Out-of-Distribution Detection with Semantic Mismatch under Masking

Yijun Yang, Ruiyuan Gao, and Qiang Xu

CUhk REliable Computing Laboratory (CURE Lab.)
Department of Computer Science and Engineering
The Chinese University of Hong Kong, Hong Kong S.A.R., China
{yjyang, rygao, qxu}@cse.cuhk.edu.hk

Abstract. This paper proposes a novel out-of-distribution (OOD) detection framework named MOODCAT for image classifiers. MOODCAT masks a random portion of the input image and uses a generative model to synthesize the masked image to a new image conditioned on the classification result. It then calculates the semantic difference between the original image and the synthesized one for OOD detection. Compared to existing solutions, MOODCAT naturally learns the semantic information of the in-distribution data with the proposed mask and conditional synthesis strategy, which is critical to identify OODs. Experimental results demonstrate that MOODCAT outperforms state-of-the-art OOD detection solutions by a large margin. Our code is available at https://github.com/cure-lab/MOODCat.

Keywords: OOD detection, Robust AI, Generative model

1 Introduction

Deep neural networks (DNNs) are trained under a "close-world" assumption [13, 24], where all the samples fed to the model are assumed to follow a narrow semantic distribution. However, when deployed in the wild, the model is exposed to an "open-world" with all kinds of inputs not necessarily following this distribution [9]. Such out-of-distribution (OOD) samples with significantly different semantics may mislead DNN models and generate wrong prediction results with extremely high confidence, thereby hindering DNN's deployment safety [1, 7, 15, 16, 34].

To distinguish OOD samples from the in-distribution (In-D) data, some propose to reuse the features extracted from the original DNN model to tell the difference [16, 27–29, 43, 44]. However, such a feature-sharing strategy inevitably results in the trade-off between the prediction accuracy for In-D samples and the OOD detection capabilities. There are also various density-based OOD detection methods [3, 36, 37], which try to model the In-D data with probabilistic measures such as energy and likelihood. However, the trustworthiness of these measures is not guaranteed [22]. Another popular OOD detection mechanism uses generative models (e.g., variational autoencoder (VAE)) to reconstruct the input [6, 39]. Based on the assumption that In-D data can be well reconstructed while OODs

cannot since they are not seen during training, one could measure the distance between the original input and the reconstructed one and detect OOD with a threshold. However, this assumption is not sound. There are cases where OODs are faithfully reconstructed with the generative models, causing misjudgements [22].

In this paper, we propose a novel distance-based OOD detection framework, named *Masked OOD Catcher* (MOODCAT), wherein we consider the semantic mismatch under masking as the distance metric. Specifically, for image classifiers, we first randomly mask a portion of the input image, use a generative model to synthesize the masked image to a new image conditioned on the classification result, and then calculate the semantic difference between the original image and the synthesized one for OOD detection.

Our insight is that, the classification result carries discriminative semantic information and it imposes strong constraints onto the synthesis procedure, especially when trying to recover the masked portions. With MOODCAT, for correctly classified In-D data, the generative model can use the unmasked region to make up the masked part with sufficient training. In contrast, for OOD samples that are semantically different, the synthesized image based on the classification result tends to be dramatically different, especially for the masked region.

MOODCAT is a standalone OOD detector, and it does not require fine-tuning the original classifier. Consequently, it can be combined with any classifier to equip it with OOD detection capability without affecting its accuracy. We perform comprehensive evaluations on standard OOD detection benchmarks [45] with six datasets and four detection settings. Results show that our method can outperform state-of-the-art (SOTA) solutions by a large margin. We summarize the contributions of this work in the following:

- We propose a novel OOD detection framework by identifying semantic mismatch under masking, MOODCAT. To the best of our knowledge, this is the first work that *explicitly* considers semantics information for OOD detection.
- We present a novel masking and conditional synthesis flow in MOODCAT, and investigate various masking strategies and conditional generator designs for OOD detection.
- To tell the semantic difference between the original image and the synthesized one, we employ an anomalous scoring model composed of various quality assessment metrics (e.g., DISTS [8] and LPIPS [53]) and a newly-proposed conditional binary classifier.

The rest of the paper is constructed as follows. Section 2 surveys related OOD detection methods. We detail our proposed MOODCAT framework in Section 3. Section 4 presents our experimental results and the corresponding ablation studies. Finally, Section 5 concludes this paper.

2   Related Work

2.1   Existing OOD Detection Methods

In general, OOD detection methods can be categorized into: classification-based, density-based and distance-based methods [46].
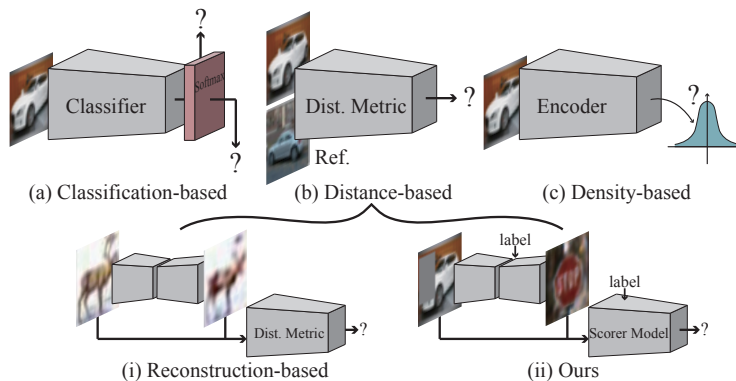
Fig. 1: Comparison of OOD Detection Methods. MooDCat is a distance-based solution, and it relies on conditional image synthesis rather than reconstruction.

Classification-based methods derive OOD scores based on the output of DNNs, as shown in Fig. 1(a). Maximum Softmax Probability (MSP) [16] simply uses the maximum softmax probability as the indicator of In-D data. ODIN [27] is applies a temperature scaling to the softmax value for OOD detection. Follow-up works include methods based on the output of DNNs [28, 29, 43, 44], the gradient of DNNs [20] and data generation or augmentation [41, 42]. Although simple to implement, most of them alter the training process of the original classifier, thereby reducing the classification accuracy for In-D samples.

Density-based methods usually apply some probabilistic models for the distribution of In-D samples and regard test data in low-density regions as OOD [3, 26, 37], as shown in Fig. 1(c). Some methods in this category also resort to generative models [36] to learn the distribution of data. However, recent research found that the learned density model may assign high likelihood value to some OODs, since the obtained likelihood could be dominated by low-level features such as location and variance instead of the high-level semantics, which is related to the specific network architecture and data used for learning [3, 31].

Distance-based methods consider that OODs should be relatively far away from In-Ds. They either calculate the centroids of In-D classes in the feature space [19, 50] (Fig. 1(b)) or reconstruct the input itself (Section. 2.2, Fig. 1(i)) for OOD detection. However, for high-level semantic features, their assumption for distance disparity may not hold, and high reconstruction quality cannot ensure In-Ds. In this paper, we use conditional synthesis on masked images to highlight the semantics difference in the image space.

2.2 Reconstruction-based OOD Detection

Reconstruction-based methods, which fall into the category of distance-based methods, are closely related to the proposed MooDCat technique. These methods are based on the assumption that In-D data can be well-reconstructed from a

trained generative model, but OOD cannot as they are not seen during training (see Fig. 1(i)). Previous reconstruction-based detectors generally distinguish OOD samples by comparing pixel-level quality "degradation" of the reconstruction for given input [6, 39]. However, without prior-knowledge about OOD samples, there is no guarantee for such quality degradation. In contrast, MOODCAT tries to synthesize In-D images instead of reconstructing the inputs, which is in line with the objective of the generative model.

The framework of MOODCAT (Fig. 1(ii)) is inspired by [47], which detects adversarial examples (AE) by generating synthesized images conditioned on the output of the misled classifier. AE detection is quite different from OOD detection because adversarial examples are In-D samples with imperceptible perturbations. The classification label itself is sufficient to train the generative model to differentiate AEs and benign samples. This is not the case for OOD samples, which motivates the proposed MOODCAT solution for OOD detection, as detailed in Section 3.

2.3   OOD Detection with External OOD Data

Recently, some researchers propose to involve data from other datasets to simulate OOD samples for model training. Representative "OOD-aware" techniques include Outlier Exposure (OE) [17], Maximum Classifier Discrepancy (MCD) [49], and Unsupervised Dual Grouping (UDG) [45]. OE relies on large-scale purified OOD samples, whereas MCD and UDG only need extra unlabeled data, which contains both In-D and OOD data. However, all of them are classification-based methods, where external data are used to train a modified classifier model. Following the same unlabeled extra data setting of MCD and UDG, we present that including extra training data (In-D, OOD mixture) into the training process can further improve MOODCAT's performance. Since our MOODCAT works independently with the original classifiers, MOODCAT will not degrade the accuracy of the original classifiers. We provide the detailed description in Section 3.6 and experimental results in Section 4.

2.4   Open Set Recognition

A similar problem to OOD detection is the so-called Open Set Recognition (OSR) problem, which aims to distinguish the known and unknown classes [35, 46]. Several existing works [10, 11, 32, 35] targeted on OSR also employ generative models, whereas differ from MOODCAT significantly. Specifically, OSRCI [32] uses Generative Adversarial Network (GAN) as data augmentation to train the classifier; C2AE [35] and CVAECapOSR [11] are conditional VAE/Autoencoder-based detectors. C2AE identifies outliers based on reconstruction errors, and requires K-time inference to give the final decision. In contrast, MOODCAT infers once and makes the decision based on semantic contradiction. CVAECapOSR use Conditional VAE (CVAE) to model the distribution of In-D samples and detect outliers at latent space (i.e., without using the generator), whereas MOODCAT detects at image space.
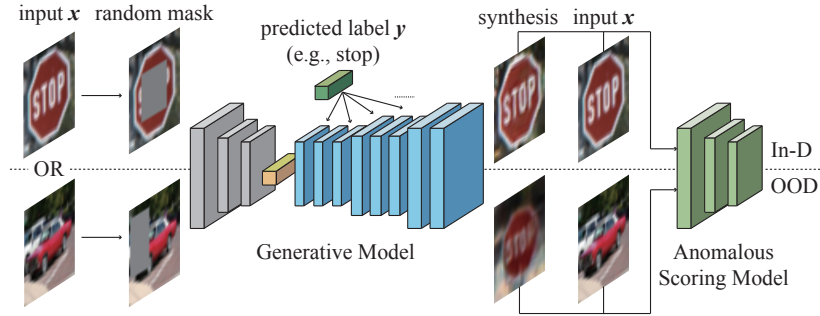
Fig. 2: Pipeline of MoodCat. We first mask a portion of the input image. Next, a generative model synthesizes the masked image to a new image conditioned on y, and then an anomalous scoring model measures the semantic difference between the input image and the synthesized one for OOD detection.

