples respectively.

To measure the impact of data diversity we also experiment with varying the number of OE classes. We defer these results to Appendix A. In summary, we find that performance overall increases with OE data diversity, but already using just one OE class still performs relatively well.

While our results show that surprisingly few samples are needed to achieve competitive performance for end-to-end models, more recent methods in deep AD tend to use transfer learning. Interestingly, we find that one can achieve state-of-the-art AD performance using pre-trained models with *no training samples* (i.e., in a zero-shot setting).

5.3 Transfer learning enables zero-shot AD with state-of-the-art performance

More recent progress in deep AD has been achieved through transfer learning, which has further improved the state of the art on standard deep AD benchmarks. These methods use a network pre-trained on large datasets to provide rich representations as a starting basis for deep AD. Again, we look into how useful auxiliary data is for these algorithms. Here we investigate the effect of using OE data (vs. no OE data at all) and the implicit use of an extraneous dataset via pre-training. We also consider the situation where *no normal data* is used. Table 2 shows the results for transfer learning-based methods on CIFAR-10 and ImageNet. Note that, since DN2, PANDA, and ADIP employ ImageNet pre-trained networks, they cannot be compared on the ImageNet AD benchmark in a fair way.

Table 2: Mean AUC detection performance in % for methods with transfer learning on the CIFAR-10 and ImageNet-1K one vs. rest benchmark, using either 80MTI or ImageNet-22K (with the 1K classes removed) as OE (* indicates results from the literature).

	Unsupervised			Supervised OE		
	DN2*	PANDA*	CLIP	PANDA*	ADIP*	BCE-CL
CIFAR-10	92.5	96.2	98.5	98.9	99.1	**99.6**
ImageNet	$\times$	$\times$	99.88	$\times$	$\times$	**99.90**

Discussion These results highlight the remarkable efficacy of pre-training. Disregarding the CLIP results for now, we see that pre-training improves over previous unsupervised deep results. The results of PANDA indicates that OE is still useful for deep AD: OE provides additional information that is not learned in the pre-training task. Undoubtedly, the most interesting result here is the observation that one can outperform all previous state-of-the-art methods using no additional training from the benchmarks. In a zero-shot setting, CLIP (unsupervised) outperforms all previous end-to-end (see Table 1) and unsupervised methods. Fine-tuning CLIP with OE along with normal samples (BCE-CL), further marginally improves CLIP's results setting a new state of the art on the CIFAR-10 and ImageNet benchmarks.

Similar to results in other areas of deep learning (Bommasani et al., 2021), the use of large pre-trained networks offers an effective and convenient way to improve performance. Though transfer learning seems to be a natural endpoint for a certain class of deep AD problems, it still leaves a more general question about the difference between supervised and unsupervised approaches to deep AD in settings where transfer learning is not appropriate. This may happen when there are very subtle semantic novelties (Vaze et al., 2022), when one simply cannot use a pre-trained network; for example, when one must train a network from scratch due to architectural considerations (e.g., when one requires a smaller architecture due to hardware constraints), or when there are security concerns regarding the white box nature of pre-trained networks (Samek et al., 2021). Additionally, some AD techniques offer no obvious way to utilize a pre-trained network, like the recently introduced explainable one-class variant (Liznerski et al., 2021) or methods based on probabilistic models.

5.4 On the robustness of HSC vs. BCE

Our previous experiments (Section 5.2) have shown that, though end-to-end BCE overall outperforms HSC, an unsupervised OE approach is more effective when only very few (< 32) OE samples are available. This indicates a certain degree of robustness to the anomalous training samples for HSC. This robustness with regards to OE is likely inherited from the learning objectives of unsupervised and semi-supervised AD, which encourage the normal representations to be concentrated thereby avoiding the issue with skewed OE data (see Figure 1). So there seems to be a regime, albeit an extremely small one, where the intuition with skewed data in Figure 1 *does* hold.

To demonstrate the robustness of HSC when one has few OE samples we investigate the extreme case where an OE dataset consists of only one sample in more detail. How much can a single sample help or hinder the different approaches to deep AD? This is analogous to the experiments in Section 5.2 with the number of OE samples fixed to one. To investigate robustness, we search for the OE sample that gives the worst test AUC for each class and report the average AUC over all classes. While it is desirable for a method to be unaffected by detrimental OE examples, one would still want an AD method to exploit beneficial OE examples. To measure this trade-off, we perform an analogous experiment where we search for the best performing OE sample and again report the average test AUC over all classes.

As it is computationally prohibitive to test every possible OE sample, we utilize an evolutionary algorithm to attempt to minimize or maximize the class AUC. A detailed description of the algorithm can be found in Appendix H. To ensure that this optimization scheme does not find poor local minima, we also evaluate the AUC of many randomly chosen OE samples (see Appendix D). We find that the evolutionary algorithm almost always finds better optima. We report our results in Table 3. Figure 3 shows optimal OE examples on CIFAR-10 for the normal classes "ship" and "cat," and on ImageNet for "airplane" and "dragonfly."

Table 3: Mean AUC detection performance in % for the best and worst single OE samples on the CIFAR-10 AD benchmark with 80MTI as OE and on the ImageNet-1K AD benchmark with ImageNet-22K (without the 1K classes) as OE.

	CIFAR-10		ImageNet	
	HSC	BCE	HSC	BCE
Best OE	77.7	69.9	79.3	75.5
Worst OE	43.3	31.6	39.2	26.3

Discussion On both datasets, we observe that HSC performs better than BCE when using both the best and the worst OE samples. Looking at the samples chosen for the optima, there also appears to be more consistency within a setting (class, dataset, best or worst) for HSC. For example on "best" with "ship," HSC's images are all brownish outdoor photos, whereas the BCE samples vary from greenery to stylized text to an image containing mostly sky. This is likely due to the fact that HSC has, in some sense, an initial notion of anomalousness due to its unsupervised term. For instance, the most useful OE samples are those not already contained in this notion of anomalousness, resulting in HSC having a stable region for selecting OE samples that yield the greatest improvement. BCE lacks this notion so it can benefit from a large variety of OE samples. With HSC, choosing one optimal sample achieves roughly the same performance as using 32 random ones. Interestingly, it seems that near-distribution outliers are less useful as OE samples since samples with similar color patterns as the normal ones occur more often as the "worst" samples. Previous

(a) "ship" is normal (b) "cat" is normal (c) "airplane" is normal (d) "dragonfly" is normal

Figure 3: OE samples for CIFAR-10 with 80MTI as OE (a-b) and for ImageNet1k with ImageNet22k as OE (c-d). The first row shows normal samples, the next two rows show the best samples for HSC (top) and BCE (bottom), and the last two rows show the worst samples for HSC (top) and BCE (bottom).

works have found that near-distribution OE samples are useful for OE (Lee et al., 2018; Goyal et al., 2020), however, our results suggest that this may not hold when little OE data is available.

HSC focuses on low frequency features To gain further insight into the difference between BCE and HSC, we extend the previous experiments to include frequency-domain corruptions. This sort of analysis has been insightful in other works on deep learning (Yin et al., 2019). We find that HSC is again more robust than BCE since it's generally less affected by the frequency corruptions and tends to focus on low frequency signals in the input. We defer the results and discussion to Appendix B and also show some more examples for the best and worst single OE samples in Appendix I.

6 Broader impact

Anomaly detection methods on images may be deployed on tasks which have societal implications such as screening images or automated surveillance, and it is thus imperative that these tasks are done in a fair and transparent way. The use of OE is potentially harmful as there may be OE images biasing the model towards detecting certain entities as anomalous. Our paper aids in this since it demonstrates that no huge corpora are necessary, which enables a controlled selection of OE samples. Further, we have shown that HSC is more robust, and the fact that it chooses optimal OE samples that coincide with human intuition suggests that it is more interpretable than BCE, where the rationale for optimal OE samples is opaque. This makes HSC more suitable for critical applications. We trained some of our models with the 80MTI dataset, which is known to contain problematic data such as offensive labels, but were required to do so to be comparable with the previous line of research.

7 Conclusion

We presented surprising results that challenge common assumptions in AD. Neither does deep AD on natural images seem to require specialized AD methodologies nor huge amounts of Outlier Exposure (OE). A standard classifier outperforms all end-to-end methods on common AD benchmarks, for which it only needs 256 random OE samples on ImageNet and only one well-chosen OE sample for competitive performance. Using transfer learning, standard classifiers set a new state of the art on CIFAR-10 and, in a zero-shot setting *without using any additional training data,* on ImageNet. Despite the overall strong performance of standard classifiers, we find that semi-supervised one-class methods are more robust to the choice of OE when only few OE examples are available. Our results provide insights about deep AD that are useful for future research.

References

Davide Abati, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Latent space autoregression for novelty detection. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 481–490, 2019.

Samet Akcay, Amir Atapour-Abarghouei, and Toby P Breckon. GANomaly: Semi-supervised anomaly detection via adversarial training. In *Asian Conference on Computer Vision*, pp. 622–637, 2018.

Liron Bergman and Yedid Hoshen. Classification-based anomaly detection for general data. In *International Conference on Learning Representations*, 2020.

Liron Bergman, Niv Cohen, and Yedid Hoshen. Deep nearest neighbor anomaly detection. *arXiv preprint arXiv:2002.10445*, 2020.

Paul Bergmann, Michael Fauser, David Sattlegger, and Carsten Steger. Mvtec ad–a comprehensive real-world dataset for unsupervised anomaly detection. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 9592–9600, 2019.

Tobias Blickle and Lothar Thiele. A mathematical analysis of tournament selection. In *ICGA*, volume 95, pp. 9–15. Citeseer, 1995.

Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

Varun Chandola, Arindam Banerjee, and Vipin Kumar. Anomaly detection: A survey. *ACM Computing Surveys*, 41(3):1–58, 2009.

Pierre Charbonnier, Laure Blanc-Féraud, Gilles Aubert, and Michel Barlaud. Deterministic edge-preserving regularization in computed imaging. *IEEE Transactions on Image Processing*, 6(2):298–311, 1997.

Jinghui Chen, Saket Sathe, Charu C Aggarwal, and Deepak S Turaga. Outlier detection with autoencoder ensembles. In *SIAM International Conference on Data Mining*, pp. 90–98, 2017.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pp. 1597–1607. PMLR, 2020.

Lucas Deecke, Robert A Vandermeulen, Lukas Ruff, Stephan Mandt, and Marius Kloft. Image anomaly detection with generative adversarial networks. In *European Conference on Machine Learning and Principles and Practice of Knowledge Discovery in Databases*, pp. 3–17, 2018.

Lucas Deecke, Lukas Ruff, Robert A Vandermeulen, and Hakan Bilen. Transfer-based semantic anomaly detection. In *International Conference on Machine Learning*, pp. 2546–2558. PMLR, 2021.

Thomas Defard, Aleksandr Setkov, Angelique Loesch, and Romaric Audigier. Padim: a patch distribution modeling framework for anomaly detection and localization. In *International Conference on Pattern Recognition*, pp. 475–489. Springer, 2021.

Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 248–255, 2009.

Andrew F Emmott, Shubhomoy Das, Thomas Dietterich, Alan Fern, and Weng-Keen Wong. Systematic construction of anomaly detection benchmarks from real data. In *KDD 2013 Workshop on Outlier Detection and Description*, pp. 16–21, 2013.

Sarah Erfani, Mahsa Baktashmotlagh, Sutharshan Rajasegarar, Shanika Karunasekera, and Chris Leckie. R1SVM: A randomised nonlinear approach to large-acale anomaly detection. In *AAAI Conference on Artificial Intelligence*, pp. 432–438, 2015.

Sarah M Erfani, Sutharshan Rajasegarar, Shanika Karunasekera, and Christopher Leckie. High-dimensional and large-scale anomaly detection using a linear one-class SVM with deep learning. *Pattern Recognition*, 58:121–134, 2016.

Stanislav Fort, Jie Ren, and Balaji Lakshminarayanan. Exploring the limits of out-of-distribution detection. *Advances in Neural Information Processing Systems*, 34, 2021.

Félix-Antoine Fortin, François-Michel De Rainville, Marc-André Gardner Gardner, Marc Parizeau, and Christian Gagné. Deap: Evolutionary algorithms made easy. *The Journal of Machine Learning Research*, 13(1):2171–2175, 2012.

Spyros Gidaris, Praveer Singh, and Nikos Komodakis. Unsupervised representation learning by predicting image rotations. In *International Conference on Learning Representations*, 2018.

Izhak Golan and Ran El-Yaniv. Deep anomaly detection using geometric transformations. In *Advances in Neural Information Processing Systems*, pp. 9758–9769, 2018.

Nico Görnitz, Marius Kloft, Konrad Rieck, and Ulf Brefeld. Toward supervised anomaly detection. *Journal of Artificial Intelligence Research*, 46:235–262, 2013.

Sachin Goyal, Aditi Raghunathan, Moksh Jain, Harsha Vardhan Simhadri, and Prateek Jain. Drocc: Deep robust one-class classification. In *International Conference on Machine Learning*, pp. 3711–3721. PMLR, 2020.

Denis Gudovskiy, Shun Ishizaka, and Kazuki Kozuka. Cflow-ad: Real-time unsupervised anomaly detection with localization via conditional normalizing flows. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 98–107, 2022.

Frank R Hampel, Elvezio M Ronchetti, Peter J Rousseeuw, and Werner A Stahel. *Robust Statistics: The Approach Based on Influence Functions*. John Wiley & Sons, 2005.

Simon Hawkins, Hongxing He, Graham Williams, and Rohan Baxter. Outlier detection using replicator neural networks. In *International Conference on Data Warehousing and Knowledge Discovery*, volume 2454, pp. 170–180, 2002.

Dan Hendrycks, Mantas Mazeika, and Thomas G Dietterich. Deep anomaly detection with outlier exposure. In *International Conference on Learning Representations*, 2019a.

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*, pp. 15637–15648, 2019b.