3 Proposed Method

3.1 Design Goals

This work considers the scenario where we have an In-D data trained classifier ($\mathcal{C}$), which needs to be deployed in the wild. Consequently, the classifier will be threatened by OOD samples. We aim at building an OOD detection method, which can identify the OOD samples effectively without compromising the classification accuracy of the classifier. Our model is assumed to have access to the predicted label, y, but we do not modify any part of the classifier, including but may not limit to the architecture and the trained weights. As a result, our method can be a plug-and-play detector that easily cooperates with classifiers.

3.2 Method Overview

As pointed out by [46], OOD samples ($x_o \in \mathcal{O}$) are defined by label-shifted samples or samples with non-overlapping labels w.r.t the training data, or In-Ds ($x_{in} \in \mathcal{I}$). Hence, the semantics of any OOD sample contradicts with any In-D sample. This is the observation that motivates us to design a framework for OOD detection by spotlighting their semantic discrepancy.

Fig. 2 depicts the overview of our method. The proposed Masked Out-of-Distribution Catcher (MoodCat) contains three stages: *randomly masking*, *generative synthesis* and *scoring*. Specifically, we first randomly mask the input image x as $x_m = \mathbf{M}(x)$, where $\mathbf{M}(\cdot)$ indicates the randomly masking operation. Then, we apply a generative model, $\mathbf{G}$, to synthesize a new image, x', by taking x_m as the template and conditioning on the label y. Finally, we apply an anomalous scoring model to judge the discrepancy between the input and its synthesis.

Through masking, x_m will partially lose its original semantic meaning, and thus leave more space for $\mathbf{G}$ to synthesize new content. With y as the condition,

the newly synthesized content should be consistent with the semantic meaning indicated by y. Note that, we use the ground truth label y to train the generative model and use the output from the classifier when inference, i.e., $y = \mathcal{C}(x)$.

Here, we analyze different situations with In-D or OOD samples. On the one hand, if an In-D sample x_{in} comes, the predicted label y matches x_{in}'s intrinsic semantic meaning appropriately. Although the input image, $x_{in \cdot m}$, is partially masked, there should be some visual clue related to its semantic meaning, e.g., wings of a bird or paws of a dog. As a result, $\mathbf{G}$ can synthesize x'_{in} quite faithful to x_{in}. As exemplified in the upper half of Fig. 2, the synthesis of the "stop" sign can be very close to the original input. On the other hand, when it comes to an OOD sample x_o, the predicted label provided by the classifier is irreverent to x_o's semantic meaning. Even with x_o as a template, the generative model will try its best to synthesize contents related to the semantic label. As a result, the mismatch of semantic meaning between input and label can be spotlighted by the discrepancy between input and its synthesis. As the example shows in the lower part in Fig. 2, if an OOD sample (*car*) is wrongly predicted as a "stop" sign. The synthesis will be highly related to the "stop" sign rather than the original image.

Through such conditional synthesis, we can spotlight the discrepancy caused by OOD samples. Thus OODs can be easily distinguished by comparing the pair of input with its synthesis, (x, x').

3.3   Masking Mechanism

In MOODCAT, the generative model uses the input image as a template and synthesizes an image with the same semantic meaning as the given label. A high-quality synthesis can better highlight the contradiction. However, due to the intrinsic contradiction between the input image and the label for OOD, too much information from the input image can degrade the quality of generation. Therefore, we propose to apply masking on x to remove some redundant information while leaving more space for the generative model to synthesize.

The use of masking follows the key motivation of MOODCAT in OOD detection, which applies generation for synthesis rather than reconstruction. Previous reconstruction-based methods (e.g., [6,39]) tend to reconstruct the image based on pixel-level dependency. In practice, the assumption that an OOD sample cannot be reconstructed well may not hold since they do not consider any semantics. However, our generative model aims at semantic synthesis. The masking mechanism can cooperate with the predicted label from the classifier to spotlight the contradiction caused by OOD.

The contribution of randomly masking is twofold: **1)** masking the input image can encourage the generative model to better depict the semantic meaning of the given label on the synthesis, especially to an OOD sample; **2)** masking, as a typical data augmentation method, can encourage the encoder to summarize the features of the input from a holistic perspective, thus improve the quality of synthesis, especially when synthesizing with In-Ds as templates. Obviously, the above two aspects both contribute to apart the behavior of In-D and OOD. As a result, a large discrepancy lies in the OOD sample and its synthesis.

3.4   Generative Model

The Generative model is responsible for generating a synthesis by taking both the masked input x_m and the pre-assigned semantic label y into consideration. As shown in Fig. 2, we select the Encoder ($\mathbf{E}$) and Decoder ($\mathbf{D}$) architecture as the generative model, i.e. $\mathbf{G} = \mathbf{E} \cdot \mathbf{D}$. This architecture is inspired by [47]. The encoder $\mathbf{E}$ acts as a feature extractor (as shown by the gray part in Fig. 2). By taking the masked image x_m as input, $\mathbf{E}$ is expected to capture necessary low-level features and encoder them as a latent vector, $z = \mathbf{E}(x_m)$. As done by VAE [21], we use the KL Divergence to regulate the latent vector z, which can be formulated as Eq. (1).

$$\mathcal{L}_{KLD} = D_{KL}[\mathcal{N}(\mu(x_m), \Sigma(x_m)) \| \mathcal{N}(0, 1)], \tag{1}$$

where $\mathcal{N}(\mu, \Sigma)$ indicates the Gaussian distribution with respect to μ and Σ. We use the reparameterization trick from VAE on the latent variable z during training, $z = \mu(x_m) + \Sigma(x_m) \cdot \epsilon$, where $\epsilon \sim \mathcal{N}(0, 1)$.

The decoder, $\mathbf{D}$, is trained to generate a synthesis $x' = \mathbf{D}(z, y)$. The given semantic label y is used to control the semantic meaning of the synthesis, while z is used to provide low-level features from the template image x. This synthetic target is fulfilled through the class-conditional batch normalization layer [5]. This layer is usually used in conditional image generation [30, 52]. Since the normalization is determined by the given semantic label y, the semantic meaning of the synthesis can be highly dependent on it. As a result, if the semantic meaning of x is consistent with y (in the case of In-D samples), the synthesis can be highly close to x. However, if input an OOD sample, the semantic contradiction between the input image and the label will lead the synthesis to be far away from the input image, thus spotlighting the contradiction.

We implement $\mathbf{D}$ based on the generator architecture proposed in [2]. We apply the classic ℓ_1, ℓ_2 and $\mathcal{SSIM}$ [38] as part of loss items to constrain that x' resembles x. Furthermore, we adopt the U-net based discriminator [40] to operate an adversarial loss on the training process to further improve the quality of the synthetic image. Compared with the vanilla discriminator, this U-net based one can additionally provide a per-pixel real/fake map to locate the fake parts in the image. Therefore, the generative model can be trained to focus on both local and global features with more realistic details. Due to space limitations, we detailed the training process and corresponding objective functions of $\mathbf{G}$ in Appendix.

3.5   Anomalous Scoring Model

As analyzed in the former sections, MoodCat can generate high-quality syntheses in terms of similarity for In-D samples, but not for OOD samples. To distinguish OODs from In-Ds, we develop an anomalous scoring model. The proposed anomalous scoring model is built on two types of scorers: one is the *conditional binary classifier*, and the other is *Image Quality Assessment models* (IQA) [53]. Both can provide assessments of the syntheses by referring to input images.

Conditional Binary Classifier. Identifying the semantic mismatch lies between OOD and its synthesis can be seen as a binary classification task. With the trained generative model, we can train a binary classifier for this in a supervised way. This is feasible because the binary classifier can learn to identify OODs by the similarity between the given image and its conditional synthesis. We also provide the semantic label to the classifier for judgement, this can further ease the distinguishing procedure.

Note that, we do not rely on OOD data through training. To mitigate the lack of OOD samples during training, we leverage the In-D samples with the synthetic results under mismatched labels to simulate the behavior of OOD samples. As a result, the binary classifier can learn to identify the semantic mismatch, which is the spotlighted feature for OOD samples. Moreover, we condition the binary classifier on the semantic label via the projection layer [30]. With the prior knowledge of the class, the binary classifier can learn a fine-grained decision boundary for each class, leading to better performance.

Another good property of this training strategy is that there is no need for specific information from the DNN model, here the classifier ($\mathcal{C}$), to be protected. Actually, the judgement of the binary classifier is only conditioned on given label, which is independent of the DNN model. Therefore, the trained conditional binary classifier can fit various DNN models in a plug-and-play manner.

Fig. 3 demonstrates the training process of the proposed conditional binary classifier, $\mathcal{C}_b$. *hinge loss* serves as the objective, as formulated in Eq. (2), where the x'_y indicates the synthesis generated under the groundtruth label y of x, and the (x, x'_y) is used as the positive pair. $x'_{y'}$ depicts the synthesis generated under a randomly sampled mismatched semantic label $y' \neq y$, then $(x, x'_{y'})$ is used as the negative pair during training. During inference, $\mathcal{C}_b$ is used to score the input image pair. The score can be used to flag OOD samples by a given threshold.
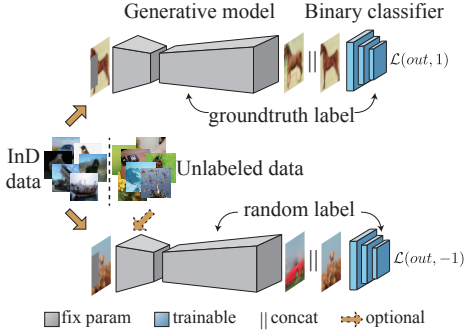

Fig. 3: Training pipeline of the proposed Conditional Binary Classifier.

$$\mathcal{L}_{\mathcal{C}_b} = ReLU(1 - \mathcal{C}_b((x, x'_y), y)) + ReLU(1 + \mathcal{C}_b((x, x'_{y'}), y')) \qquad (2)$$

Image Quality Assessment Models. IQA models [53] are widely adopted to evaluate the perceptual quality of a synthesis by referring to the source image in many computer vision tasks, such as denoising, super-resolution and compression. Here, we apply IQA models as the perceptual metric for the quality of synthesis, forming part of our anomalous scoring model. Since our generative model has already highlighted the contradiction caused by OOD through conditional synthesis, IQA models can be directly applied for detection.

In all, the *Conditional Binary Classifier* and *IQA* models work in a cascade way, where any scorer flags an OOD can lead to the final rejection. Different scoring mechanisms can evaluate the quality of generation from different perspectives, thus supporting each other for better performance than anyone alone.

3.6   Learning with Unlabeled Data

Recently, researchers have shown that including OOD data into the training process can improve the performance of OOD detection [17, 49]. However, they may rely on well labeled data that need to manually identify In-D and OOD elements [1], e.g., [17]. On the contrary, unlabeled data can be easily collected from various sources with little cost. Actually, MOODCAT can be further improved with unlabeled data.

The Conditional Binary Classifier introduced in Section 3.5 is trained by treating In-D data with synthesis of mismatched semantic labels as negative pairs. This procedure can be directly combined with external data. Note that, our negative pair only require that the semantic label used for synthesis is different from that of input image. To eliminate possible In-D samples from unlabeled data, we can apply a classifier trained with In-D data to generate pseudo label for unlabeled data. Then, during training, when we randomly sample the mismatched semantic label, we uniformly sample from all possible labels other than the pseudo label. In this way, the sampled semantic label is ensured to be mismatched for both OOD and some possible In-D data from the unlabeled data. Therefore, all of them can be utilized correctly to improve the performance of MOODCAT.

4   Experiments

4.1   Evaluation Settings

Benchmarks. We evaluate MOODCAT on the most recent semantic OOD detection benchmarks, SC-OOD benchmarks [45]. SC-OOD benchmarks provide extensive semantic-level OOD detection settings for evaluation. Specifically, from SC-OOD, images from different datasets are filtered to ensure that only those containing different semantic meanings are considered as OOD samples. SC-OOD focus on the semantic difference between samples, thus being more practical for the real-world model deployment than other previous OOD benchmarks [16,17,27], which are built by setting one dataset as In-D and all others as OOD.
Datasets. Following the settings in [45], we employ CIFAR-10 [23], CIFAR-100 [23] as In-D samples, respectively, and others as OOD samples. When setting CIFAR-10 as In-D, we employ six datasets as OOD datasets, including SVHN [33], CIFAR-100 [23], TEXTURE [4], PLACES365 [54], LSUN [48] and TINY-IMAGENET [25]. For CIFAR-100 benchmarks, the OOD datasets are the same as that of CIFAR-10, except for swapping CIFAR-100 for CIFAR-10 as OOD.

---

[1] Note that, images from another dataset are not necessarily to be OOD w.r.t semantic meaning [45].

4.2   Evaluation Metrics

We employ FPR@TPR95%, AUROC, AUPR, and Classification Accuracy as the evaluation metrics following [29,45]. In this paper, unless otherwise specified, we denote the In-D as Positive (P), and the OOD as Negative (N).

FPR@TPR95% presents the False Positive Rate (FPR) when the True Positive Rate (TPR) equals 95%. This metric reflects the ratio of falsely identified OOD when most of In-D samples are correctly recognized.

AUROC. The Area Under Receiver Operating Characteristic curve (AUROC) is an overall evaluation metric to reflect the detection capability of a detector.

AUPR-In AUPR calculates the Area Under the Precision-Recall curve. AUPR is a complementary metric that reflects the impact of imbalanced datasets. For AUPR-In metric, In-D samples are denoted as positive samples.

AUPR-Out indicates the same measure as AUPR-In mentioned above, whereas the OOD samples are deemed as positive during calculating AUPR-Out.

Classification Accuracy presents the classifier's performance on the In-D samples. It indicates the impact on the original classifier caused by OOD detector.

4.3   Experimental Results

We evaluate MOODCAT with two settings: **1)** MOODCAT trained with In-D dataset only; **2)** MOODCAT trained with external unlabeled data (Section 3.6). We report the results in Table 7 and Table 8, respectively. Experiments are performed with ResNet18 [14] classifier[2] for fair comparison.

Main Results. As shown in Table 7 and Table 8, experimental results indicate that MOODCAT outperforms or at least on par with SOTA methods on CIFAR-10/CIFAR-100 benchmarks without/with external training data. Since our method detects OODs relying on their semantic-level mismatching instead of low-level distribution shift, the performance of MOODCAT is stable across various OODs. As a plug-and-play model, MOODCAT causes no classifier performance degradation.

CIFAR-10 *Benchmark.* Table 7 reports the detection performance of ODIN [27], EBO [29] and MOODCAT on CIFAR-10 benchmarks. ODIN and EBO are developed without external OOD data. When comparing these two methods, we implemented MOODCAT in the same setting. We additionally report results with external data for training to show the improvements. As shown in Table 7, MOODCAT outperforms ODIN and EBO in most cases. For example, for the AUROC, which reflects the overall performance of a detector, MOODCAT outperforms baselines on all six OOD benchmarks. Furthermore, MOODCAT presents a more stable performance across all OOD datasets. As for statistics, we report the standard deviation (Std) for each metric [3]. As in Table 7, the Std of MOODCAT can be much lower than baselines. To be more specific, EBO performs quite well when encounter OODs sourcing from SVHN (AUROC=92.08%), but when it comes to

---

[2] For more results under other classifier architectures (WRN28 [51], DenseNet [18]), please refer to Appendix.

[3] FPR@TPR95% is only a single point on the PR curve. It may not reflect the overall performance in terms of standard deviation.

Table 1: OOD Detection Performance on Cifar-10 as In-D without using external OOD data for training. All the values are in percentages. ↑/↓ indicates higher/lower value is better. The best results are in **bold**. We also add our results with external data in gray.

Detection Methods	OOD	FPR@ TPR95% ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑	Classification Accuracy ↑
ODIN [27]	Svhn	52.27	83.26	63.76	92.60	**95.02**
	Cifar-100	61.19	78.40	73.21	80.99	**95.02**
	Tiny-ImageNet	59.09	79.69	79.34	77.52	**92.54**
	Texture	42.52	84.06	86.01	80.73	**95.02**
	Lsun	47.85	84.56	81.56	85.58	**95.02**
	Places365	53.94	82.01	54.92	93.30	**93.87**
	Mean/Std	52.00	82.00/2.48	73.13/11.79	85.12/6.59	**94.42/1.03**
EBO [29]	Svhn	**30.56**	92.08	80.95	96.28	**95.02**
	Cifar-100	56.98	79.65	75.09	81.23	**95.02**
	Tiny-ImageNet	57.81	81.65	81.80	78.75	**92.54**
	Texture	52.11	80.70	83.34	75.20	**95.02**
	Lsun	50.56	85.04	82.80	85.29	**95.02**
	Places365	52.16	83.86	58.96	93.90	**93.87**
	Mean/Std	50.03	83.83/4.51	77.16/9.40	85.11/8.44	**94.42/1.03**
Ours	Svhn	**37.72**/24.27	**92.99**/95.93	**87.43**/92.98	**96.70**/98.05	**95.02**
	Cifar-100	**42.32**/39.92	**89.88**/91.45	**89.75**/91.54	**90.24**/91.73	**95.02**
	Tiny-ImageNet	**40.60**/32.41	**90.57**/93.34	**90.59**/93.63	**90.76**/93.41	**92.54**
	Texture	**26.12**/6.86	**94.15**/98.69	**96.33**/99.29	**91.68**/97.71	**95.02**
	Lsun	**43.86**/33.31	**90.61**/93.40	**91.07**/93.85	**90.02**/93.22	**95.02**
	Places365	**42.34**/35.51	**90.16**/92.77	**75.28**/82.25	**96.83**/94.82	**93.87**
	Mean	**38.83**/28.71	**91.39**/94.27	**88.40**/92.26	**92.71**/94.82	**94.42**
	Std	-	**1.75**/2.61	**7.07**/5.57	**3.20**/2.56	**1.03**

Cifar-100, the AUROC drops by 12% to 79.65%. On the contrary, the AUROC for MoodCat are all around a high mean value. This stability may due to that MoodCat utilize semantic information, which is exactly the definition of OOD. Comparing to the classification features or other low-level features, the semantic contradiction exists more generally.

As discussed in Section 3.6, MoodCat can be equipped with external OOD data for better detection ability. In Table 7, we report the performance for MoodCat using Tiny-ImageNet as external unlabeled training data in gray. These results evidence the effectiveness of the training strategy with external data for MoodCat, where significant performance improvements can be found. Due to space limitation, for this setting, we only compare results with baselines on Cifar-100 (see below) and leave that on Cifar-10 benchmarks in Appendix. **Cifar-100 Benchmark.** OE [17], MCD [49] and UDG [45] rely on Tiny-ImageNet as external OOD data for training. We use them as baselines to show the effectiveness of MoodCat under this setting. Note that MoodCat treats all external data as unlabeled during training.

Table 8 shows the results on Cifar-100 benchmarks. As can be seen, though taking advantage of external data, MCD, OE and UDG suffer from the complicated Cifar-100 dataset. By contrast, MoodCat's performance on Cifar-100 is comparable with that on Cifar-10 against same OOD shown in Table 7. This is due to MoodCat's detection mechanism, which relying on the semantic mis-

Table 2: OOD Detection Performance on Cifar-100 as In-D with Tiny-ImageNet as external data for training. All the values are in percentages. ↑/↓ indicates higher/lower value is better. The best results are in **bold**. We also add our results without external data in gray.