Dan Hendrycks, Steven Basart, Mantas Mazeika, Andy Zou, Joseph Kwon, Mohammadreza Mostajabi, Jacob Steinhardt, and Dawn Song. Scaling out-of-distribution detection for real-world settings. In *International Conference on Machine Learning*, pp. 8759–8773. PMLR, 2022.

Fu Jie Huang and Yann LeCun. Large-scale learning with SVM and convolutional nets for generic object categorization. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 284–291, 2006.

Peter J Huber and Elvezio M Ronchetti. *Robust Statistics*. John Wiley & Sons, 2nd edition, 2009.

Sergey Ioffe and Christian Szegedy. Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift. In *International Conference on Machine Learning*, volume 37, pp. 448–456, 2015.

Ki Hyun Kim, Sangwoo Shim, Yongsub Lim, Jongseob Jeon, Jeongwoo Choi, Byungchan Kim, and Andre S. Yoon. RaPP: Novelty detection with reconstruction along projection pathway. In *International Conference on Learning Representations*, 2020.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015.

Hans-Peter Kriegel, Matthias Schubert, and Arthur Zimek. Angle-based outlier detection in high-dimensional data. In *International Conference on Knowledge Discovery & Data Mining*, pp. 444–452, 2008.

Yann LeCun, Bernhard E Boser, John S Denker, Donnie Henderson, Richard E Howard, Wayne E Hubbard, and Lawrence D Jackel. Handwritten digit recognition with a back-propagation network. In *NIPS*, pp. 396–404, 1990.

Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training confidence-calibrated classifiers for detecting out-of-distribution samples. In *International Conference on Learning Representations*, 2018.

Chun-Liang Li, Kihyuk Sohn, Jinsung Yoon, and Tomas Pfister. Cutpaste: Self-supervised learning for anomaly detection and localization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9664–9674, 2021.

Shiyu Liang, Yixuan Li, and R Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. In *International Conference on Learning Representations*, 2018.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *International Conference on Computer Vision*, pp. 2980–2988, 2017.

Philipp Liznerski, Lukas Ruff, Robert A. Vandermeulen, Billy Joe Franks, Marius Kloft, and Klaus-Robert Müller. Explainable deep one-class classification. In *International Conference on Learning Representations*, 2021.

Eric Nalisnick, Akihiro Matsukawa, Yee Whye Teh, Dilan Gorur, and Balaji Lakshminarayanan. Do deep generative models know what they don't know? In *International Conference on Learning Representations*, 2019.

Duc Tam Nguyen, Zhongyu Lou, Michael Klar, and Thomas Brox. Anomaly detection with multiple-hypotheses predictions. In *International Conference on Machine Learning*, volume 97, pp. 4800–4809, 2019.

Guansong Pang, Chunhua Shen, Longbing Cao, and Anton Van Den Hengel. Deep learning for anomaly detection: A review. *ACM Computing Surveys*, 54(2), 2021. doi: 10.1145/3439950.

Pramuditha Perera, Ramesh Nallapati, and Bing Xiang. OCGAN: One-class novelty detection using GANs with constrained latent representations. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2898–2906, 2019.

Wolfgang Polonik. Measuring mass concentrations and estimating density contour clusters-an excess mass approach. *The Annals of Statistics*, 23(3):855–881, 1995.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, volume 139, pp. 8748–8763, 2021.

Tal Reiss, Niv Cohen, Liron Bergman, and Yedid Hoshen. Panda: Adapting pretrained features for anomaly detection and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 2806–2814, 2021.

Karsten Roth, Latha Pemula, Joaquin Zepeda, Bernhard Schölkopf, Thomas Brox, and Peter Gehler. Towards total recall in industrial anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14318–14328, 2022.

Lukas Ruff, Robert A Vandermeulen, Nico Görnitz, Lucas Deecke, Shoaib A. Siddiqui, Alexander Binder, Emmanuel Müller, and Marius Kloft. Deep one-class classification. In *International Conference on Machine Learning*, volume 80, pp. 4390–4399, 2018.

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*, 2020.

Lukas Ruff, Jacob R Kauffmann, Robert A Vandermeulen, Grégoire Montavon, Wojciech Samek, Marius Kloft, Thomas G Dietterich, and Klaus-Robert Müller. A unifying review of deep and shallow anomaly detection. *Proceedings of the IEEE*, 109(5):756–795, 2021. doi: 10.1109/JPROC.2021.3052449.

Mayu Sakurada and Takehisa Yairi. Anomaly detection using autoencoders with nonlinear dimensionality reduction. In *2nd Workshop on Machine Learning for Sensory Data Analysis (MLSDA 2014)*, pp. 4–11, 2014.

Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J Anders, and Klaus-Robert Müller. Explaining deep neural networks and beyond: A review of methods and applications. *Proceedings of the IEEE*, 109(3):247–278, 2021.

Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Ursula Schmidt-Erfurth, and Georg Langs. Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In *International Conference on Information Processing in Medical Imaging*, pp. 146–157, 2017.

Thomas Schlegl, Philipp Seeböck, Sebastian M Waldstein, Georg Langs, and Ursula Schmidt-Erfurth. f-AnoGAN: Fast unsupervised anomaly detection with generative adversarial networks. *Medical Image Analysis*, 54:30–44, 2019.

Hannah M Schlüter, Jeremy Tan, Benjamin Hou, and Bernhard Kainz. Self-supervised out-of-distribution detection and localization with natural synthetic anomalies (nsa). *arXiv preprint arXiv:2109.15222*, 2021.

Bernhard Schölkopf and Alex J Smola. *Learning with Kernels*. MIT press, 2002.

Bernhard Schölkopf, John C Platt, John Shawe-Taylor, Alex J Smola, and Robert C Williamson. Estimating the support of a high-dimensional distribution. *Neural Computation*, 13(7):1443–1471, 2001.

Kihyuk Sohn, Chun-Liang Li, Jinsung Yoon, Minho Jin, and Tomas Pfister. Learning and evaluating representations for deep one-class classification. In *International Conference on Learning Representations*, 2021.

Ingo Steinwart, Don Hush, and Clint Scovel. A classification framework for anomaly detection. *Journal of Machine Learning Research*, 6(Feb):211–232, 2005.

Jihoon Tack, Sangwoo Mo, Jongheon Jeong, and Jinwoo Shin. CSI: Novelty detection via contrastive learning on distributionally shifted instances. In *Advances in Neural Information Processing Systems*, volume 33, pp. 11839–11852, 2020.

David Martinus Johannes Tax. *One-Class Classification*. PhD thesis, Delft University of Technology, 2001.

Antonio Torralba, Rob Fergus, and William T Freeman. 80 million tiny images: A large data set for nonparametric object and scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(11):1958–1970, 2008.

Alexandre B Tsybakov. On nonparametric estimation of density level sets. *The Annals of Statistics*, 25(3): 948–969, 1997.

Sagar Vaze, Kai Han, Andrea Vedaldi, and Andrew Zisserman. Open-set recognition: A good closed-set classifier is all you need. In *International Conference on Learning Representations*, 2022.

Régis Vert and Jean-Philippe Vert. Consistency and convergence rates of one-class SVMs and related algorithms. *Journal of Machine Learning Research*, 7(May):817–854, 2006.

Jingjing Wang, Sun Sun, and Yaoliang Yu. Multivariate triangular quantile maps for novelty detection. In *Advances in Neural Information Processing Systems*, pp. 5061–5072, 2019a.

Siqi Wang, Yijie Zeng, Xinwang Liu, En Zhu, Jianping Yin, Chuanfu Xu, and Marius Kloft. Effective end-to-end unsupervised outlier detection via inlier priority of discriminative network. In *Advances in Neural Information Processing Systems*, pp. 5960–5973, 2019b.

Dong Yin, Raphael Gontijo Lopes, Jon Shlens, Ekin Dogus Cubuk, and Justin Gilmer. A fourier perspective on model robustness in computer vision. In *Advances in Neural Information Processing Systems*, pp. 13255–13265, 2019.

Xinjie Yu and Mitsuo Gen. *Introduction to evolutionary algorithms*. Springer Science & Business Media, 2010.

Sergey Zagoruyko and Nikos Komodakis. Wide residual networks. In *British Machine Vision Conference*, 2016.

Houssam Zenati, Manon Romain, Chuan-Sheng Foo, Bruno Lecouat, and Vijay Chandrasekhar. Adversarially learned anomaly detection. In *IEEE International Conference on Data Mining*, pp. 727–736, 2018.

Chong Zhou and Randy C Paffenroth. Anomaly detection with robust deep autoencoders. In *International Conference on Knowledge Discovery & Data Mining*, pp. 665–674, 2017.

A Diversity of the Outlier Exposure data

Here we evaluate how data diversity influences detection performance for unsupervised and supervised OE, again comparing HSC to BCE. For this purpose, instead of 80MTI, we now use CIFAR-100 as OE varying the number of anomaly classes available for the CIFAR-10 benchmark. The OE data is varied by choosing k classes at random for each random seed and using the union of these classes as the OE dataset.

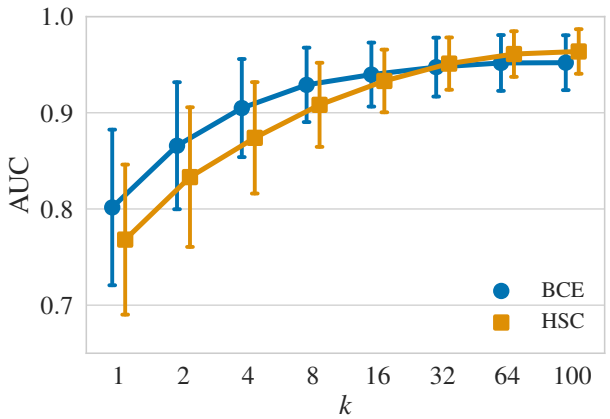


Figure 4: Mean AUC detection performance in % (over 10 classes with 10 seeds per class) on the CIFAR-10 with CIFAR-100 OE one vs. rest benchmark when varying the number of k classes that comprise the OE dataset.

The results are presented in Figure 4. As expected, the performance increases with the diversity of the OE dataset. Interestingly, drawing OE samples from just $k = 1$ class, i.e. binary classification between the normal class and a single OE class (which is not present as an anomaly class at test time!) already yields good detection performance on the CIFAR-10 benchmark. For example, training a standard classification network to discern between automobiles and beavers performs competitively as an automobile anomaly detector, even when no beavers are present as anomalies during test time.

Compared to Section 5.2 in the main paper, where we see a transition to BCE outperforming HSC at $2^3 = 8$ OE samples, we here see that with even one OE class we have passed this transition: there are already enough samples in a single class that BCE outperforms HSC. The takeaway is that OE sample diversity is not as important as one may expect, simply having many OE smaples suffices to enter the regime where BCE outperforms HSC. Again, with many samples, BCE and HSC's performances are comparable.

B HSC focuses on low frequency features

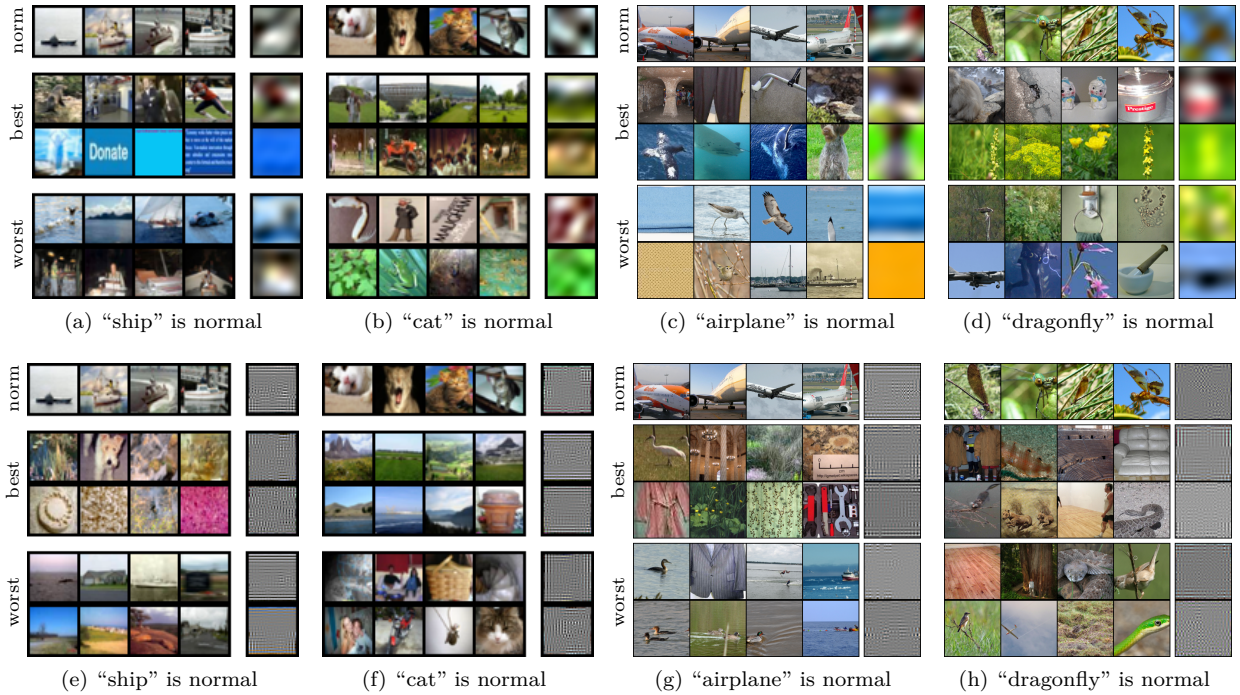
To gain further insight into the difference between BCE and HSC and why the best (and worst) found OE samples are quite different for these methods, we repeat the experiment from Section 5.4 in the main paper with frequency-domain corruptions. That is, we low-pass-filter (LPF) or high-pass-filter (HPF) the *entire* dataset (training, testing, and OE) and then proceed exactly as we did in Section 5.4. An LPF removes all higher frequencies and preserves only global information such as a scenery's color. This roughly corresponds to blurring images. An HPF removes all lower frequencies and roughly corresponds to edge detection. The AUC scores are in Table 4. Figure 5 contains examples of the best and worst OE samples and also shows filtered examples. We provide details on the filter implementation in Appendix C, where we evaluate the general performance of BCE and HSC with varying filter magnitudes.

HSC seems to be more robust than BCE since it's generally less affected by the frequency corruptions. Focusing on CIFAR-10, we see that for HSC frequency corruption makes little change to performance except for the Best OE HPF experiment, where HSC's performance is significantly lower than in the LPF variant, causing HSC to behave similarly to BCE. This may have some implication that when useful signal is contained

in the OE sample, corresponding to the "Best OE" experiments, it is concentrated in the low frequency spectrum. It seems as though BCE is not capable of exploiting this data. Interestingly it was also found in Yin et al. (2019) that low frequency features tend do be more robust ("low frequency bias results in improved robustness to corruptions"). On ImageNet we observe that HSC again performs better on the LPF experiments.

Table 4: Mean AUC detection performance in % for the best and worst single OE samples on CIFAR-10 using 80MTI as OE and on ImageNet-1K using ImageNet-22K (with the 1K classes removed) as OE. All images have been either high-pass-filtered (HPF) or low-pass-filtered (LPF) both during training and testing. Arrows indicate the change compared to Table 3.

	CIFAR-10		ImageNet	
	HSC	BCE	HSC	BCE
Best OE LPF	77.5→	68.5→	77.2↘	73.8→
Worst OE LPF	44.1→	31.1→	44.6 ↑	26.1→
Best OE HPF	68.8 ↓	66.4↘	75.0↘	77.3→
Worst OE HPF	43.6→	38.0 ↑	44.1 ↑	27.9→



(a) "ship" is normal (b) "cat" is normal (c) "airplane" is normal (d) "dragonfly" is normal

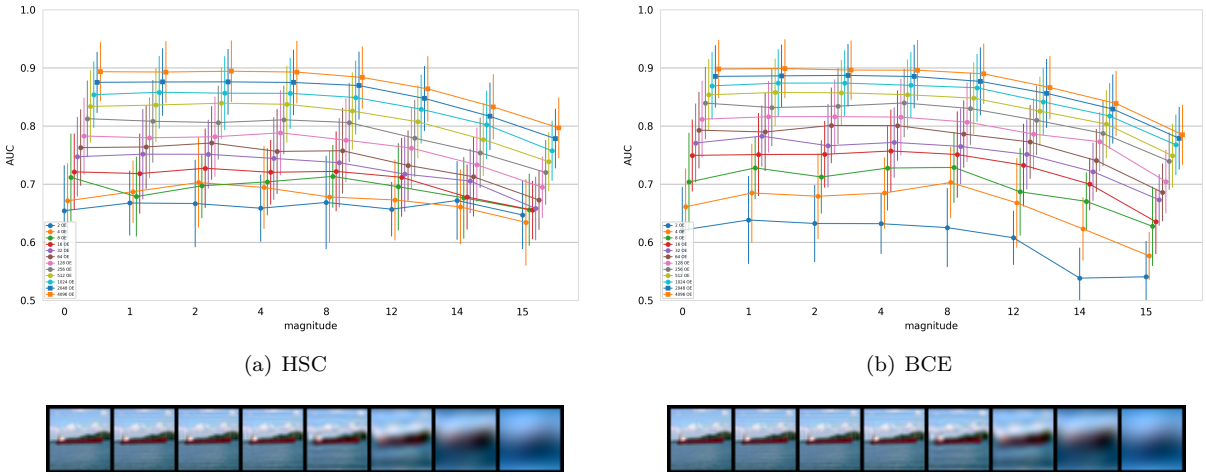
(e) "ship" is normal (f) "cat" is normal (g) "airplane" is normal (h) "dragonfly" is normal

Figure 5: OE samples for low-pass-filtered (a-d) and high-pass-filtered (e-h) versions of CIFAR-10 with 80MTI as OE (a,b,e,f) and ImageNet1k with ImageNet22k as OE (c,d,g,h). In each figure, the first row shows normal samples, the next two rows the best samples for HSC (top) and BCE (bottom), and the last two rows the worst samples for HSC (top) and BCE (bottom). The last column shows the filtered version of the images, which is what the network sees during training and testing.

C Frequency sensitivity analysis

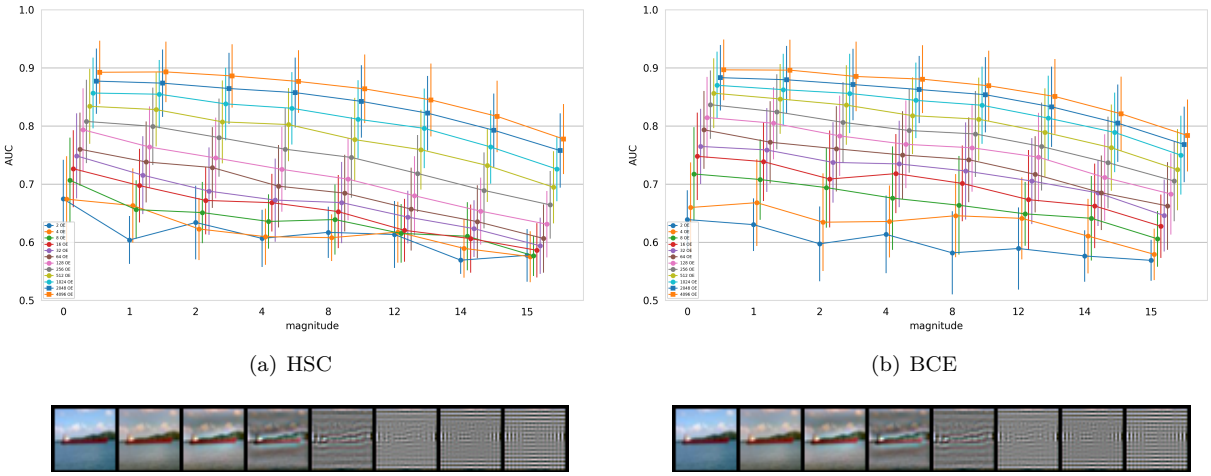
To understand why so few OE samples are that effective, we investigate the gemeral detection performance of HSC and end-to-end BCE for images with limited frequency spectrums. Analog to Appendix B, we either

low-pass (LPF) or high-pass-filter (HPF) all images, both during training and testing, both normal and anomalous samples. We train and evaluate either HSC or BCE for varying OE dataset sizes and different magnitudes of filters. Note that, due to computational constraints, we decrease the number of epochs and restrict the augmentations for the frequency experiments. Figures 6, 7, 8, and 9 show the results on LPF CIFAR-10, HPF CIFAR-10, LPF ImageNet, and HPF Imagenet, respectively. Each point in the plots corresponds to the mean AUC detection performance over all classes and 2 seeds per class. Different colors/markers correspond to different amounts of random OE samples used. The magnitudes shown on the horizontal axes correspond to the number of rows and columns removed in the frequency domain. For example, an LPF with a magnitude of m sets the first and last m rows and the first and last m columns of the Fourier-transformed image to zero, before applying the inverse Fourier transformation. A magnitude of 0 corresponds to unfiltered images, a magnitude of 15 on CIFFAR-10 images (which have a resolution of 32×32) corresponds to filtered images where just 4 "pixels" in the center remain in the frequency domain. Similarly, an HPF with a magnitude of m sets the center of size $m \times m$ to zero. The extended OE robustness experiments in Appendix B and Appendix D use a magnitude of 14 for CIFAR-10 and a magnitude of 110 for ImageNet.



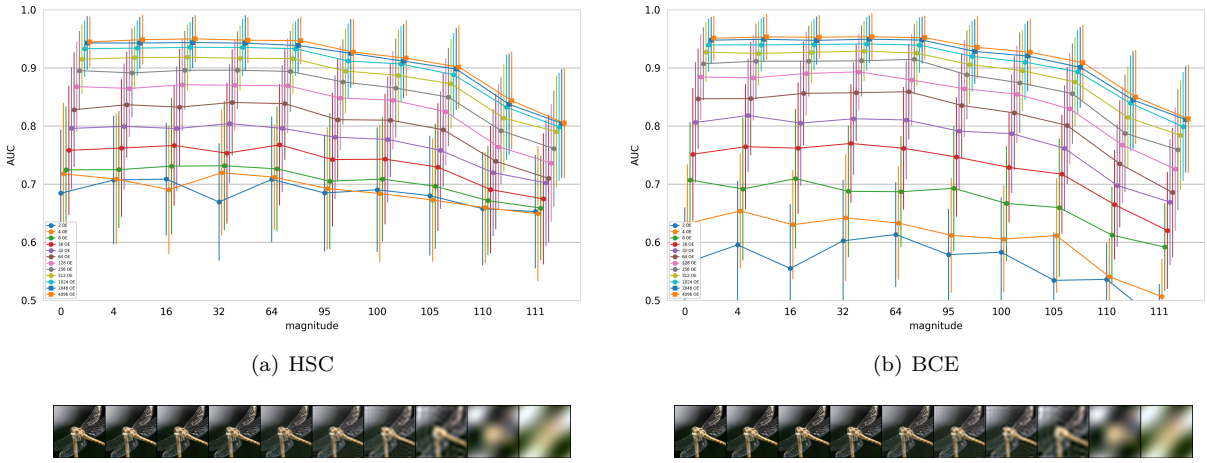
(a) HSC (b) BCE

Figure 6: LPF CIFAR-10 with 80MTI OE AD benchmark.



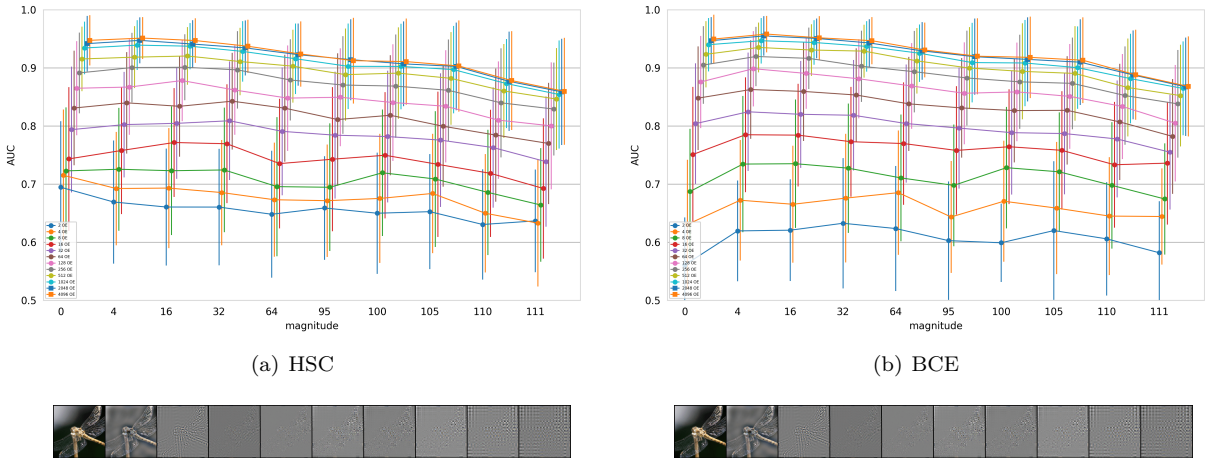
(a) HSC (b) BCE

Figure 7: HPF CIFAR-10 with 80MTI OE AD benchmark.



(a) HSC (b) BCE

Figure 8: LPF ImageNet1k with ImageNet22k (with ImageNet1k removed) OE AD benchmark.



(a) HSC (b) BCE

Figure 9: HPF ImageNet1k with ImageNet22k (with ImageNet1k removed) OE AD benchmark.

D OE robustness results with random search

In Section 5.4 we use an evolutionary algorithm to find the best and worst single OE samples for AD. The main difference between an evolutionary algorithm and a pure random search is that the former "fine-tunes" samples by picking the next ones depending on the best-performing previous ones. However, the algorithm also explores completely new samples. To show that this is the case and the algorithm doesn't quickly converge to a local optimum, we here present results using a pure random search for the best single OE samples and compare those to the ones in the main paper.

Table 5 shows the mean AUC over ten classes (analogue to Table 3 in the main paper and Table 4 in Appendix B). We see that pure random search consistently yields slightly worse samples on CIFAR-10 and mostly very similar performing samples on ImageNet. There is a minor exception for unfiltered images with BCE, where random search found on average 2% better performing single OE samples, resulting in the same conclusion, though. Figure 10 shows examples for the best samples found via a random search for unfiltered and filtered images. We see that the results are quite similar to those found via the evolutionary algorithm.

Table 5: Mean AUC detection performance in % for the best single OE samples on the CIFAR-10 AD benchmark using 80MTI as OE and on the ImageNet-1K AD benchmark using ImageNet-22K (with the 1K classes removed) as OE. All images have been either unfiltered, high-pass-filtered (HPF), or low-pass-filtered (LPF). Here we use pure random search instead of an evolutionary algorithm.

	CIFAR-10		ImageNet	
	HSC	BCE	HSC	BCE
Best OE	76.5	68.7	79.5	77.8
Best OE LPF	77.0	66.6	76.7	73.3
Best OE HPF	68.1	65.9	75.0	77.2

(a) “ship” is normal (b) “cat” is normal (c) “airplane” is normal (d) “dragonfly” is normal

(e) “ship” is normal (f) “cat” is normal (g) “airplane” is normal (h) “dragonfly” is normal

(i) “ship” is normal (j) “cat” is normal (k) “airplane” is normal (l) “dragonfly” is normal

Figure 10: Best OE samples for CIFAR-10 with 80MTI as OE (a-b, e-f, i-j) and ImageNet1k with ImageNet22k as OE (c-d, g-h, k-l). In each figure, the first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom). In Figures (e-h), all samples (train, test, anomalous, and normal) are low-pass-filtered, in Figures (i-l) high-pass-filtered. These figures also contain a separate last column showing the filtered version of the images, which is what the network sees during training or testing. Here we use pure random search instead of an evolutionary algorithm.