Detection Methods	OOD	FPR@ TPR95% ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑	Classification Accuracy ↑
MCD [49]	Svhn	85.82	76.61	65.50	85.52	68.80
	Cifar-10	87.74	73.15	76.51	67.24	68.80
	Tiny-ImageNet	84.46	75.32	85.11	59.49	62.21
	Texture	83.97	73.46	83.11	56.79	68.80
	Lsun	86.08	74.05	84.21	58.62	67.51
	Places365	82.74	76.30	61.15	87.19	70.47
	Mean/Std	85.14	74.82/1.47	75.93/10.31	69.14/13.81	67.77/2.88
OE [17]	Svhn	68.87	84.23	75.11	91.41	70.49
	Cifar-10	79.72	78.92	81.95	74.28	70.49
	Tiny-ImageNet	83.41	76.99	86.36	60.56	63.69
	Texture	86.56	73.89	84.48	54.84	70.49
	Lsun	83.53	77.10	86.28	60.97	69.89
	Places365	78.24	79.62	67.13	88.89	72.02
	Mean/Std	80.06	78.46/3.46	80.22/**7.66**	71.83/15.58	69.51/2.94
UDG [45]	Svhn	60.00	88.25	**81.46**	93.63	68.51
	Cifar-10	83.35	76.18	78.92	71.15	68.51
	Tiny-ImageNet	81.73	77.18	86.00	61.67	61.80
	Texture	75.04	79.53	87.63	65.49	68.51
	Lsun	78.70	76.79	84.74	63.05	67.10
	Places365	73.89	79.87	65.36	89.60	69.83
	Mean/Std	75.45	79.63/4.48	80.69/8.14	74.10/14.01	67.38/**2.87**
Ours	Svhn	58.16/**51.60**	87.38/**88.99**	78.25/80.89	93.81/**94.81**	**76.65**
	Cifar-10	54.31/**50.17**	85.91/**87.76**	86.27/**88.18**	85.91/**87.79**	**76.65**
	Tiny-ImageNet	55.33/**46.07**	86.95/**89.42**	87.55/**89.73**	86.67/**89.28**	**69.56**
	Texture	46.70/**42.22**	89.20/**90.56**	93.48/**94.43**	83.28/**85.13**	**76.65**
	Lsun	53.43/**47.85**	87.98/**89.96**	88.82/**90.33**	87.32/**89.23**	**76.10**
	Places365	54.20/**47.72**	87.41/**89.30**	71.68/**74.83**	95.78/**96.48**	**77.56**
	Mean	53.69/**47.61**	87.47/**89.33**	84.34/**86.40**	88.80/**90.45**	**75.53**
	Std	-	0.95/**1.09**	7.19/**7.94**	4.33/**4.89**	2.96

match. Even when the In-D dataset becomes complicated, the semantic mismatch in OODs is still obvious. For methods detect OODs based on DNN-extracted features, e.g. MCD, OE, UDG, they may suffer from poor decision boundaries as the number of classes increases. Similarly as in the case on Cifar-10, to show the effectiveness of the proposed training strategy with external data, we also report the performance for MoodCat without external data in gray. The improvement can be seen on all metrics across all OOD settings. For space limitation, we leave the comparison on Cifar-100 benchmarks without external data in Appendix.

4.4   Comparison with Open Set Recognition Methods

The experimental protocol in OSR is to randomly selecting K classes from a specific n-class dataset as "known" classes and the left $n-K$ as "unknown" classes. To make a fair comparison, we retrain MoodCat under the experimental settings in CVAECapOSR [11], where only 4 or 6 classes from Cifar-10 are used for training. We report AUROC scores in Table 3, and the results for other methods

Table 3: Comparison with Open Set Recognition methods. AUROC scores on the detection of known and unknown classes. CIFAR indicates splitting Cifar-10 to 6 known classes, and 4 unknown. CIFAR $+N$ samples known 4 classes form Cifar-10, N unknown classes from Cifar-100. The **bold** indicates the best. For more details about dataset splits, please refer to CVAECapOSR [11].

Method	CIFAR	CIFAR $+10$	CIFAR $+50$
OSRCI [32]	$69.9_{\pm 3.8}$	83.8	82.7
C2AE [35]	$71.1_{\pm 0.8}$	$81.0_{\pm 0.5}$	$80.3_{\pm 0.0}$
CVAECapOSR [11]	$83.5_{\pm 2.3}$	$88.8_{\pm 1.9}$	$88.9_{\pm 1.7}$
MoodCat (ours)	$\mathbf{89.48}_{\pm 0.50}$	$\mathbf{89.36}_{\pm 0.74}$	$\mathbf{89.23}_{\pm 0.19}$

Table 4: OOD Detection Performance under different combinations of anomalous scorers. MoodCat is trained on Cifar-10 (In-D) without external data. OODs are from Cifar-100. All the values are in percentages. $\uparrow/\downarrow$ indicates the higher/lower value is better. The best results are in **bold**.

Anomalous Scorer	FPR@TPR95%$\downarrow$	AUROC$\uparrow$	AUPR-In $\uparrow$	AUPR-Out$\uparrow$
$\mathcal{C}_b$	42.80	89.13	88.58	89.85
$LPIPS$	76.62	73.93	72.68	73.23
$DISTS$	82.03	72.14	71.83	70.35
$\mathcal{C}_b + LPIPS$	42.14	89.49	89.18	90.11
$\mathcal{C}_b + DISTS$	42.31	89.35	89.09	89.98
$LPIPS + DISTS$	76.06	74.78	74.25	73.86
$\mathcal{C}_b + LPIPS + DISTS$	**41.95**	**89.57**	**89.30**	**90.16**

are from CVAECapOSR [11]. As can be seen, MoodCat outperforms these methods in all three settings, especially in the original CIFAR setting, where MoodCat outperforms the second best method about 6%.

4.5   Ablation Study

In this section, we first analyze the effectiveness of every anomalous scorer and how scorers cooperate to achieve the final decent performance. Then we conduct experiments on masking, and give insights on masking's effectiveness.

Anomalous scoring model.   Table 4 summarizes the performances of every scorer, every two scorers and all three scorers working together. As can be seen in Table 4, the $\mathcal{C}_b + LPIPS + DISTS$ combination wins the best detection performance in terms of all evaluation metrics, which means our proposed $\mathcal{C}_b$ has a high flexibility to cooperate with IQA models (see Section 3.5). We can also observe that coupling scorers usually lead to a better detection capability than that of any single scorer within the coupling. However, adding extra scorers inevitably increases computational and memory overheads. The cost of basic version of MoodCat, i.e. $\mathcal{C}_b$, **E**, **D**, is relatively small, i.e. Params 4.552M, MACs 0.408G, when compared to that of the widely adopted classifier architectures, e.g., ResNet50 (Params 23.251M, MACs 1.305G). Due to space limitations, we detail the computational and memory costs of MoodCat and those of the baselines in

Table 5: Ablation study on Masking. We set the masking ratio as 0.3 for "Fixed High Ratio" and "Patched", 0.1 for "Fixed Low Ratio", and that of "Randomly" varies from 0.1 to 0.3. MOODCAT employs the **Randomly** masking style.

Training	Inference	FPR@95 ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
w/o mask	w/o mask	40.67	90.79	90.91	90.88
	Randomly	$38.57_{\pm0.75}$	$90.99_{\pm0.18}$	$90.85_{\pm0.21}$	$91.31_{\pm0.19}$
with mask	w/o mask	37.74	91.52	91.5	91.77
	Fix Low Ratio	37.5	91.64	91.6	91.9
	Fix High Ratio	36.16	91.44	91.03	91.9
	Randomly	$\mathbf{36.15}_{\pm0.94}$	$\mathbf{91.78}_{\pm0.17}$	$\mathbf{91.68}_{\pm0.24}$	$\mathbf{92.08}_{\pm0.14}$

Table 6: Ablation study on label conditioning

In-Data	Methods	FPR@95 ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
Cifar10 In-D	**ours**	**41.95**	**89.57**	**89.30**	**90.16**
Cifar100 OOD	uncond.	86.94	63.62	71.26	63.01
Cifar100 In-D	**ours**	**50.17**	**87.76**	**88.18**	**87.79**
Cifar10 OOD	uncond.	96.74	52.48	56.29	71.24

the Appendix. In practice, MOODCAT can achieve appropriate detection ability, by tailoring scorers in MOODCAT's anomalous scoring model according to the application scenario, i.e. trade-off between the performance and costs, and we discussed this part in Appendix.

Masking. To evidence the effectiveness of masking holistically, we conduct another ablation study without masking in training. As shown in Table 5, the masking scheme indeed plays an important role in the performance of MOODCAT. Due to space limitation, we leave the ablation study on masking style in Appendix.

Label Conditioning. As for evaluating label conditioning, we degrade the cGAN to vanilla GAN without conditions. Table 6 reports this ablation study. We can see that our conditioning mechanism outperforms the unconditioned scheme by a large margin. As analyzed in Sec 3.2, the conditioning spotlights the semantic discrepancy between In-D and OOD to facilitate OOD detection.

5   Conclusion

In this paper, we propose a novel plug-and-play OOD detection method for image classifiers, MOODCAT, wherein we consider the semantic mismatch under masking as the distance metric. MOODCAT naturally learns the semantic information from the in-distribution data with the proposed mask and conditional synthesis framework. Experimental results demonstrate significantly better OOD detection capabilities of MOODCAT over SOTA solutions.

References

1. Amodei, D., Olah, C., Steinhardt, J., Christiano, P., Schulman, J., Mané, D.: Concrete problems in ai safety. arXiv preprint arXiv:1606.06565 (2016)
2. Brock, A., Donahue, J., Simonyan, K.: Large scale gan training for high fidelity natural image synthesis. In: International Conference on Learning Representations (2018)
3. Choi, H., Jang, E., Alemi, A.A.: Waic, but why? generative ensembles for robust anomaly detection. arXiv preprint arXiv:1810.01392 (2018)
4. Cimpoi, M., Maji, S., Kokkinos, I., Mohamed, S., , Vedaldi, A.: Describing textures in the wild. In: Proceedings of the IEEE Conf. on Computer Vision and Pattern Recognition (CVPR) (2014)
5. De Vries, H., Strub, F., Mary, J., Larochelle, H., Pietquin, O., Courville, A.C.: Modulating early visual processing by language. Advances in Neural Information Processing Systems **30** (2017)
6. Denouden, T., Salay, R., Czarnecki, K., Abdelzad, V., Phan, B., Vernekar, S.: Improving reconstruction autoencoder out-of-distribution detection with mahalanobis distance. arXiv preprint arXiv:1812.02765 (2018)
7. Dietterich, T.G.: Steps toward robust artificial intelligence. AI Magazine **38**(3), 3–24 (2017)
8. Ding, K., Ma, K., Wang, S., Simoncelli, E.P.: Image quality assessment: Unifying structure and texture similarity. IEEE Transactions on Pattern Analysis and Machine Intelligence (2020)
9. Drummond, N., Shearer, R.: The open world assumption. In: eSI Workshop: The Closed World of Databases meets the Open World of the Semantic Web. vol. 15 (2006)
10. Ge, Z., Demyanov, S., Chen, Z., Garnavi, R.: Generative openmax for multi-class open set classification. In: British Machine Vision Conference 2017. British Machine Vision Association and Society for Pattern Recognition (2017)
11. Guo, Y., Camporese, G., Yang, W., Sperduti, A., Ballan, L.: Conditional variational capsule network for open set recognition. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 103–111 (2021)
12. He, K., Chen, X., Xie, S., Li, Y., Dollár, P., Girshick, R.: Masked autoencoders are scalable vision learners. arXiv preprint arXiv:2111.06377 (2021)
13. He, K., Zhang, X., Ren, S., Sun, J.: Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In: Proceedings of the IEEE International Conference on Computer Vision. pp. 1026–1034 (2015)
14. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 770–778 (2016)
15. Hein, M., Andriushchenko, M., Bitterwolf, J.: Why relu networks yield high-confidence predictions far away from the training data and how to mitigate the problem. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 41–50 (2019)
16. Hendrycks, D., Gimpel, K.: A baseline for detecting misclassified and out-of-distribution examples in neural networks. In: 5th International Conference on Learning Representations (ICLR) (2017)
17. Hendrycks, D., Mazeika, M., Dietterich, T.: Deep anomaly detection with outlier exposure. In: International Conference on Learning Representations (2018)