E Hypersphere Classifier sensitivity analysis

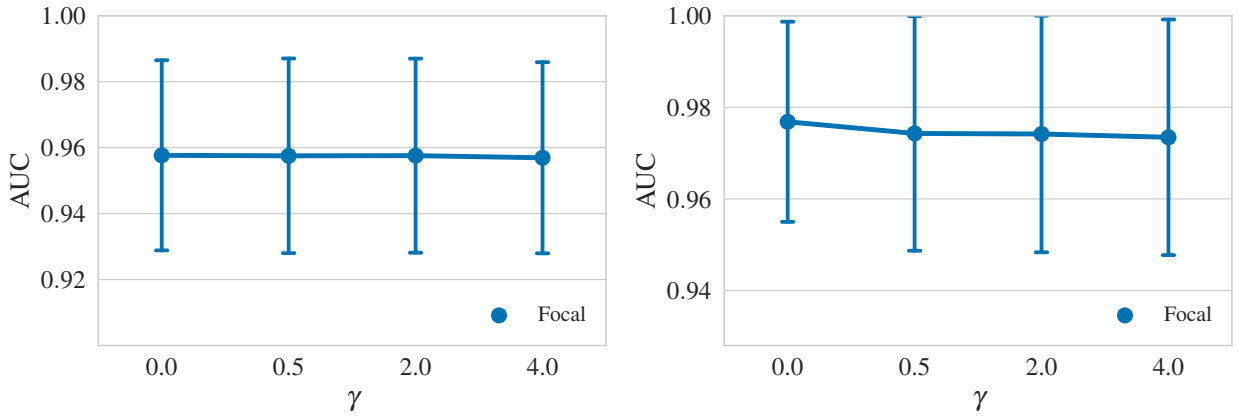
In this section, we show results for the Hypersphere Classifier (HSC) (Section 3.2) when varying the radial function $l(\boldsymbol{z}) = \exp(-h(\boldsymbol{z}))$. For this, we run the CIFAR-10 one vs. rest benchmark with 80MTI as OE, as presented in Table 1 in the main paper, for different functions $h : \mathbb{R}^r \to [0, \infty), \boldsymbol{z} \mapsto h(\boldsymbol{z})$. We also alter training to be with or without data augmentation in these experiments. The results are presented in Table 6. We see that data augmentation leads to an improvement in performance even in the case where we have the full 80MTI dataset as OE. HSC shows the overall best performance with data augmentation and using the robust Pseudo-Huber loss $h(\boldsymbol{z}) = \sqrt{\|\boldsymbol{z}\|^2 + 1} - 1$.

Table 6: Mean AUC detection performance in % (over 10 seeds) on the CIFAR-10 one vs. rest benchmark using 80MTI as OE for different choices of $h(\boldsymbol{z})$ in the radial function l of HSC.

Data augment.	$\|\boldsymbol{z}\|_1$	$\|\boldsymbol{z}\|_2$	$\|\boldsymbol{z}\|_2^2$	$\sqrt{\|\boldsymbol{z}\|^2+1}-1$
w/o	90.6	92.3	89.1	91.8
w/	92.5	94.1	94.5	96.1

F Focal Loss with varying γ

Here we include results showing how mean AUC detection performance changes with γ on the Focal loss. Since we balance every batch to contain 128 normal and 128 OE samples during training, we set the weighting factor α to be $\alpha = 0.5$ (Lin et al., 2017). Again, note that $\gamma = 0$ corresponds to standard binary cross entropy. Figure 11 shows that mean AUC performance is insensitive to the choice of γ on the CIFAR-10 and ImageNet-1K one vs. rest benchmarks.



(a) CIFAR-10 (b) ImageNet-1K

Figure 11: Focal loss detection performance in mean AUC in % when varying γ on the CIFAR-10 with 80MTI OE (a) and ImageNet-1K with ImageNet-22K OE (b) one vs. rest benchmarks.

G Network architectures and optimization

We provide details on the network architectures and optimization below, where we distinguish between end-to-end methods and the ones that use transfer learning (CLIP-based).

G.1 End-to-end methods

We always use the same underlying network ϕ_θ in each experimental setting for our HSC, Deep SAD, Focal, and BCE implementations to control architectural effects. For Focal and BCE, the output of the network ϕ_θ is followed by a linear layer with sigmoid activation. For the experiments on CIFAR-10 we use standard LeNet-style networks (LeCun et al., 1990) having three convolutional layers followed by two fully connected layers. We use batch normalization (Ioffe & Szegedy, 2015) and (leaky) ReLU activations in these networks. For our experiments on ImageNet we use the same WideResNet (Zagoruyko & Komodakis, 2016) as Hendrycks et al. (2019b), which has ResNet-18 as its architectural backbone. We use Adam (Kingma & Ba, 2015) for optimization and balance every batch to contain 128 normal and 128 OE samples during training. For data augmentation, we use standard color jitter, random cropping, horizontal flipping, and Gaussian pixel noise. Due to computational constraints, there is an exception for the OE robustness experiments (Section 5.4) and

the frequency analysis (Appendix B, Appendix C), where we use no augmentations for CIFAR-10 and only random cropping for ImageNet. We provide further dataset-specific details below.

CIFAR-10 On CIFAR-10, we use LeNet-style networks having three convolutional layers and two fully connected layers. Each convolutional layer is followed by batch normalization, a leaky ReLU activation, and max-pooling. The first fully connected layer is followed by batch normalization, and a leaky ReLU activation, while the last layer is just a linear transformation. The number of kernels in the convolutional layers are, from first to last: 32-64-128. The fully connected layers have 512-256 units respectively. We use Adam (Kingma & Ba, 2015) for optimization and balance every batch to contain 128 normal and 128 OE samples during training. We train for 200 epochs starting with a learning rate of $\eta = 0.001$ and have learning rate milestones at 100 and 150 epochs. The learning rate is reduced by a factor of 10 at every milestone. For the OE robustness experiments (Section 5.4) and the frequency analysis (Appendix B, Appendix C), we trained for 30 epochs with a milestone at 25 instead.

ImageNet On ImageNet, we use exactly the same WideResNet (Zagoruyko & Komodakis, 2016) as was used in Hendrycks et al. (2019b), which has a ResNet-18 as architectural backbone. We use Adam (Kingma & Ba, 2015) for optimization and balance every batch to contain 128 normal and 128 OE samples during training. We train for 150 epochs starting with a learning rate of $\eta = 0.001$ and milestones at epochs 100 and 125. The learning rate is reduced by a factor of 10 at every milestone. For the OE robustness experiments (Section 5.4) and the frequency analysis (Appendix B, Appendix C), we trained for 30 epochs with a milestone at 25 instead.

G.2 CLIP

Apart from the following changes, we use the same setup for our CLIP-based implementations as for the end-to-end methods. One of the changes is that we use the pre-trained CLIP network architecture (Radford et al., 2021). For fine-tuning CLIP with a BEC classifier, we use SGD with Nesterov momentum instead of ADAM, train for 80 epochs starting with a learning rate of $\eta = 0.0001$, and have learning rate milestones at 50, 60, 70, and 75 epochs. The learning rate is reduced by a factor of 10 at every milestone. This applies completely to ImageNet. For CIFAR-10, we do the same but start with a learning rate of $\eta = 0.00002$ instead.

H The evolutionary algorithm for finding OE samples

In Section 5.4 we search for a sample in the OE dataset that is either particularly harmful or useful for the AD model. A common approach to perform a discrete search in a large set is to use evolutionary algorithms (Yu & Gen, 2010; Fortin et al., 2012). We here stick to a simple version that uses tournament selection (Blickle & Thiele, 1995) with three competitors. In our scenario, the individuals are images of the OE dataset and randomly initialized. Mutating an individual works by first sampling random images from the OE set and then randomly picking one among those 50 in that subset that have the least L^2 distance to the individual. Mating works similarly, it picks one with the least L^2 distance to both parents. The fitness of an individual is the test AUC of an AD model trained with the full normal dataset and just the individual as the only training outlier. The algorithm's objective is to either maximize ("best" OE samples) or minimize ("worst" OE samples) the average fitness of its final generation.

This experimental setup is exceptionally computationally expensive, with the evaluation of a single individual's fitness requiring a full training of a neural network. One search for one class takes 9 days on a DGX A100 GPU. Because of this we decrease the number of training epochs, restrict the augmentations, and average over just two different random seeds to evaluate an individual's fitness (see Appendix G). On ImageNet, we only consider the first 10 classes (see Appendix I) out of the 30 that are used in Hendrycks et al. (2019a), but evaluate these with the full 30-classes one vs. rest benchmark.

Algorithm 1 provides a detailed explanation of the evolutionary algorithm. For the sake of readability, we fix the evolutionary hyperparameters to the values used in our experiments. We set the generation size to 64, the number of generations to 50, the tournament size to 3, the mating chance to 0.05, the mutation chance to 0.55, the initial candidate pool size to 10000, and the final candidate pool size to 50. We apply this algorithm

to each class separately, thereby changing the normal set X and test set X_{test}. For finding the worst single OE samples instead of the best ones, we change the selection procedure to simply replace d_{temp} with the worst in $\text{RND}(D', 3)$ instead of the best.

Algorithm 1 Evolve OE samples

Input: AD model ϕ, OE dataset D, normal train set X, full test set X_{test}.
Output: A collection of optimal single training outliers (individuals) D'.
Define:

COIN(p): returns TRUE with chance p, else FALSE.
RND(D, k): selects k samples in set D randomly.
TRAIN(ϕ, d): randomly initializes ϕ's weights, then trains ϕ with normal dataset X and the single training outlier d as OE.
EVAL(ϕ): computes and returns the test AUC of ϕ using the one vs. rest approach on X_{test}.

Algorithm:

```
# Initialize:
D' ← RND(D, 64)
for all d' ∈ D' do
    TRAIN(φ, d')
    a ← EVAL(φ)
    Set fitness of d' to a
end for

for 50 iterations do
    # Select:
    D_temp ← D'
    for all d_temp ∈ D_temp do
        Replace d_temp in D_temp with the best in RND(D', 3).    [best according to fitness]
    end for
    D' ← D_temp

    # Mate:
    for all i ∈ [0, 2, 4, ..., |D'|] do
        if COIN(0.05) then
            d'_1 ← D'[i]
            d'_2 ← D'[i + 1]
            for all j ∈ [0, 1] do
                P ← RND(D, 10000)
                sort(P) according to (‖p − d'_1‖² + ‖p − d'_2‖²) for every p ∈ P.
                P ← P[: 50], get the 50 samples with least average distance to d'_1 and d'_2.
                Replace D'[i + j] with RND(P, 1).
            end for
        end if
    end for

    # Mutate:
    for all d' ∈ D' do
        if COIN(0.55) then
            P ← RND(D, 10000)
            sort(P) according to ‖p − d'‖² for every p ∈ P.
            P ← P[: 50], get the 50 samples with least distance to d'.
            Replace d' in D' with RND(P, 1).
        end if
    end for

    # Evaluate:
    for all d' ∈ D' do
        TRAIN(φ, d')
        a ← EVAL(φ)
        Set fitness of d' to a
    end for

end for
```

I Results on individual classes

All our experiments use the standard one vs. rest AD benchmark (Section 4) to evaluate the performance of AD models. Due to space constraints, we report results averaged over all classes in the main paper. Here, we provide full class-wise results for all experiments, including also figures for each class of the varying amount of OE (Section 5.2) and OE robustness (Section 5.4, Appendix B) experiments.

I.1 Image AD with end-to-end methods

For the CIFAR-10 one vs.-rest benchmark experiment from Section 5.1, we report the results for all individual classes and end-to-end methods in Table 7. Additionally, for the CIFAR-10 and the ImageNet-1K one vs.-rest benchmark experiments, we report the results with standard deviations for our implementations in Tables 8 and 9, respectively.

Table 7: Mean AUC detection performance in % (over 10 seeds) for all individual classes and end-to-end methods on the CIFAR-10 one vs. rest benchmark with 80MTI OE from Section 5.1.

	Unsupervised					Unsupervised OE			Supervised OE		
Class	SVDD*	DSVDD*	GT*	GT+*	CSI*	GT+*	DSAD	HSC	Focal*	Focal	BCE
Airplane	65.6	61.7	74.7	77.5	89.9	90.4	94.2	96.3	87.6	95.9	**96.4**
Automobile	40.9	65.9	95.7	96.9	99.1	**99.3**	98.1	98.7	93.9	98.7	98.8
Bird	65.3	50.8	78.1	87.3	93.1	**93.7**	89.8	92.7	78.6	92.3	93.0
Cat	50.1	59.1	72.4	80.9	86.4	88.1	87.4	89.8	79.9	88.8	**90.0**
Deer	75.2	60.9	87.8	92.7	93.9	**97.4**	95.0	96.6	81.7	96.6	97.1
Dog	51.2	65.7	87.8	90.2	93.2	**94.3**	93.0	94.2	85.6	94.1	94.2
Frog	71.8	67.7	83.4	90.9	95.1	97.1	96.9	97.9	93.3	97.8	**98.0**
Horse	51.2	67.3	95.5	96.5	98.7	**98.8**	96.8	97.6	87.9	97.6	97.6
Ship	67.9	75.9	93.3	95.2	97.9	**98.7**	97.1	98.2	92.6	98.0	98.1
Truck	48.5	73.1	91.3	93.3	95.5	**98.5**	96.2	97.4	92.1	97.5	97.7
Mean AUC	58.8	64.8	86.0	90.1	94.3	95.6	94.5	95.9	87.3	95.8	**96.1**

Table 8: Mean AUC detection performance in % (over 10 seeds) with standard deviations for all individual classes for our end-to-end implementations on the CIFAR-10 one vs. rest benchmark with 80MTI OE from Section 5.1.

	Unsupervised OE		Supervised OE	
Class	DSAD	HSC	Focal	BCE
Airplane	94.2 ± 0.34	96.3 ± 0.13	95.9 ± 0.11	96.4 ± 0.17
Automobile	98.1 ± 0.19	98.7 ± 0.07	98.7 ± 0.09	98.8 ± 0.06
Bird	89.8 ± 0.54	92.7 ± 0.27	92.3 ± 0.32	93.0 ± 0.14
Cat	87.4 ± 0.38	89.8 ± 0.27	88.8 ± 0.33	90.0 ± 0.27
Deer	95.0 ± 0.22	96.6 ± 0.17	96.6 ± 0.10	97.1 ± 0.10
Dog	93.0 ± 0.30	94.2 ± 0.13	94.1 ± 0.21	94.2 ± 0.12
Frog	96.9 ± 0.22	97.9 ± 0.08	97.8 ± 0.07	98.0 ± 0.09
Horse	96.8 ± 0.15	97.6 ± 0.10	97.6 ± 0.16	97.6 ± 0.09
Ship	97.1 ± 0.21	98.2 ± 0.09	98.0 ± 0.11	98.1 ± 0.08
Truck	96.2 ± 0.22	97.4 ± 0.13	97.5 ± 0.12	97.7 ± 0.16
Mean AUC	94.5 ± 3.30	95.9 ± 2.68	95.8 ± 2.97	96.1 ± 2.71

Table 9: Mean AUC detection performance in % (over 10 seeds) for all individual classes for our end-to-end implementations on the ImageNet-1K one vs. rest benchmark with ImageNet-22K OE from Section 5.1. Note that for GT+*, Focal*, and CSI* as reported in Table 1 in the main paper, Hendrycks et al. (2019b) and Tack et al. (2020) do not provide results on a per class basis.

	Unsupervised OE		Supervised OE	
Class	DSAD	HSC	Focal	BCE
Acorn	98.5 ± 0.28	98.8 ± 0.42	99.0 ± 0.15	99.0 ± 0.19
Airliner	99.6 ± 0.16	99.8 ± 0.10	99.9 ± 0.02	99.8 ± 0.04
Ambulance	99.0 ± 0.13	99.8 ± 0.13	99.2 ± 0.14	99.9 ± 0.07
American alligator	92.9 ± 1.06	98.0 ± 0.32	94.7 ± 0.67	98.2 ± 0.27
Banjo	97.0 ± 0.51	98.2 ± 0.41	97.0 ± 0.33	98.7 ± 0.22
Barn	98.5 ± 0.29	99.8 ± 0.05	98.7 ± 0.24	99.8 ± 0.08
Bikini	96.5 ± 0.84	98.6 ± 0.57	97.2 ± 0.89	99.1 ± 0.30
Digital clock	99.4 ± 0.33	96.8 ± 0.79	99.8 ± 0.03	97.2 ± 0.29
Dragonfly	98.8 ± 0.28	98.4 ± 0.16	99.1 ± 0.21	98.3 ± 0.04
Dumbbell	93.0 ± 0.53	91.6 ± 0.88	94.0 ± 0.04	92.6 ± 0.97
Forklift	90.6 ± 1.43	99.1 ± 0.33	94.2 ± 0.90	99.5 ± 0.09
Goblet	92.4 ± 1.05	93.8 ± 0.38	93.8 ± 0.27	94.7 ± 1.43
Grand piano	99.7 ± 0.06	97.4 ± 0.37	99.9 ± 0.04	97.6 ± 0.34
Hotdog	95.9 ± 2.01	98.5 ± 0.34	97.2 ± 0.05	98.8 ± 0.34
Hourglass	96.3 ± 0.37	96.9 ± 0.26	97.5 ± 0.17	97.6 ± 0.48
Manhole cover	98.5 ± 0.29	99.6 ± 0.34	99.2 ± 0.09	99.8 ± 0.01
Mosque	98.6 ± 0.29	99.1 ± 0.26	98.9 ± 0.30	99.3 ± 0.15
Nail	92.8 ± 0.80	94.0 ± 0.76	93.5 ± 0.32	94.5 ± 1.37
Parking meter	98.5 ± 0.29	93.3 ± 1.64	99.3 ± 0.04	94.7 ± 0.76
Pillow	99.3 ± 0.14	94.0 ± 0.47	99.2 ± 0.14	94.2 ± 0.42
Revolver	98.2 ± 0.30	97.6 ± 0.25	98.6 ± 0.11	97.7 ± 0.68
Rotary dial telephone	90.4 ± 1.99	97.7 ± 0.50	92.2 ± 0.33	98.3 ± 0.75
Schooner	99.1 ± 0.18	99.2 ± 0.20	99.6 ± 0.02	99.1 ± 0.26
Snowmobile	97.7 ± 0.86	99.0 ± 0.22	98.1 ± 0.15	99.1 ± 0.25
Soccer ball	97.3 ± 1.70	92.9 ± 1.18	98.6 ± 0.13	93.6 ± 0.61
Stingray	99.3 ± 0.20	99.1 ± 0.33	99.7 ± 0.04	99.2 ± 0.10
Strawberry	97.7 ± 0.64	99.1 ± 0.20	99.1 ± 0.03	99.2 ± 0.22
Tank	97.3 ± 0.51	98.6 ± 0.18	97.3 ± 0.47	98.9 ± 0.13
Toaster	97.7 ± 0.56	92.2 ± 0.78	98.3 ± 0.05	92.2 ± 0.65
Volcano	89.6 ± 0.44	99.5 ± 0.09	91.6 ± 0.90	99.4 ± 0.19
Mean AUC	96.7 ± 2.98	97.3 ± 2.53	97.5 ± 2.43	97.7 ± 2.34

I.2 Image AD with transfer learning-based methods

For the CIFAR-10 one vs.-rest benchmark experiment from Section 5.3, we report the results for all individual classes and pre-trained methods in Table 10. Additionally, for the CIFAR-10 and the ImageNet-1K one vs.-rest benchmark experiments, we report the results with standard deviations for our implementations in Tables 11 and 12, respectively.

Table 10: Mean AUC detection performance in % (over 10 seeds) for all individual classes and transfer learning-based methods on the CIFAR-10 one vs. rest benchmark with 80MTI OE from Section 5.3.

	Unsupervised			Supervised OE		
Class	DN2*	PANDA*	CLIP	PANDA*	ADIP*	BCE-CL
Airplane	93.9	×	99.4	×	99.2	**99.7**
Automobile	97.7	×	99.4	×	**99.8**	**99.8**
Bird	85.5	×	97.4	×	98.6	**99.2**
Cat	85.5	×	97.0	×	97.0	**98.8**
Deer	93.6	×	98.1	×	99.3	**99.6**
Dog	91.3	×	97.4	×	98.2	**99.2**
Frog	94.3	×	98.1	×	99.6	**99.9**
Horse	93.6	×	99.0	×	**99.8**	**99.8**
Ship	95.1	×	99.7	×	99.6	**99.8**
Truck	95.3	×	99.3	×	99.5	**99.9**
Mean AUC	92.5	96.2	98.5	98.9	99.1	**99.6**

Table 11: Mean AUC detection performance in % (over 10 seeds) with standard deviations for all individual classes for our transfer learning-based implementations on the CIFAR-10 one vs. rest benchmark with 80MTI OE from Section 5.3.

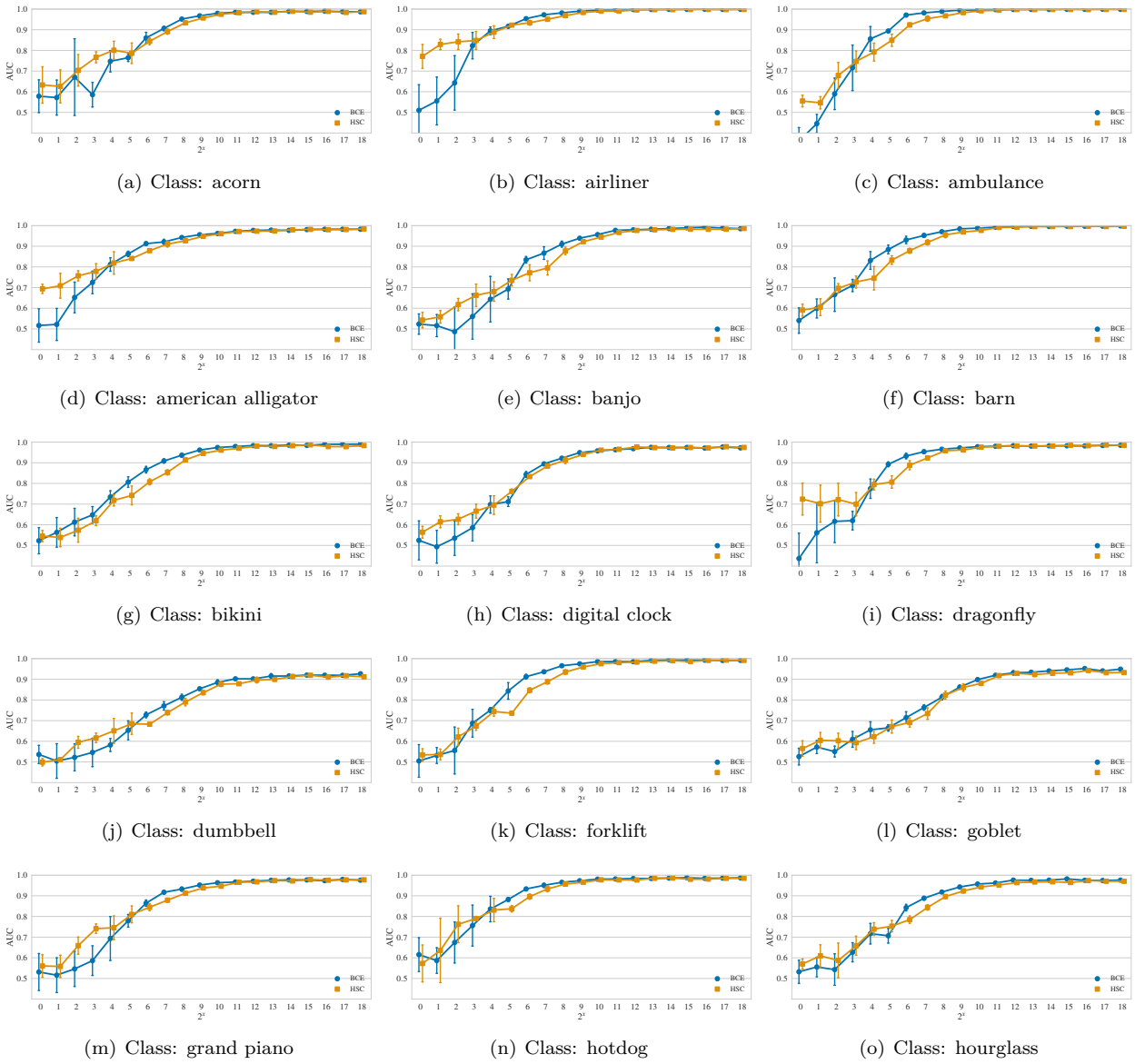
	Unsupervised	Supervised OE
Class	CLIP	BCE-CL
Airplane	99.40 ± 0.00	99.74 ± 0.02
Automobile	99.37 ± 0.00	99.82 ± 0.01
Bird	97.37 ± 0.00	99.29 ± 0.04
Cat	96.99 ± 0.00	98.86 ± 0.04
Deer	98.05 ± 0.00	99.62 ± 0.02
Dog	97.36 ± 0.00	99.26 ± 0.04
Frog	98.05 ± 0.00	99.89 ± 0.01
Horse	98.99 ± 0.00	99.84 ± 0.02
Ship	99.71 ± 0.00	99.86 ± 0.01
Truck	99.30 ± 0.00	99.90 ± 0.01
Mean AUC	98.46 ± 0.96	99.61 ± 0.34

Table 12: Mean AUC detection performance in % (over 10 seeds) for all individual classes for our transfer learning-based implementations on the ImageNet-1K one vs. rest benchmark with ImageNet-22K OE from Section 5.3.

Class	Unsupervised CLIP	Supervised OE BCE-CL
Acorn	99.78 ± 0.00	99.96 ± 0.01
Airliner	100.00 ± 0.00	100.00 ± 0.00
Ambulance	100.00 ± 0.00	100.00 ± 0.00
American alligator	99.98 ± 0.00	100.00 ± 0.00
Banjo	100.00 ± 0.00	100.00 ± 0.00
Barn	100.00 ± 0.00	100.00 ± 0.00
Bikini	99.87 ± 0.00	100.00 ± 0.00
Digital clock	99.69 ± 0.00	99.93 ± 0.02
Dragonfly	100.00 ± 0.00	100.00 ± 0.00
Dumbbell	99.91 ± 0.00	99.97 ± 0.02
Forklift	100.00 ± 0.00	100.00 ± 0.00
Goblet	99.29 ± 0.00	99.81 ± 0.04
Grand piano	100.00 ± 0.00	98.36 ± 4.87
Hotdog	99.99 ± 0.00	100.00 ± 0.00
Hourglass	99.69 ± 0.00	99.97 ± 0.02
Manhole cover	100.00 ± 0.00	100.00 ± 0.00
Mosque	100.00 ± 0.00	100.00 ± 0.00
Nail	99.61 ± 0.00	99.97 ± 0.01
Parking meter	99.52 ± 0.00	99.97 ± 0.01
Pillow	99.95 ± 0.00	100.00 ± 0.00
Revolver	100.00 ± 0.00	100.00 ± 0.00
Rotary dial telephone	99.84 ± 0.00	99.21 ± 1.75
Schooner	100.00 ± 0.00	100.00 ± 0.00
Snowmobile	99.99 ± 0.00	100.00 ± 0.00
Soccer ball	99.97 ± 0.00	99.98 ± 0.05
Stingray	100.00 ± 0.00	100.00 ± 0.00
Strawberry	99.81 ± 0.00	99.97 ± 0.07
Tank	100.00 ± 0.00	100.00 ± 0.00
Toaster	99.44 ± 0.00	99.88 ± 0.01
Volcano	100.00 ± 0.00	99.99 ± 0.04
Mean AUC	99.88 ± 0.19	99.90 ± 0.32

I.3 Varying the OE size

For the experiments on varying the number of OE samples (Section 5.2), we include plots for all individual classes in Figure 14 for CIFAR-10 and in Figures 12 and 13 for ImageNet-1K, respectively. Additionally, for the experiments on varying the diversity of OE data on CIFAR-10 with CIFAR-100 OE, we added the plots for all individual classes in Figure 15.