18. Huang, G., Liu, Z., Pleiss, G., Van Der Maaten, L., Weinberger, K.: Convolutional networks with dense connectivity. IEEE Transactions on Pattern Analysis and Machine Intelligence pp. 1–1 (2019)
19. Huang, H., Li, Z., Wang, L., Chen, S., Zhou, X., Dong, B.: Feature space singularity for out-of-distribution detection. In: Proceedings of the Workshop on Artificial Intelligence Safety (SafeAI) (2021)
20. Huang, R., Geng, A., Li, Y.: On the importance of gradients for detecting distributional shifts in the wild. Advances in Neural Information Processing Systems **34** (2021)
21. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. In: 2nd International Conference on Learning Representations (ICLR) (2014)
22. Kirichenko, P., Izmailov, P., Wilson, A.G.: Why normalizing flows fail to detect out-of-distribution data. Advances in neural information processing systems **33**, 20578–20589 (2020)
23. Krizhevsky, A., Hinton, G., et al.: Learning multiple layers of features from tiny images (2009)
24. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in Neural Information Processing Systems. vol. 25, pp. 1106–1114 (2012)
25. Le, Y., Yang, X.: Tiny imagenet visual recognition challenge. CS 231N **7**(7), 3 (2015)
26. Lee, K., Lee, K., Lee, H., Shin, J.: A simple unified framework for detecting out-of-distribution samples and adversarial attacks. Advances in neural information processing systems **31** (2018)
27. Liang, S., Li, Y., Srikant, R.: Enhancing the reliability of out-of-distribution image detection in neural networks. In: 6th International Conference on Learning Representations, ICLR 2018 (2018)
28. Lin, Z., Roy, S.D., Li, Y.: Mood: Multi-level out-of-distribution detection. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 15313–15323 (2021)
29. Liu, W., Wang, X., Owens, J., Li, Y.: Energy-based out-of-distribution detection. In: Advances in Neural Information Processing Systems. vol. 33, pp. 21464–21475. Curran Associates, Inc. (2020)
30. Miyato, T., Koyama, M.: cgans with projection discriminator. In: International Conference on Learning Representations (2018)
31. Nalisnick, E., Matsukawa, A., Teh, Y.W., Gorur, D., Lakshminarayanan, B.: Do deep generative models know what they don't know? In: International Conference on Learning Representations (2019)
32. Neal, L., Olson, M., Fern, X., Wong, W.K., Li, F.: Open set learning with counterfactual images. In: Proceedings of the European Conference on Computer Vision (ECCV). pp. 613–628 (2018)
33. Netzer, Y., Wang, T., Coates, A., Bissacco, A., Wu, B., Ng, A.Y.: Reading digits in natural images with unsupervised feature learning. In: NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011 (2011)
34. Nguyen, A., Yosinski, J., Clune, J.: Deep neural networks are easily fooled: High confidence predictions for unrecognizable images. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 427–436 (2015)
35. Oza, P., Patel, V.M.: C2ae: Class conditioned auto-encoder for open-set recognition. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 2307–2316 (2019)

36. Pidhorskyi, S., Almohsen, R., Doretto, G.: Generative probabilistic novelty detection with adversarial autoencoders. Advances in neural information processing systems **31** (2018)
37. Ren, J., Liu, P.J., Fertig, E., Snoek, J., Poplin, R., Depristo, M., Dillon, J., Lakshminarayanan, B.: Likelihood ratios for out-of-distribution detection. Advances in Neural Information Processing Systems **32** (2019)
38. Sara, U., Akter, M., Uddin, M.S.: Image quality assessment through fsim, ssim, mse and psnr—a comparative study. Journal of Computer and Communications **7**(3), 8–18 (2019)
39. Schlegl, T., Seeböck, P., Waldstein, S.M., Schmidt-Erfurth, U., Langs, G.: Unsupervised anomaly detection with generative adversarial networks to guide marker discovery. In: International conference on information processing in medical imaging. pp. 146–157. Springer (2017)
40. Schonfeld, E., Schiele, B., Khoreva, A.: A u-net based discriminator for generative adversarial networks. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 8207–8216 (2020)
41. Sricharan, K., Srivastava, A.: Building robust classifiers through generation of confident out of distribution examples. arXiv preprint arXiv:1812.00239 (2018)
42. Vernekar, S., Gaurav, A., Abdelzad, V., Denouden, T., Salay, R., Czarnecki, K.: Out-of-distribution detection in classifiers via generation. arXiv preprint arXiv:1910.04241 (2019)
43. Wang, H., Liu, W., Bocchieri, A., Li, Y.: Can multi-label classification networks know what they don't know? Advances in Neural Information Processing Systems **34** (2021)
44. Wang, Y., Li, B., Che, T., Zhou, K., Liu, Z., Li, D.: Energy-based open-world uncertainty modeling for confidence calibration. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 9302–9311 (2021)
45. Yang, J., Wang, H., Feng, L., Yan, X., Zheng, H., Zhang, W., Liu, Z.: Semantically coherent out-of-distribution detection. In: Proceedings of the IEEE/CVF International Conference on Computer Vision. pp. 8301–8309 (2021)
46. Yang, J., Zhou, K., Li, Y., Liu, Z.: Generalized out-of-distribution detection: A survey. arXiv preprint arXiv:2110.11334 (2021)
47. Yang, Y., Gao, R., Li, Y., Lai, Q., Xu, Q.: What you see is not what the network infers: Detecting adversarial examples based on semantic contradiction. In: Network and Distributed System Security Symposium (NDSS) (2022)
48. Yu, F., Seff, A., Zhang, Y., Song, S., Funkhouser, T., Xiao, J.: Lsun: Construction of a large-scale image dataset using deep learning with humans in the loop (2016)
49. Yu, Q., Aizawa, K.: Unsupervised out-of-distribution detection by maximum classifier discrepancy. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV) (2019)
50. Zaeemzadeh, A., Bisagno, N., Sambugaro, Z., Conci, N., Rahnavard, N., Shah, M.: Out-of-distribution detection using union of 1-dimensional subspaces. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 9452–9461 (2021)
51. Zagoruyko, S., Komodakis, N.: Wide residual networks. In: British Machine Vision Conference 2016. British Machine Vision Association (2016)
52. Zhang, H., Goodfellow, I., Metaxas, D., Odena, A.: Self-attention generative adversarial networks. In: International Conference on Machine Learning (2019)
53. Zhang, R., Isola, P., Efros, A.A., Shechtman, E., Wang, O.: The unreasonable effectiveness of deep features as a perceptual metric. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. pp. 586–595 (2018)

54. Zhou, B., Lapedriza, A., Khosla, A., Oliva, A., Torralba, A.: Places: A 10 million image database for scene recognition. IEEE Transactions on Pattern Analysis and Machine Intelligence (2017)

Appendix

A   Training Process of Generative Model

A.1   Objective Functions

We implement the adversarial loss with a U-net based discriminator [40], denoted as D^{Unet}. D^{Unet} contains two components: D_{enc}^{Unet} and D_{dec}^{Unet}. $D_{enc}^{Unet} \in \mathbb{R}$ provides the real/fake decision as a scalar. While $D_{dec}^{Unet} \in \mathbb{R}^I$ generates a per-pixel real/fake map for the input image, where $I = h \times w$ indicates the scale of input image. Compared to the vanilla discriminator, D^{Unet} not only determines whether the input image is realistic or fake, but also tries to locate the fake parts. Empowered by the per-pixel real/fake map, our generative model can be optimized to focus more on structural semantic features and synthesize coherent image both globally and locally as desired. We formulate the adversarial loss for the discriminator in Eq. (3)-Eq. (5):

$$\mathcal{L}_{D^{Unet}} = \mathcal{L}_{D_{enc}^{Unet}} + \mathcal{L}_{D_{dec}^{Unet}}, \tag{3}$$

$$\mathcal{L}_{D_{enc}^{Unet}} = -\mathbb{E}_x[\log D_{enc}^{Unet}(x,y)] - \mathbb{E}_x[\log(1 - D_{enc}^{Unet}(x',y))], \tag{4}$$

$$\mathcal{L}_{D_{dec}^{Unet}} = -\mathbb{E}_x\Big[\sum_I \log D_{dec}^{Unet}(x,y)\Big] - \mathbb{E}_x\Big[\sum_I \log(1 - D_{dec}^{Unet}(x',y))\Big], \tag{5}$$

where $\mathcal{L}_{D^{Unet}}$ and $\mathcal{L}_{D_{dec}^{Unet}}$ are the loss functions for D_{enc}^{Unet} and D_{dec}^{Unet}, respectively. Correspondingly, the adversarial loss applied on the generator is as follow:

$$\mathcal{L}_{\mathbf{G}} = -\mathbb{E}_x\Big[\log D_{enc}^{Unet}(x',y) + \sum_I \log D_{dec}^{Unet}(x')\Big] + \ell_1(x,x') + \ell_2(x,x') + \mathcal{SSIM}(x,x'). \tag{6}$$

A.2   Training Process

Encoder. We adopt a four-layer convolutional neural network as the feature extractor for Encoder, then two fully-connected layers are employed to output μ and Σ. The dimension of the latent variable z is set at 128.