(a) Class: acorn (b) Class: airliner (c) Class: ambulance

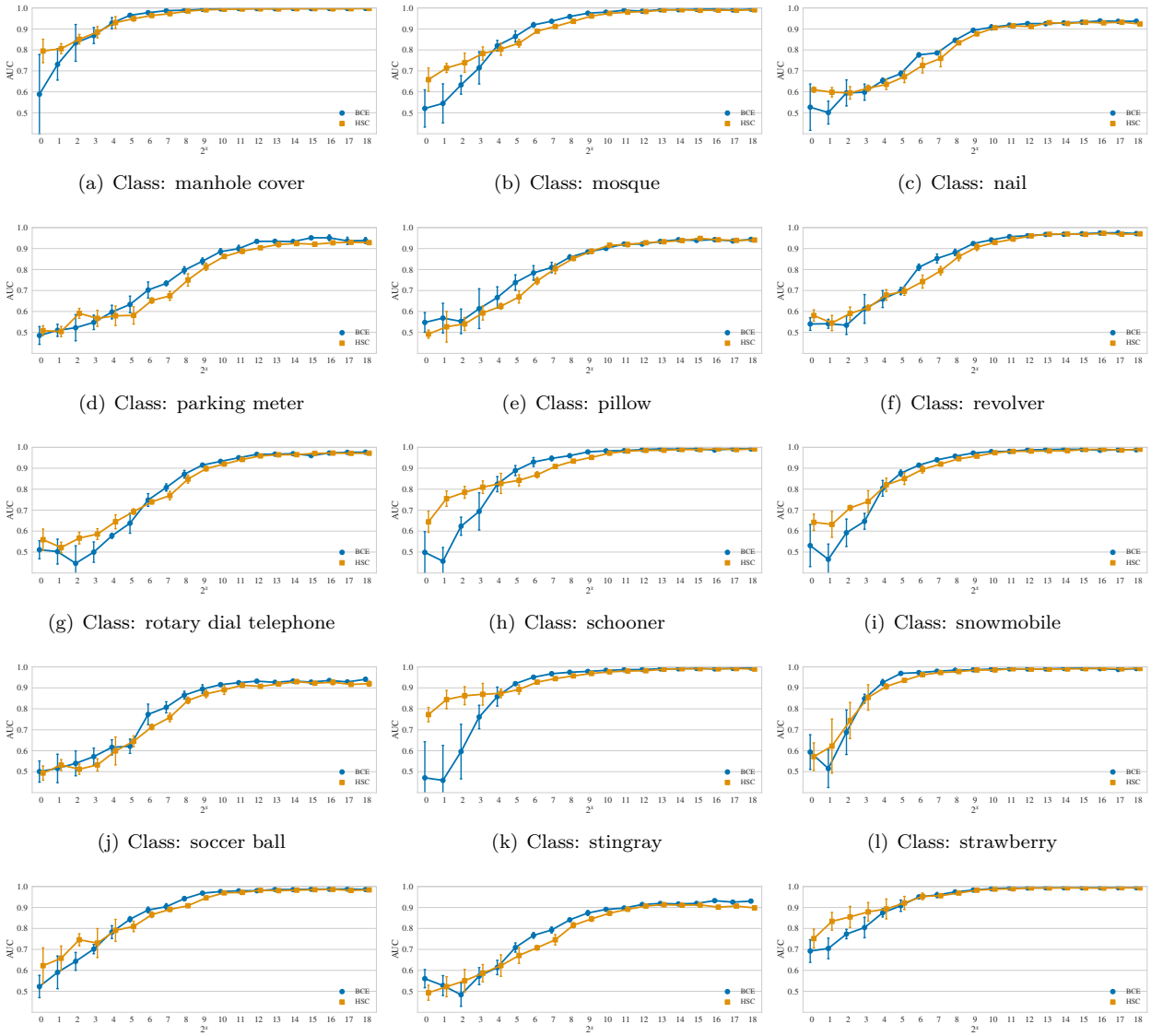
(d) Class: american alligator (e) Class: banjo (f) Class: barn

(g) Class: bikini (h) Class: digital clock (i) Class: dragonfly

(j) Class: dumbbell (k) Class: forklift (l) Class: goblet

(m) Class: grand piano (n) Class: hotdog (o) Class: hourglass

Figure 12: Mean AUC detection performance in % (over 5 seeds) for all classes of the ImageNet-1K one vs. rest benchmark from Section 5.2 when varying the number of ImageNet-22K OE samples. These plots correspond to Figure 2(b), but here we report the results for all individual classes (from class 1 (acorn) to class 15 (hourglass)).



(a) Class: manhole cover (b) Class: mosque (c) Class: nail

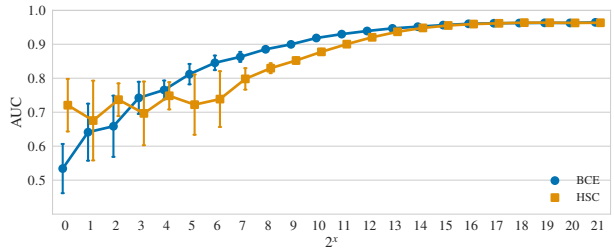
(d) Class: parking meter (e) Class: pillow (f) Class: revolver

(g) Class: rotary dial telephone (h) Class: schooner (i) Class: snowmobile

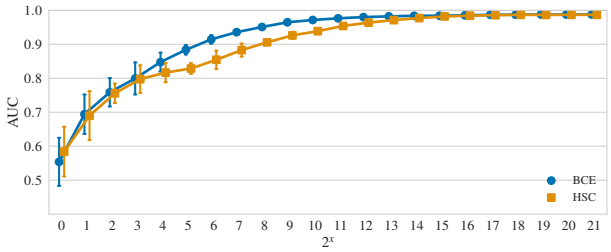
(j) Class: soccer ball (k) Class: stingray (l) Class: strawberry

(m) Class: tank (n) Class: toaster (o) Class: volcano

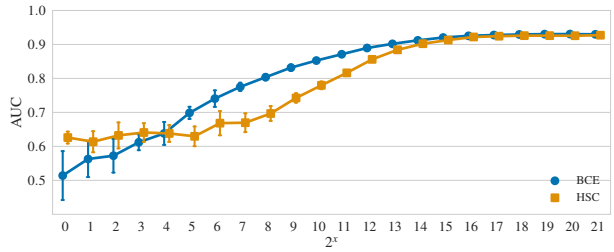
Figure 13: Mean AUC detection performance in % (over 5 seeds) for all classes of the ImageNet-1K one vs. rest benchmark from Section 5.2 when varying the number of ImageNet-22K OE samples. These plots correspond to Figure 2(b), but here we report the results for all individual classes (from class 16 (manhole cover) to class 30 (volcano)).



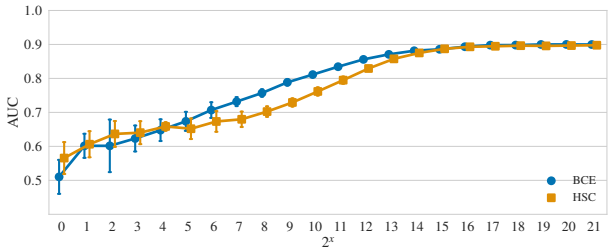
(a) Class: airplane



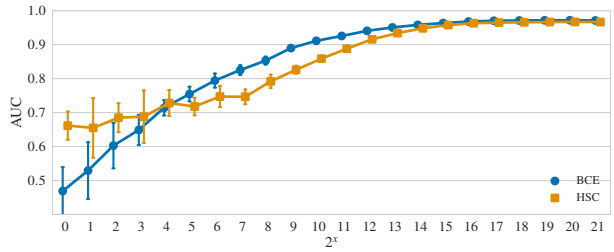
(b) Class: automobile



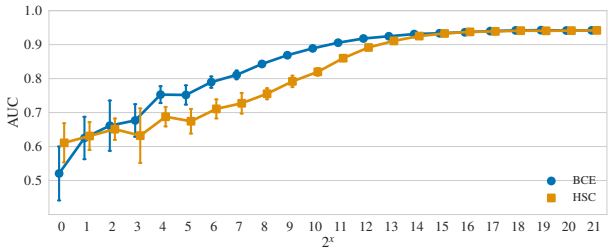
(c) Class: bird



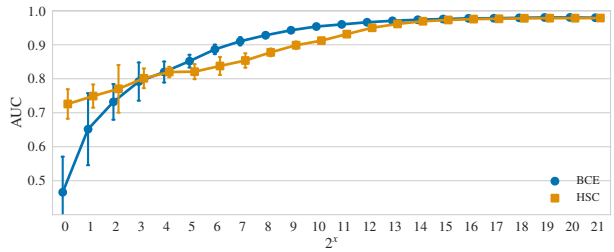
(d) Class: cat



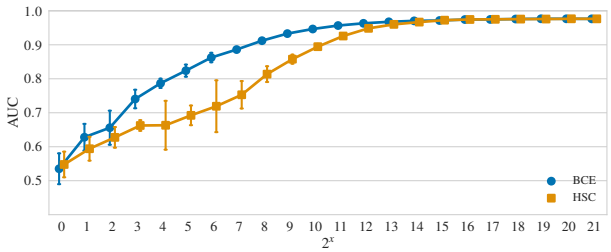
(e) Class: deer



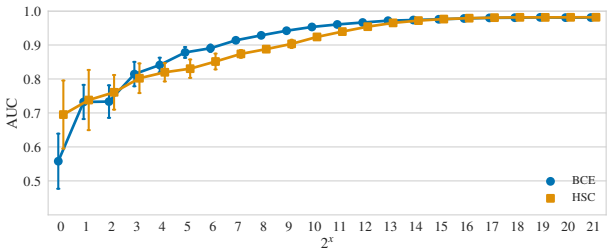
(f) Class: dog



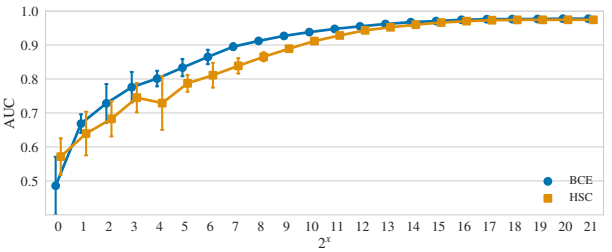
(g) Class: frog



(h) Class: horse

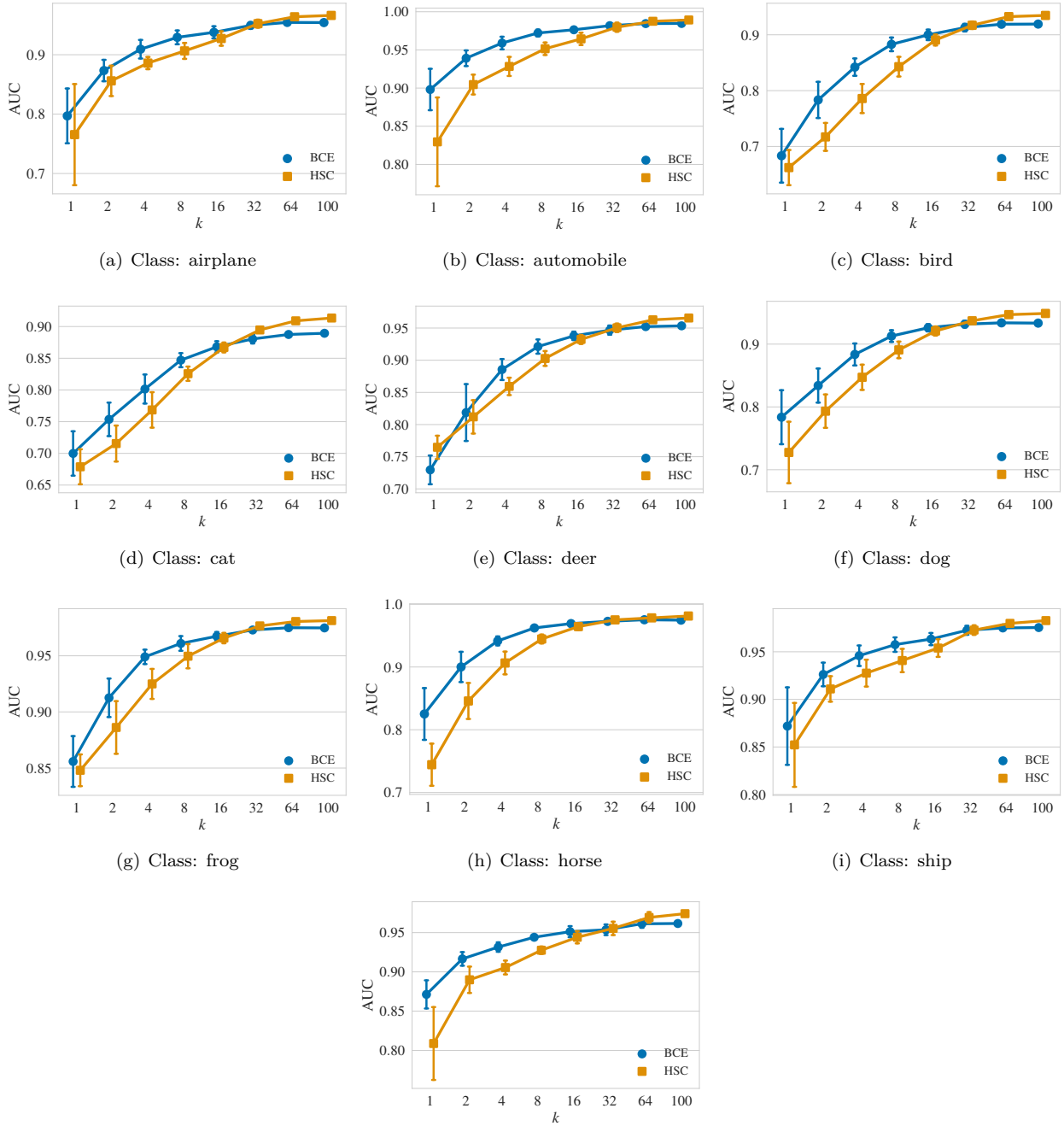


(i) Class: ship



(j) Class: truck

Figure 14: Mean AUC detection performance in % (over 10 seeds) for all classes of the CIFAR-10 one vs. rest benchmark from Section 5.2 when varying the number of 80MTI OE samples. These plots correspond to Figure 2(a), but here we report the results for all individual classes.