Decoder. We employ the generator architecture proposed in [2] as our Decoder's backbone, then reset the input size to (3, 32, 32), and the channel multiplier to 32, which represents the number of units in each layer [2]. The input latent variable size equals 128.

Discriminator. We build D^{Unet} based on the implementation of [40], changing the channel multiplier to 32.

All three models mentioned above are trained from scratches in an end-to-end way. We use Adam [?] as the optimizer, with $\beta_1 = 0$, $\beta_2 = 0.999$, learning rate fixed at $5 \cdot 10^{-5}$. The batch size is set at 96. We detail the training process of our generative model in Algorithm 1.

Algorithm 1: Training Framework of **G**

Input   : Training data $\mathcal{X} = \{x\}^N$, $\mathcal{Y} = \{y\}^N$, the random mask **M**
Output : The parameters of **E**, **D**
1 for *some training iterations* **do**
2 | $x' = \mathbf{G}(\mathbf{M}(x), y) = \mathbf{D}(\mathbf{E}(\mathbf{M}(x), y))$;
3 | Feed (x, y) and (x', y) into D^{Unet}, respectively;
4 | Optimize **D** and **E** for $\mathcal{L}_{\mathbf{G}}$(Eq. (6)) and $\mathcal{L}_{KLD}$;
5 | Optimize D^{Unet} for $\mathcal{L}_{D^{Unet}}$ (Eq. (3));
6 end
7 return E, **G**

B   Quantitative Results

In this section, we provide more experimental results on CIFAR-10 and CIFAR-100 benchmarks, respectively. Furthermore, to further validate the effectiveness of the proposed *conditional binary classifier* $(\mathcal{C}_b)$ in the anomalous scoring model, we detail its performance in each OOD dataset by varying the type of $\mathcal{C}_b$, i.e. trained with/without external OOD data.

B.1   More Results on CIFAR-10 Benchmarks

Table 7 presents the comparison of our MOODCAT trained with external unlabeled data sourced from TINY-IMAGENET, and baselines implemented with extra data. We conclude that MOODCAT outperforms or at least on par with baselines on CIFAR-10 benchmarks.

Additionally, in Table 7 we observe that OE and UDG achieve a much better performance on SVHN than on other OOD datasets. In fact, most street number images contained in SVHN have relatively flat backgrounds, as shown in Fig. 6 and Fig. 7's SVHN columns. In this case, OE and UDG can achieve excellent performance by overfitting to this specific low-level feature of SVHN instead of considering the semantic level change caused by SVHN. Thus, when encountering a more challenging case, e.g., CIFAR-100, which has the same data source as CIFAR-10 but different semantic meanings, both OE and UDG suffer a noticeable performance degradation. In contrast, MOODCAT identifies OOD according to their semantic mismatch, thus, remains stable performance on various OODs.

Table 7: OOD Detection Performance on CIFAR-10 benchmarks, MOODCAT trained with external OOD data. All the values are in percentages. ↑/↓ indicates higher/lower value is better. The best results are in **bold**.

Detection Methods	OOD	FPR@ TPR95% ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑	Classification Accuracy ↑
MCD	SVHN	60.27	89.78	85.33	94.25	90.56
	CIFAR-100	74.00	82.78	83.97	79.16	90.56
	TINY-IMAGENET	78.89	80.98	85.63	72.48	87.33
	TEXTURE	83.92	81.59	90.20	63.27	90.56
	LSUN	68.96	84.71	85.74	81.50	90.56
	PLACES365	72.08	83.51	69.44	92.52	88.51
	Mean	73.02	83.89	83.39	80.53	89.68
OE	SVHN	20.88	96.43	93.62	98.32	91.87
	CIFAR-100	58.54	86.22	86.17	84.88	91.87
	TINY-IMAGENET	58.98	87.65	90.09	82.16	89.27
	TEXTURE	51.17	89.56	93.79	81.88	91.87
	LSUN	57.97	86.75	87.69	85.07	91.87
	PLACES365	55.64	87.00	73.11	94.67	90.99
	Mean	50.53	88.93	87.55	87.83	91.29
UDG	SVHN	13.26	**97.49**	**95.66**	**98.69**	92.94
	CIFAR-100	47.20	90.98	**91.74**	89.36	92.94
	TINY-IMAGENET	50.18	91.91	**94.43**	86.99	90.22
	TEXTURE	20.43	96.44	98.12	92.91	92.94
	LSUN	42.05	93.21	94.53	91.03	92.94
	PLACES365	44.22	92.64	87.17	96.66	91.68
	Mean	36.22	93.78	93.61	92.61	92.28
Ours	SVHN	24.27	95.93	92.98	98.05	**95.02**
	CIFAR-100	**39.92**	**91.45**	91.54	**91.73**	**95.02**
	TINY-IMAGENET	**32.41**	**93.34**	93.63	**93.41**	**92.54**
	TEXTURE	**6.86**	**98.69**	**99.29**	**97.71**	**95.02**
	LSUN	**33.31**	**93.40**	**93.85**	**93.22**	**95.02**
	PLACES365	**35.51**	**92.77**	82.25	94.82	**93.87**
	Mean	**28.71**	**94.27**	92.26	**94.82**	**94.42**

B.2   More Results on CIFAR-100 benchmarks

Table 8 shows the comparison of our MOODCAT trained without external OOD data, and baselines are implemented under the same setting. We conclude that MOODCAT achieves state-of-the-art performance on CIFAR-100 benchmarks.

B.3   Ablation Study on Conditional Binary Classifier

To study how much the proposed *Conditional Binary Classifier* ($\mathcal{C}_b$) contributes to MOODCAT, we conduct several ablations on $\mathcal{C}_b$. More specifically, we consider three configurations: $\mathcal{C}_b$, $\mathcal{C}_b$(TINY-IMAGENET), and $\mathcal{C}_b$+ $\mathcal{C}_b$(TINY-IMAGENET), where $\mathcal{C}_b$ referring to the Conditional Binary Classifier trained only on In-D samples, $\mathcal{C}_b$(TINY-IMAGENET) denoted the Conditional Binary Classifier using TINY-IMAGENET as extra training data and $\mathcal{C}_b$+ $\mathcal{C}_b$(TINY-IMAGENET) indicating that $\mathcal{C}_b$ and $\mathcal{C}_b$ (TINY-IMAGENET) are used in a cascade way.

Table 9 and Table 10 demonstrate $\mathcal{C}_b$'s performance on CIFAR-10 and CIFAR-100 benchmarks in six OOD datasets, respectively. The main takeaways are: **(1)** $\mathcal{C}_b$ or $\mathcal{C}_b$(TINY-IMAGENET) alone can achieve acceptable performance; **(2)** $\mathcal{C}_b$(TINY-IMAGENET) outperforms $\mathcal{C}_b$, which means that adding external unlabeled data

Table 8: OOD Detection Performance on CIFAR-100 as In-D, MOODCAT training without external data. All the values are in percentages. ↑/↓ indicates higher/lower value is better. The best results are in **bold**.

Detection Methods	OOD	FPR@ TPR95% ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑	Classification Accuracy ↑
ODIN	SVHN	90.33	75.59	65.25	84.49	76.65
	CIFAR-10	81.28	77.90	79.93	73.39	76.65
	TINY-IMAGENET	82.74	77.58	86.26	61.38	69.56
	TEXTURE	79.47	77.92	86.69	62.97	76.65
	LSUN	80.57	78.22	86.34	63.44	76.10
	PLACES365	76.42	80.66	66.77	89.66	77.56
	Mean	81.89	77.98	78.54	72.56	75.53
EBO	SVHN	78.23	83.57	75.61	90.24	76.65
	CIFAR-10	81.25	78.95	80.01	74.44	76.65
	TINY-IMAGENET	83.32	78.34	87.08	62.13	69.56
	TEXTURE	84.29	76.32	85.87	59.12	76.65
	LSUN	84.51	77.66	86.42	61.40	76.10
	PLACES365	78.37	80.99	68.22	89.60	77.56
	Mean	81.66	79.31	80.54	72.82	75.53
Ours	SVHN	**58.16**	**87.38**	**78.25**	**93.81**	**76.65**
	CIFAR-10	**54.31**	**85.91**	**86.27**	**85.91**	**76.65**
	TINY-IMAGENET	**55.33**	**86.95**	**87.55**	**86.67**	**69.56**
	TEXTURE	**46.70**	**89.20**	**93.48**	**83.28**	**76.65**
	LSUN	**53.43**	**87.98**	**88.82**	**87.32**	**76.10**
	PLACES365	**54.20**	**87.41**	**71.68**	**95.78**	**77.56**
	Mean	**53.69**	**87.47**	**84.34**	**88.80**	**75.53**

into the training process can improve the detection ability; **(3)** coupling scorers, here $\mathcal{C}_b + \mathcal{C}_b$(TINY-IMAGENET), usually leads to a better detection capability than that of any single scorer within the coupling. Above findings align with what we have reported in our paper, and further indicate that $\mathcal{C}_b$ plays a key role in the proposed anomalous scoring model.

B.4   Ablation Study on Masking Style

We try several masking forms as exemplified in Fig. 4, and summarize the corresponding experimental results in Table 11. Experiments show the randomly masking outperforms other strategies.

From the first three rows in Table 11, we notice that masking can indeed help with performance improvement. However, as we can observe from the second column in Fig. 4, a fixed mask with high ratio (e.g., 0.3) can lead the synthesis to loss of fine details. In addition, we implement a patched masking like [12]. However, such masking style may break the continuity within the image, thus lead to low quality on the synthesis for In-D. We also try a non-masking strategy, shuffling, but it further breaks the continuity of the image. Finally, we identify that the most effective strategy is randomly masking. As can be seen, both the quality of the synthesis in Fig. 4 and the overall performance in Table 11 outperform other strategies.

Table 9: Conditional Binary Classifier Performance on Cifar-10 benchmarks. All the values are in percentages. ↑/↓ indicates higher/lower value is better. The best results are in **bold**. $\mathcal{C}_b$ and $\mathcal{C}_b$(Tiny-ImageNet) indicates the proposed model trained without/with external unlabeled Tiny-ImageNet data, respectively.

Anomalous Scoring Model	OOD	FPR@ TPR95% ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
$\mathcal{C}_b$	Svhn	48.01	86.85	75.20	94.34
	Cifar-100	42.80	89.13	88.58	89.85
	Tiny-ImageNet	40.54	89.78	89.27	90.42
	Texture	42.54	87.47	91.33	83.85
	Lsun	43.76	90.15	90.18	90.17
	Places365	43.49	89.40	72.82	96.65
	Mean	43.52	88.80	84.56	90.88
$\mathcal{C}_b$ (Tiny-ImageNet)	Svhn	39.47	91.49	83.58	96.23
	Cifar-100	37.43	91.31	91.02	91.73
	Tiny-ImageNet	31.92	93.10	93.01	93.34
	Texture	25.74	94.25	96.17	91.89
	Lsun	32.74	93.55	93.83	93.40
	Places365	34.45	92.78	81.48	97.71
	Mean	33.63	92.75	89.85	94.05
$\mathcal{C}_b +$ $\mathcal{C}_b$ (Tiny-ImageNet)	Svhn	**39.44**	**91.50**	**83.60**	**96.25**
	Cifar-100	**36.64**	**91.40**	**91.15**	**91.85**
	Tiny-ImageNet	**31.86**	**93.12**	**93.04**	**93.38**
	Texture	**25.37**	**94.34**	**96.24**	**92.00**
	Lsun	**32.67**	**93.55**	**93.84**	**93.41**
	Places365	**34.42**	**92.79**	**81.51**	**97.72**
	Mean	**33.40**	**92.78**	**89.90**	**94.10**

B.5   Experiments on Advanced Classifier architectures

We empower UDG with wider (WRN28) and deeper (DenseNet) classifier. Table 12 shows the comparison results with Cifar-100 as In-D samples using WRN28 and DenseNet architecture.

As can be observed from the table, while UDG performs better on these architectures compared to ResNet18, it still lags far behind our results.

C   Qualitative Results

In this section, we demonstrate several batches of visual examples of MoodCat including both In-D and OOD cases.

In-D samples with their syntheses. Fig. 5 visualizes In-D samples and their corresponding syntheses from Cifar-10 and Cifar-100, respectively. Note that we expect the syntheses to resemble the input images for In-D samples with correct labels.

OOD samples with their syntheses. Fig. 6 visualizes OOD samples from six datasets, which are employed in the Cifar-10 benchmarks, along with their corresponding masked images and the syntheses generated by our MoodCat.

Table 10: Conditional Binary Classifier Performance on Cifar-100 benchmarks. All the values are in percentages. ↑/↓ indicates higher/lower value is better. The best results are in **bold**. $\mathcal{C}_b$ and $\mathcal{C}_b$(Tiny-ImageNet) indicates the proposed model trained without/with external unlabeled Tiny-ImageNet data, respectively.

Anomalous Scoring Model	OOD	FPR@ TPR95% ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
$\mathcal{C}_b$	Svhn	65.18	81.32	65.61	91.35
	Cifar-10	55.11	85.75	85.78	85.99
	Tiny-ImageNet	54.69	86.27	86.26	86.43
	Texture	56.63	83.30	88.40	77.17
	Lsun	54.77	86.96	87.20	86.83
	Places365	54.18	86.36	67.60	95.54
	Mean	56.76	84.99	80.14	87.22
$\mathcal{C}_b$ (Tiny-ImageNet)	Svhn	54.61	**86.30**	**74.30**	93.80
	Cifar-10	49.82	87.57	87.74	87.69
	Tiny-ImageNet	45.86	89.38	89.48	89.43
	Texture	48.24	87.16	91.55	81.83
	Lsun	44.43	90.07	90.25	90.00
	Places365	46.89	88.93	72.99	96.41
	Mean	48.31	88.24	84.39	89.86
$\mathcal{C}_b$+ $\mathcal{C}_b$ (Tiny-ImageNet)	Svhn	**54.31**	**86.30**	**74.30**	**93.81**
	Cifar-10	**49.62**	**87.60**	**87.77**	**87.77**
	Tiny-ImageNet	**45.46**	**89.39**	**89.48**	**89.48**
	Texture	**47.18**	**87.37**	**91.71**	**82.17**
	Lsun	**44.01**	**90.08**	**90.26**	**90.04**
	Places365	**46.73**	**88.95**	**73.02**	**96.43**
	Mean	**47.89**	**88.28**	**84.42**	**89.95**

In Fig. 7, the In-D dataset changes to Cifar-100. We employed OOD samples sourced from the same six OOD datasets as those of the Cifar-100 benchmarks in Fig. 7. Note that when OOD is fed to MoodCat, we prefer to have a clear distinction between the synthesis generated by MoodCat and the input image.

D   Further Discussion

D.1   Computational Cost Analysis

MoodCat is designed as an auxiliary model that works in parallel with the classifier. This auxiliary architecture ensures MoodCat a plug-and-play model without compromising the classifier's accuracy. Meanwhile, MoodCat can satisfy high-performance requirements in the context of OOD detection. However, as an auxiliary model, MoodCat inevitably introduces extra computation and memory costs.

Table 13 summarizes the computational cost of MoodCat, and that of ODIN, i.e., ResNet18, and that of widely adopted classifier architectures, ResNet18, WResNet28, WResNet101 in terms of number of multiply add operations (MAC), and number of model parameters (Params). As can be observed, the cost of basic

Fig. 4: Visualization of different masking styles and their impacts on synthesized images. The semantic label is assigned as "car" for both the In-D image and the OOD image. We set the masking ratio as 0.3 for "Fixed High Ratio" and "Patched", 0.1 for "Fixed Low Ratio", and that of "Randomly" varies from 0.1 to 0.3. MoodCat employs the **Randomly** masking style.

Table 11: Ablation studies on different masking styles. The results are obtained by setting Cifar-10 as In-D, Cifar-100 as OOD, with MoodCat trained on extra Tiny-ImageNet acting as OOD. The **bolded** values are the highest performance. All the values are in percentages. ↑/↓ indicates higher/lower value is better.

Mask Style	FPR@TPR95% ↓	AUROC ↑	AUPR-In ↑	AUPR-Out ↑
Without Masking	40.53	91.26	91.25	91.55
Fixed Low Ratio	40.20	91.33	91.32	91.59
Fixed High Ratio	39.57	91.56	91.48	91.87
Patched	39.81	91.34	91.33	91.64
Shuffling	44.14	88.73	88.15	89.18
Randomly	**39.48**	**91.66**	**91.65**	**91.95**

In-D
(CIFAR10)

Masked
sample

Synthesis
for *groundtruth*

In-D
(CIFAR10)

Masked
sample

Synthesis
for *groundtruth*

In-D
(CIFAR100)

Masked
sample

Synthesis
for *groundtruth*

In-D
(CIFAR100)