(a) Class: airplane (b) Class: automobile (c) Class: bird

(d) Class: cat (e) Class: deer (f) Class: dog

(g) Class: frog (h) Class: horse (i) Class: ship

(j) Class: truck

Figure 15: Mean AUC detection performance in % (over 10 seeds) for all CIFAR-10 classes from the experiment in Appendix A on varying the number of classes k of the CIFAR-100 OE data. These plots correspond to Figure 4, but here we report the results for all individual classes.

I.4 Robustness towards choice of OE samples

Here we provide class-wise results for the best and worst single OE samples found via an evolutionary algorithm (Section 5.4). On ImageNet, we only consider the first 10 classes of the ones used by (Hendrycks et al., 2019b): "Acorn", "Airliner", "Ambulance", "American alligator", "Banjo", "Barn", "Bikini", "Digital clock", "Dragonfly", "Dumbbell". However, the full 30-class one vs. rest benchmark was used to evaluate for each of these ten classes. Tables 13 and 14 show the performance of the best or worst single OE sample (from 80MTI) for each class in CIFAR-10, while the former shows results for unfiltered images and the latter for low-pass-filtered or high-pass-filtered images. Tables 15 and 16 show the same for ImageNet-1K with ImageNet-22K as OE. Figures 16, 17, 18, 19, 20, and 21 show the 4-5 best and worst samples for HSC and BCE, on CIFAR-10 or ImageNet-1K, for unfiltered or filtered images, respectively.

Table 13: Class-wise AUC detection performance in % for the best and worst single OE samples found via an evolutionary algorithm (Section 5.4) on the CIFAR-10 one vs. rest AD benchmark using 80MTI as OE.

	Best OE		Worst OE	
Class	HSC	BCE	HSC	BCE
Airplane	85.2	76.1	37.4	29.9
Automobile	78.1	71.5	36.4	32.9
Bird	74.0	67.9	47.2	35.8
Cat	72.5	64.9	38.7	35.1
Deer	79.9	68.2	55.1	29.8
Dog	71.6	70.6	41.0	35.0
Frog	84.2	71.7	47.7	24.8
Horse	66.2	66.2	41.9	38.7
Ship	86.5	72.8	46.2	27.1
Truck	78.8	69.1	41.2	26.9
Mean AUC	77.7	69.9	43.3	31.6

Table 14: Class-wise AUC detection performance in % for the best and worst single OE samples found via an evolutionary algorithm (Section 5.4) on the CIFAR-10 one vs. rest AD benchmark using 80MTI as OE. All images are either low-pass-filtered (LPF) or high-pass-filtered (HPF), both during training and testing.

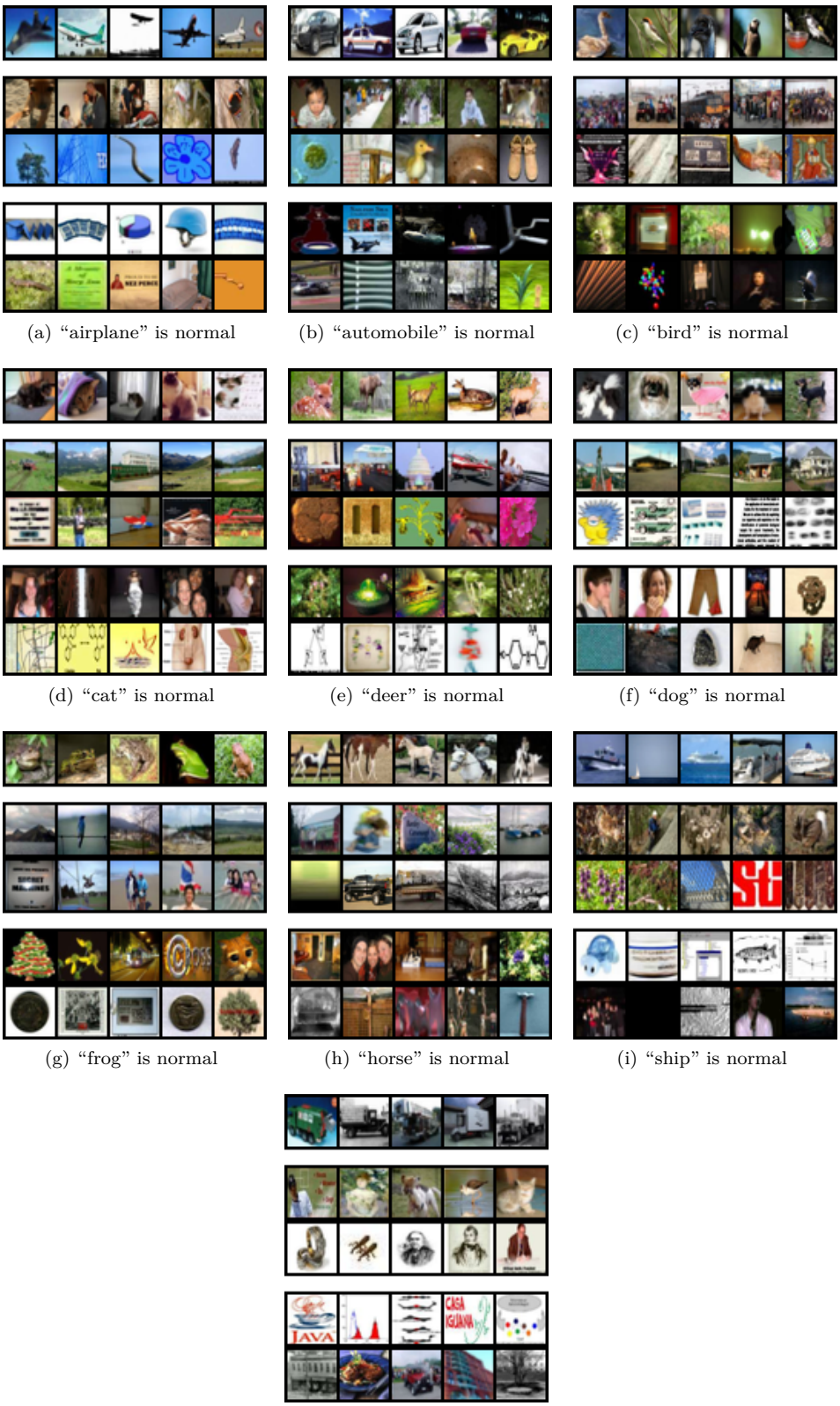
	Best OE				Worst OE			
Class	HSC LPF	BCE LPF	HSC HPF	BCE HPF	HSC LPF	BCE LPF	HSC HPF	BCE HPF
Airplane	83.0	68.5	68.5	64.6	36.0	22.1	40.7	32.5
Automobile	75.0	69.5	60.1	69.0	47.5	32.6	40.2	40.3
Bird	71.1	63.5	67.9	61.6	40.3	39.1	46.4	39.2
Cat	73.6	63.2	65.5	63.3	44.0	34.2	48.6	39.4
Deer	76.9	69.2	68.5	63.8	38.0	31.7	47.2	38.9
Dog	74.2	64.3	69.4	67.1	43.0	34.1	53.3	37.3
Frog	83.5	73.1	78.2	70.3	40.1	28.2	38.0	37.0
Horse	72.5	66.5	66.7	63.0	50.9	35.5	49.0	40.3
Ship	85.7	75.8	74.8	72.2	47.8	23.0	35.5	35.8
Truck	79.9	71.0	67.9	69.2	53.0	30.1	37.3	39.7
Mean AUC	77.5	68.5	68.8	66.4	44.1	31.1	43.6	38.0

Table 15: Class-wise AUC detection performance in % for the best and worst single OE samples found via an evolutionary algorithm (Section 5.4) on the ImageNet-1K one vs. rest AD benchmark using ImageNet-22k (with the 1K classes removed) as OE.

	Best OE		Worst OE	
Class	HSC	BCE	HSC	BCE
Acorn	84.0	76.6	43.6	22.3
Airliner	91.1	80.8	41.3	15.7
Ambulance	84.0	86.4	24.0	22.9
American alligator	81.2	75.7	54.6	30.1
Banjo	81.3	75.8	37.2	26.8
Barn	78.4	68.0	40.0	29.0
Bikini	66.7	67.0	41.5	38.9
Digital clock	67.7	73.2	32.5	28.1
Dragonfly	86.6	84.8	36.1	13.6
Dumbbell	71.8	66.5	40.8	35.9
Mean AUC	79.3	75.5	39.2	26.3

Table 16: Class-wise AUC detection performance in % for the best and worst single OE samples found via an evolutionary algorithm (Section 5.4) on the ImageNet-1K one vs. rest AD benchmark using ImageNet-22k (with the 1K classes removed) as OE. All images are either low-pass-filtered (LPF) or high-pass-filtered (HPF), both during training and testing.

	Best OE				Worst OE			
Class	HSC LPF	BCE LPF	HSC HPF	BCE HPF	HSC LPF	BCE LPF	HSC HPF	BCE HPF
Acorn	82.8	83.8	80.4	75.9	41.2	20.0	45.8	31.0
Airliner	78.2	77.3	83.6	86.3	44.6	21.7	52.5	18.0
Ambulance	86.7	76.3	79.1	83.3	53.6	26.1	44.4	29.2
American alligator	69.3	67.4	78.3	80.1	43.4	32.3	42.2	29.0
Banjo	83.0	74.8	71.3	75.9	48.0	21.5	39.1	31.9
Barn	79.6	71.5	73.0	77.6	45.2	25.6	44.4	26.2
Bikini	70.4	68.0	68.6	65.9	47.4	35.6	45.9	34.9
Digital clock	63.3	65.7	68.4	76.2	41.2	35.9	34.9	26.0
Dragonfly	85.0	85.6	83.7	84.6	33.0	12.9	51.4	20.2
Dumbbell	73.2	68.1	63.8	67.2	48.1	29.1	40.0	32.8
Mean AUC	77.2	73.8	75.0	77.3	44.6	26.1	44.1	27.9



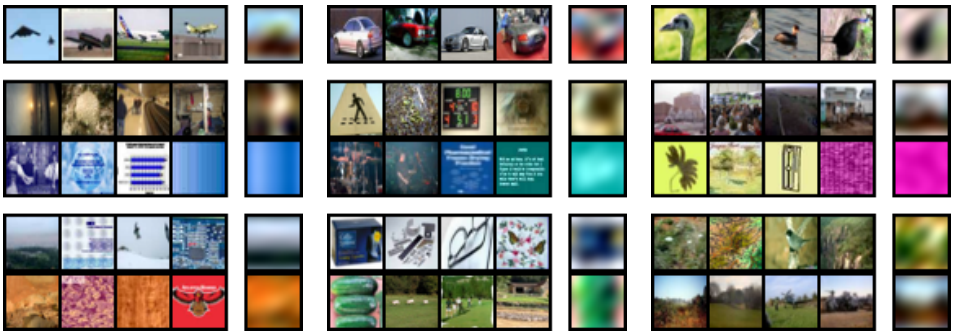
(a) “airplane” is normal (b) “automobile” is normal (c) “bird” is normal

(d) “cat” is normal (e) “deer” is normal (f) “dog” is normal

(g) “frog” is normal (h) “horse” is normal (i) “ship” is normal

(j) “truck” is normal

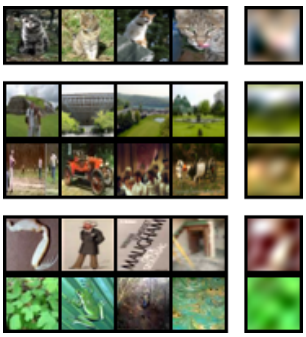
Figure 16: Optimal OE samples for CIFAR-10 with 80MTI as OE. The first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom), and the last two rows the worst samples found via HSC (top) and BCE (bottom).



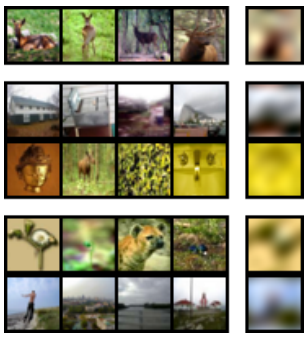
(a) “airplane” is normal

(b) “automobile” is normal

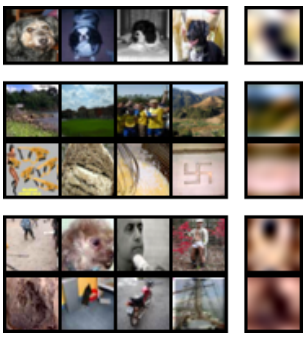
(c) “bird” is normal



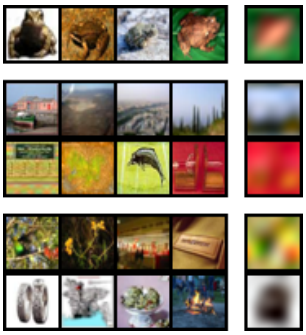
(d) “cat” is normal



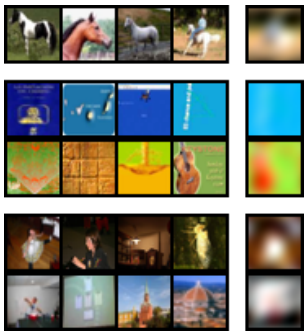
(e) “deer” is normal



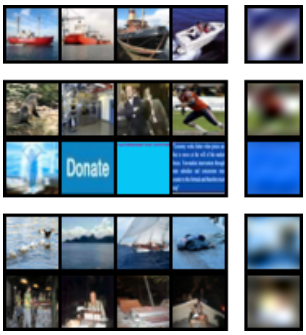
(f) “dog” is normal



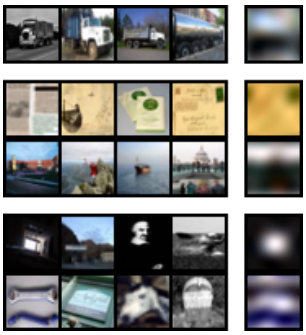
(g) “frog” is normal



(h) “horse” is normal



(i) “ship” is normal



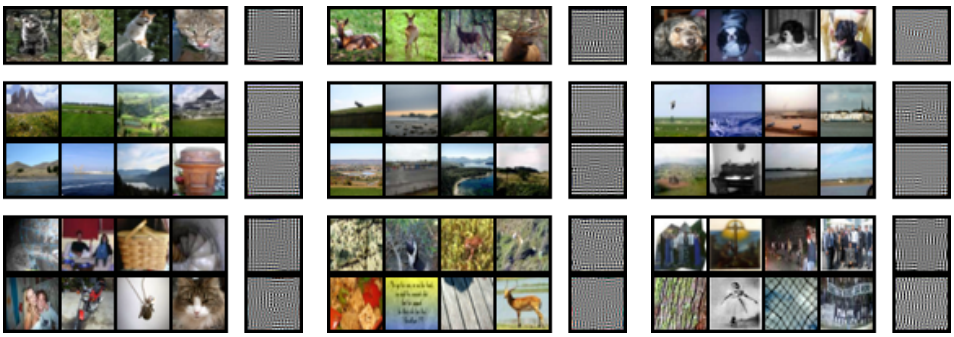
(j) “truck” is normal

Figure 17: Optimal OE samples for low-pass-filtered CIFAR-10 with low-pass-filtered 80MTI as OE. The first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom), and the last two rows the worst samples found via HSC (top) and BCE (bottom). The last column shows the low-pass-filtered version of the images, which is what the network sees during training and testing.

(a) “airplane” is normal

(b) “automobile” is normal

(c) “bird” is normal



(d) “cat” is normal

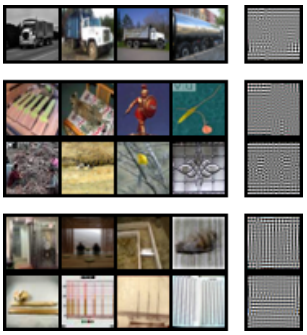
(e) “deer” is normal

(f) “dog” is normal

(g) “frog” is normal

(h) “horse” is normal

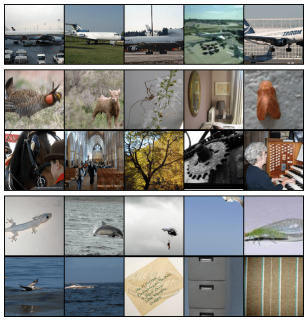
(i) “ship” is normal



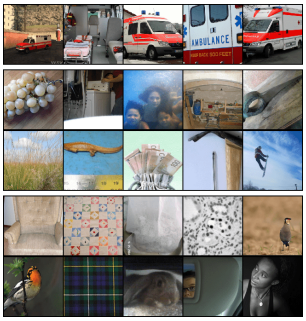
(j) “truck” is normal

Figure 18: Optimal OE samples for high-pass-filtered CIFAR-10 with high-pass-filtered 80MTI as OE. The first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom), and the last two rows the worst samples found via HSC (top) and BCE (bottom). The last column shows the high-pass-filtered version of the images, which is what the network sees during training and testing.

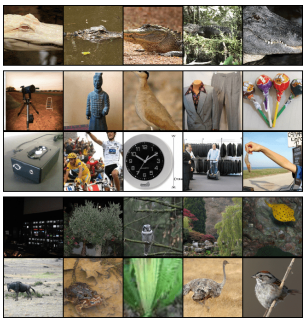
(a) “acorn” is normal



(b) “airliner” is normal



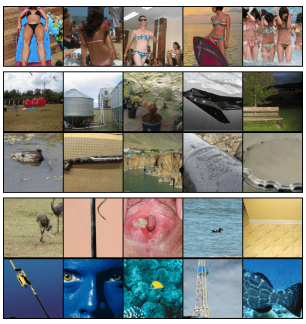
(c) “ambulance” is normal



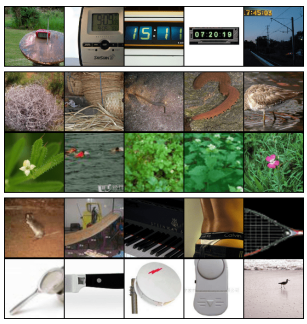
(d) “american alligator” is normal

(e) “banjo” is normal

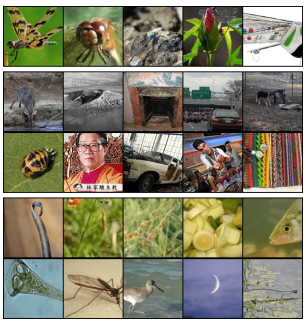
(f) “barn” is normal



(g) “bikini” is normal



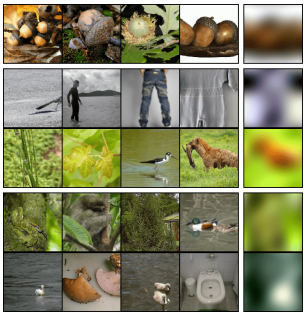
(h) “digital clock” is normal



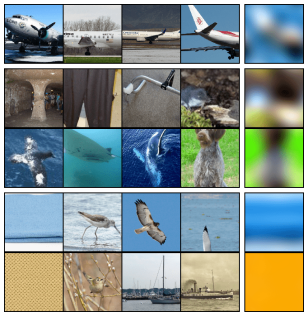
(i) “dragonfly” is normal

(j) “dumbbell” is normal

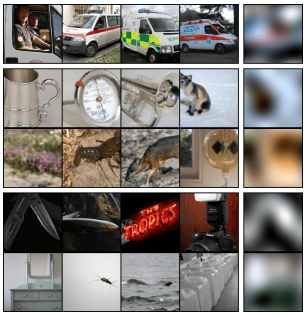
Figure 19: Optimal OE samples for ImageNet1k with ImageNet22k (with the 1K classes removed) as OE. The first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom), and the last two rows the worst samples found via HSC (top) and BCE (bottom).



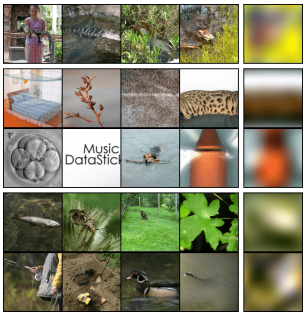
(a) “acorn” is normal



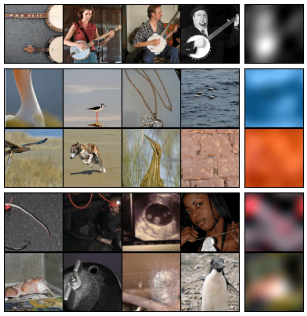
(b) “airliner” is normal



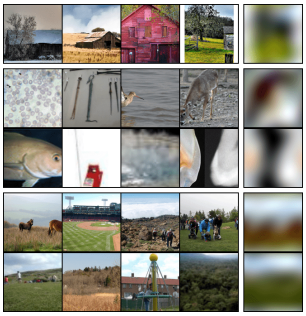
(c) “ambulance” is normal



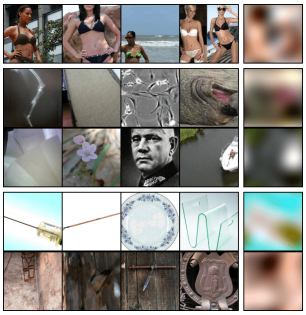
(d) “american alligator” is normal



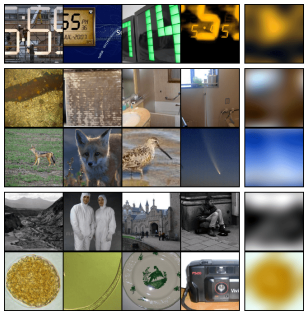
(e) “banjo” is normal



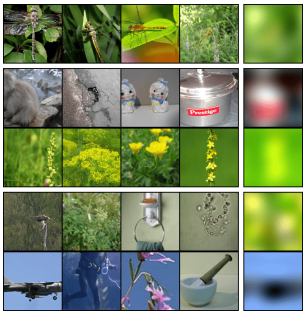
(f) “barn” is normal



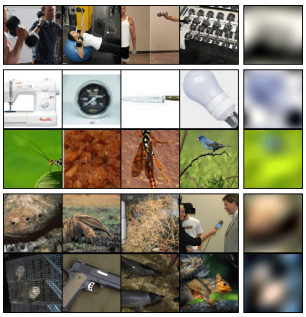
(g) “bikini” is normal



(h) “digital clock” is normal

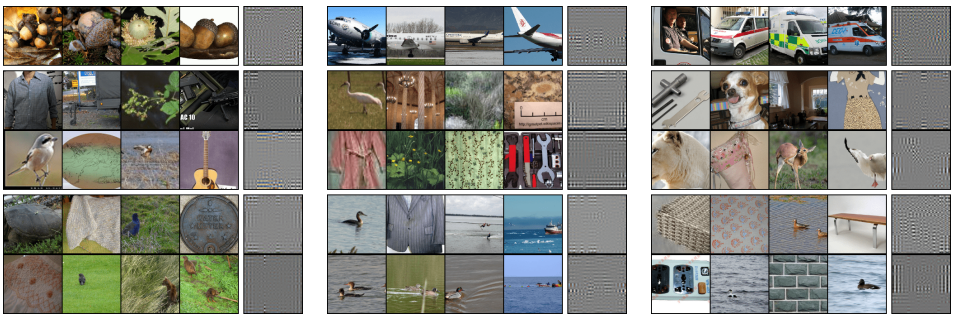


(i) “dragonfly” is normal



(j) “dumbbell” is normal

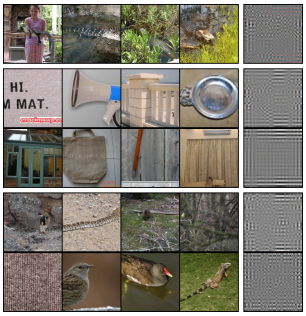
Figure 20: Optimal OE samples for low-pass-filtered ImageNet1k with low-pass-filtered ImageNet22k (with the 1K classes removed) as OE. The first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom), and the last two rows the worst samples found via HSC (top) and BCE (bottom). The last column shows the low-pass-filtered version of the images, which is what the network sees during training and testing.



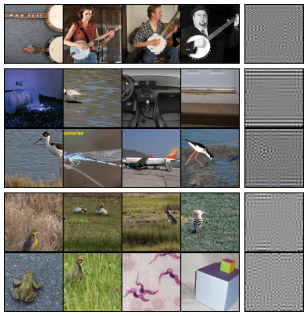
(a) “acorn” is normal

(b) “airliner” is normal

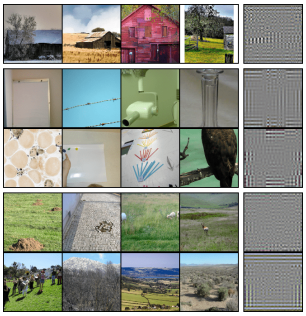
(c) “ambulance” is normal



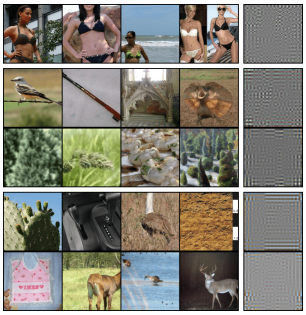
(d) “american alligator” is normal



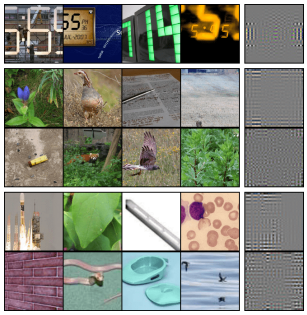
(e) “banjo” is normal



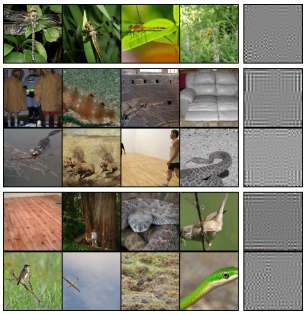
(f) “barn” is normal



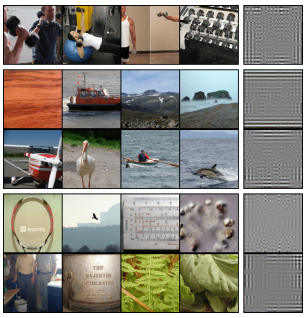
(g) “bikini” is normal



(h) “digital clock” is normal



(i) “dragonfly” is normal



(j) “dumbbell” is normal

Figure 21: Optimal OE samples for high-pass-filtered ImageNet1k with high-pass-filtered ImageNet22k (with the 1K classes removed) as OE. The first row shows normal samples, the next two rows the best samples found via HSC (top) and BCE (bottom), and the last two rows the worst samples found via HSC (top) and BCE (bottom). The last column shows the high-pass-filtered version of the images, which is what the network sees during training and testing.