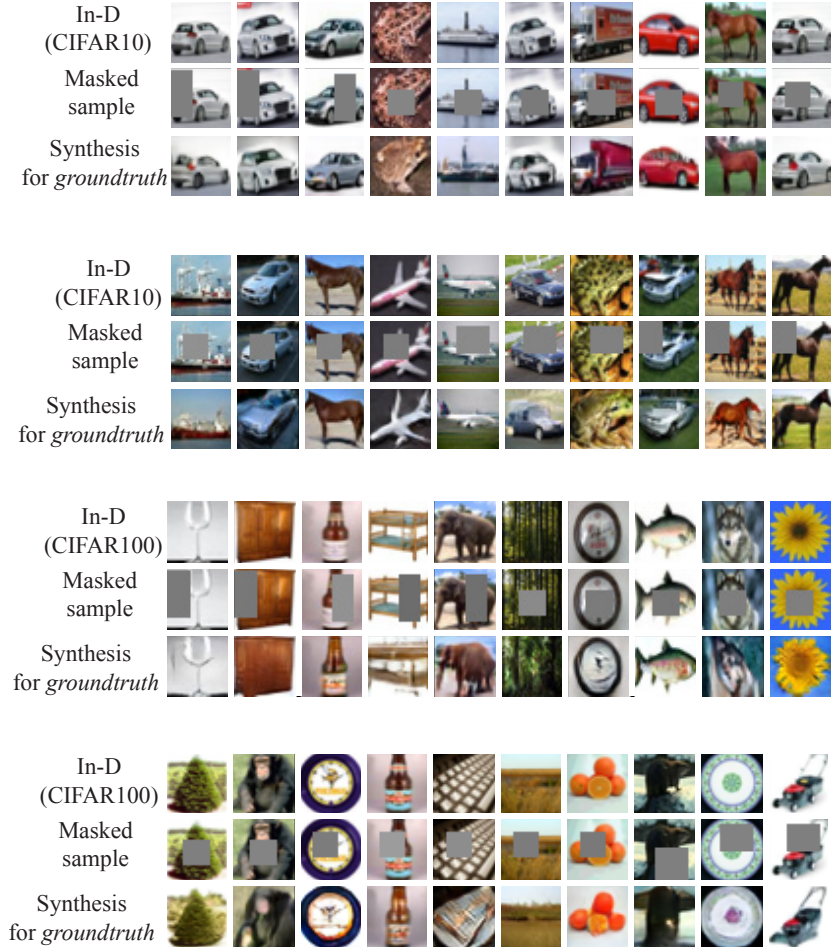Masked
sample

Synthesis
for *groundtruth*

Fig. 5: Visualization results of MooDCat with Cifar-10/ Cifar-100 as In-D. We exemplify several In-D samples in each panel's first row, following the intermediate masked version, and the last row presents their corresponding synthetic version generated by MooDCat with the groundtruth labels.
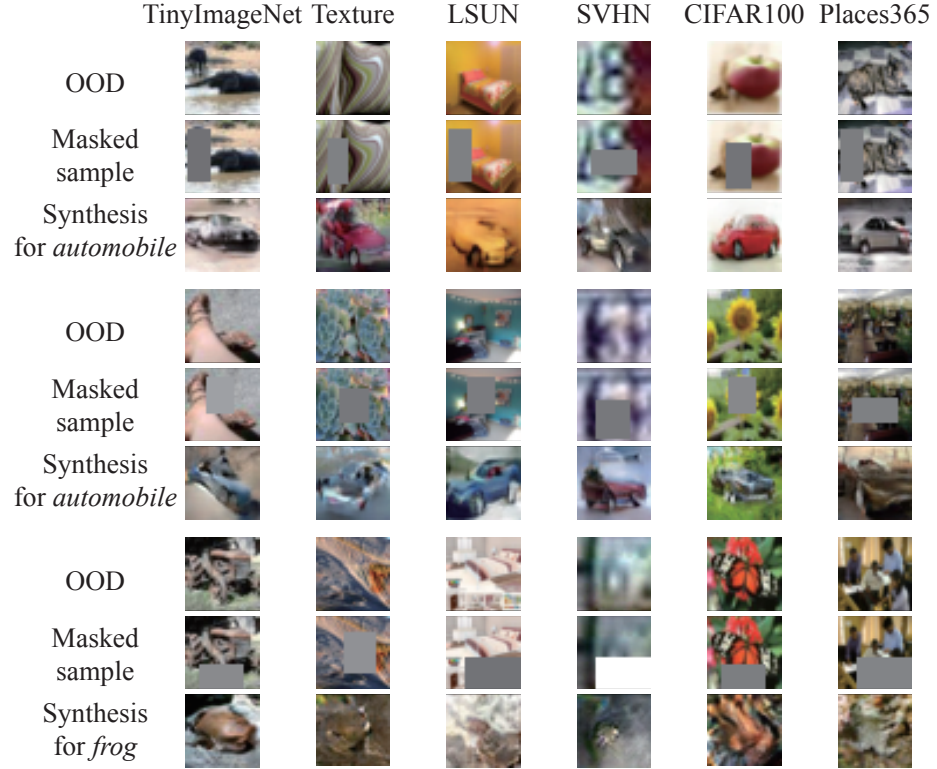
Fig. 6: OOD visualization results of MOODCAT trained on CIFAR-10. In each panel, we exemplify OOD samples across six OOD datasets in the first row, following is the intermediate masked version, the last row presents their corresponding synthetic version generated by MOODCAT with the given semantic label, the same below.
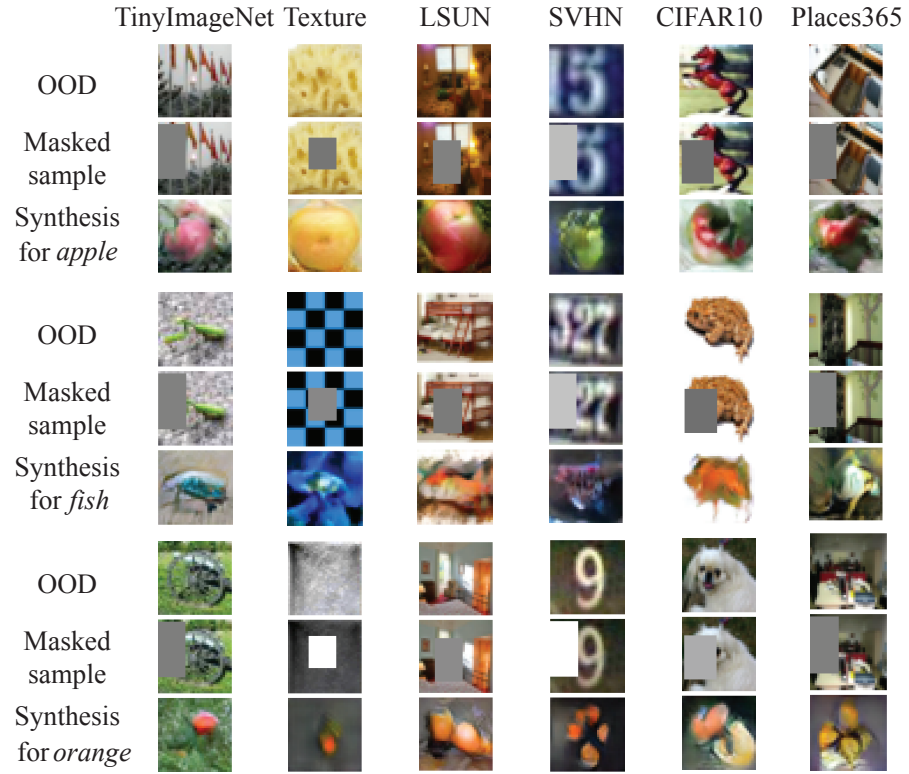
Fig. 7: OOD visualization results of MoodCat trained on Cifar-100.

Table 12: Experiments on advanced model architectures. Performance comparison with UDG on Cifar-100 benchmarks. For our method, we use the results in the main paper with a ResNet18 classifier. We give advantage to UDG, which is reimplemented with deeper/wider WideResNet-28, DenseNet, while MoodCat's parameter number is equivalent to ResNet18. **Bold** are the best.

Architecture	OOD dataset	FPR@TPR95 ↓	AUROC ↑	AUPR In ↑	AUPR Out ↑
WideResNet28 UDG	Svhn	66.76	85.29	76.14	92.33
	Cifar-10	82.35	76.67	78.52	72.63
	Tiny-ImageNet	78.91	79.04	87.00	65.06
	Texture	73.62	79.01	85.53	67.08
	Lsun	77.04	79.79	87.49	66.93
	Places365	72.25	81.49	66.72	90.65
	Mean$_{\pm Std}$	$75.16_{\pm 5.49}$	$80.22_{\pm 2.93}$	$80.23_{\pm 8.11}$	$75.78_{\pm 12.44}$
DenseNet UDG	Svhn	80.67	75.54	75.65	70.99
	Cifar-10	85.87	74.06	77.16	68.90
	Tiny-ImageNet	82.36	76.81	85.76	61.56
	Texture	76.32	78.93	63.79	89.02
	Lsun	79.12	78.91	66.83	88.23
	Places365	73.59	76.27	82.76	65.20
	Mean$_{\pm Std}$	$79.66_{\pm 4.36}$	$76.75_{\pm 1.92}$	$75.33_{\pm 8.64}$	$73.98_{\pm 11.79}$
MoodCat (Ours, Res18)	Svhn	**51.6**	**88.99**	**80.89**	**94.81**
	Cifar-10	**50.17**	**87.76**	**88.18**	**87.79**
	Tiny-ImageNet	**46.07**	**89.42**	**89.73**	**89.28**
	Texture	**42.22**	**90.56**	**94.43**	**85.13**
	Lsun	**47.85**	**89.96**	**90.33**	**89.23**
	Places365	**47.72**	**89.3**	**74.83**	**96.48**
	Mean$_{\pm Std}$	$\mathbf{47.61}_{\pm 3.29}$	$\mathbf{89.33}_{\pm 9.95}$	$\mathbf{86.4}_{\pm 7.19}$	$\mathbf{90.45}_{\pm 4.33}$

version of MoodCat, i.e. $\mathcal{C}_b$, **E**, **D**, is relatively small, Params 4.552M, MACs 0.408G, when compared to that of ResNet18 (Params 11.174M, MAC 0.556G) and other widely adopted classifier architectures, e.g., WResNet28 (Params 36.479M, MACs 5.248G). Note that the performance of basic MoodCat, whose anomalous scoring model only contains $\mathcal{C}_b$, is still acceptable as shown in Table 9 and Table 10. Thus, if computational cost is a real concern in practice, the operator can adopt MoodCat with $\mathcal{C}_b$ alone as an anomalous scorer. For the MoodCat supported by IQA models, e.g., LPIPS, DISTS, the total computational cost is comparable to that of WResNet28 or WResNet101, yet slightly larger than ResNet18. Thus, if the detection ability is put at the first place, one can explore to enhance the anomalous scoring model by employing extra IQA models. Actually, there is a trade-off between the OOD detection performance and the computational cost of MoodCat, and our anomalous scoring model leaves the design space for the deployer to explore according to the real-world application.

Table 13: Computational and memory costs of MoodCat and its components.

Model	E	D	$\mathcal{C}_b$	MoodCat basic	LPIP /DISTS	MoodCat	ResNet 18	WResNet 28	WResNet 101
Params (M)	0.460	3.821	0.271	**4.552**	14.715	33.982	11.174	36.479	126.89
MACs (G)	0.0049	0.297	0.105	**0.408**	0.630	1.718	0.556	5.248	22.84

OOD (CIFAR100)					
Masked sample					
Synthesis					
horse	*fog*	*dog*	*ship*	*ship*	

(a) False Positive

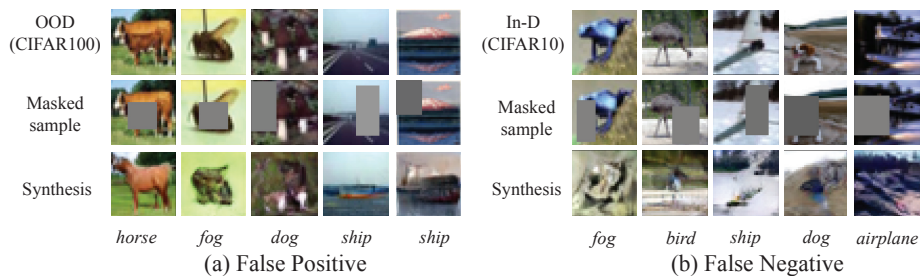|					
In-D (CIFAR10)					
Masked sample					
Synthesis					
fog	*bird*	*ship*	*dog*	*airplane*	

(b) False Negative

Fig. 8: Failure cases of MoodCat. We exemplify both False Positive and False Negative failure cases in (a) and (b), respectively. (a) False Positive failure cases, where samples come from OOD dataset (Cifar-100) are falsely identified as In-D samples (Cifar-10). (b) False Negative failure cases, where samples belong to In-D are wrongly flagged as OOD samples. The predicted label for each input sample are provided under the corresponding synthetic image.

D.2   Failure Cases

Fig. 8 demonstrates some of MoodCat's failure cases. In Fig. 8 (a), the OOD samples sourcing from Cifar-100, are falsely distinguished as In-D samples (Cifar-10). As can be seen, OODs and their synthetic images resemble to each other for same degree. For example, the first column's "cattle" partly contains some features such as legs and the tail, which match the given semantic label "horse" well, resulting in the synthesis having high image quality while resembling to the input image, therefore leading to the final misjudgement.

Fig. 8 (b) presents several False Negative samples, i.e., samples sourcing from In-D are wrongly predicted as OOD samples. As can be observed, the In-D sample with rare characteristics, e.g. a blue fog, an ostrich with its head down, are more likely to be misclassified as OOD. In addition, if the mask happens to cover the object completely, MoodCat can hardly recover the input image without necessary features, as the cases shown in the third and fourth columns of Fig. 8 (b). Moreover, an poor semantic meaning in the In-D sample itself can lead to the final misclassification. For example, in the last column of Fig. 8 (b), even humans can hardly tell what is depicted in the input image, let alone MoodCat.