A Related Work

OOD Detection Methods. OOD detection has received considerable attention in recent years with the need for trustworthy predictions from models (Fang et al., 2022; Galil et al., 2022). Existing methods in OOD detection can be mainly classified into post-hoc methods and fine-tuning methods according to whether need to adjust the model parameters. Further, fine-tuned methods can be classified as the representation-based methods, OOD data generation methods and outlier exposure methods (Yang et al., 2021). For the post-hoc methods, they believe a well-trained ID classifier can already lead to effective OOD detection (Hendrycks & Gimpel, 2016), constructing appropriate OOD score function to distinguish ID and OOD data. Some methods build OOD score function based on the logit of the classifier output (Hendrycks & Gimpel, 2016; Liang et al., 2018a; Liu et al., 2020; Sun et al., 2021; Wang et al., 2021), gradient (Liang et al., 2018b; Huang et al., 2021; Igoe et al., 2022), and embedding feature (Sun et al., 2022; Lee et al., 2018b; Sastry & Oore, 2020).

Fine-tuning based methods consider that the training process can further adjust the latent space, which is beneficial for the model to better separate ID and OOD in different scenarios. For the representation-based methods, recent works has found that good feature representations are beneficial for separating ID and OOD. Some approaches attempt to utilize data augmentation (Tack et al., 2020; Sun et al., 2022), constative learning (Sehwag et al., 2020; Wang et al., 2022) and constraints on embedding features (Ming et al., 2023; Wei et al., 2022) to achieve enhanced representation. The adopted scoring functions in representation-based methods, however, can be complex. This complexity may lead to an overestimation of the true effects of representation learning, necessitating further studies. For OOD data generation methods, they try to use the existing ID data to obtain the data near the boundary of the ID and the data far away from the ID by sampling in low-density regions or distance metrics, and thus regularize the model to better separate the ID and OOD (Lee et al., 2018a; Vernekar et al., 2019; Du et al., 2022; Tao et al., 2023). For outlier exposure methods, they help the model training by introducing additional surrogate OOD data for detection in unseen OOD scenarios. Some methods directly make the model learn from OOD data with low OOD score predictions (Hendrycks et al., 2018; Liu et al., 2020). Some methods studies different sampling strategies and regularization strategies (Van Amersfoort et al., 2020; Li & Vasconcelos, 2020; Chen et al., 2021; Ming et al., 2022b). Compared with other fine-tuning methods, outlier detection shows superior performance, but the quality and difficulty of obtaining surrogate OOD data largely hinders its detection ability in the real world, which is a challenge addressed by our approach in this paper.

Diffusion Models. Diffusion models (Sohl-Dickstein et al., 2015; Ho et al., 2020) have emerged as the new state-of-the-art family of deep generative models, which not only ensure high-fidelity results but also exhibit improved training stability compared to GAN (Goodfellow et al., 2014; Yang et al., 2022). Current research on diffusion models is mostly based on three predominant formulations, *denoising diffusion probabilistic models* (DDPM) (Sohl-Dickstein et al., 2015; Ho et al., 2020; Nichol & Dhariwal, 2021), *score-based generative models* (SGM) (Song & Ermon, 2019; 2020) and *stochastic differential equations* (SDE) (Song et al., 2021; Song & Ermon, 2020). While ensuring high-fidelity generation results, some recent approaches begin to explore high-speed sampling (Song et al., 2020; Lu et al., 2022a;b).

Diffusion models have been widely used in various fields. Specifically, in the field of computer vision, it is used for super-resolution, repainting, image editing, (Meng et al., 2021; Rombach et al., 2022; Saharia et al., 2022; Lugmayr et al., 2022) etc. In the multi-modal domain, diffusion models are applied to text-to-image generation, text-to-audio generation, and text-to-3D generation (Avrahami et al., 2022; Gu et al., 2022; Nichol et al., 2022; Xu et al., 2023; Popov et al., 2021) etc. as a technical support. Moreover, recent works exploit the powerful representational and generative capabilities of diffusion models as data augmentation, e.g. for image classification tasks (Azizi et al., 2023; Burg et al., 2023; Gowal et al., 2021), for medical image analysis (Rahman et al., 2023; Ozbey et al., 2023; Wu et al., 2023). In this paper, we utilize the dm method to generate surrogate OOD data for training the model effectively to accurately distinguish between ID and OOD instances in unseen OOD scenarios.

B Visualization

We contrast a number of different strategies for exploiting synthetic outliers based on diffusion model. And we perform visual analysis of their synthesized outlier results separately.

Figure 4: Visual reconstruction experiment visualization, which presents the generated outliers for CIFAR benchmarks.

B.1 VISUALIZATION OF VISUAL RECONSTRUCTION

By adding Gaussian noise $\mathcal{N}$ to the visual embeddings, and generating denoised data from perturbed visual latents instead of using Gaussian noise as z_0, we can obtain outliers. The results are shown in Figure 4. Specifically, we perform a t-step ($\mathrm{t} = 400$) diffusion process and obtain outliers through image reconstruction while reducing the weight of the text guidance.

According to the visualization results, several intriguing phenomena are observable. The outliers generated by visual reconstruction resemble different image styles within the same category as the ID data, rather than the newly categorized ones that indicate semantic shift.

B.2 VISUALIZATION OF ADDING NOISE TO TEXT CLASSES

By introducing Gaussian noise $\mathcal{N}$ to the embedded text categories $\mathcal{T}(\mathrm{prompt}(\mathrm{y}))$ and generating outliers through the process of text conditional generation, we perturb the text embeddings by adding Gaussian noise $\mathcal{N}$. The results are presented in Figure 5. It can be observed from the generated outliers that the method of adding Gaussian noise to the text embeddings lacks stability. The addition of a small noise disturbance to the partial text embedding leads to the generation of outliers with large semantic deviation during the text-to-image generation process. However, some text embeddings are not sensitive to noise perturbation, and therefore, they are unable to synthesize OOD data through noise perturbation, e.g. **frog** and **horse**. Choosing the appropriate level of noise perturbation for all ID text embeddings is challenging.

B.3 VISUALIZATION OF VISUAL INTERPOLATION

Interpolation using diffusion models has been widely employed in various tasks (Wang & Golland, 2023), e.g. video frame interpolation and customization. In this section, we utilize image interpolation to synthesize outliers. Specifically, we implement linear interpolation (lerp) within the visual latent space $\mathbf{z} = \mathcal{E}(\mathrm{x})$. The results are presented in Figure 6. It can be observed that the quality of the generated outliers decreases when there is a large visual semantic gap between the two interpolated targets.

B.4 VISUALIZATION OF TEXTUAL INTERPOLATION

Different from visual interpolation experiments, text interpolation does not require the addition of noise and can be performed directly between text embeddings. Specifically, we utilize $\beta\mathcal{T}(\mathrm{prompt}(y_i)) + (1-\beta)\mathcal{T}(\mathrm{prompt}(y_j))$ to interpolate between embeddings of different text categories. The results are presented in Figure 7. It can be observed that reliable outliers only occur

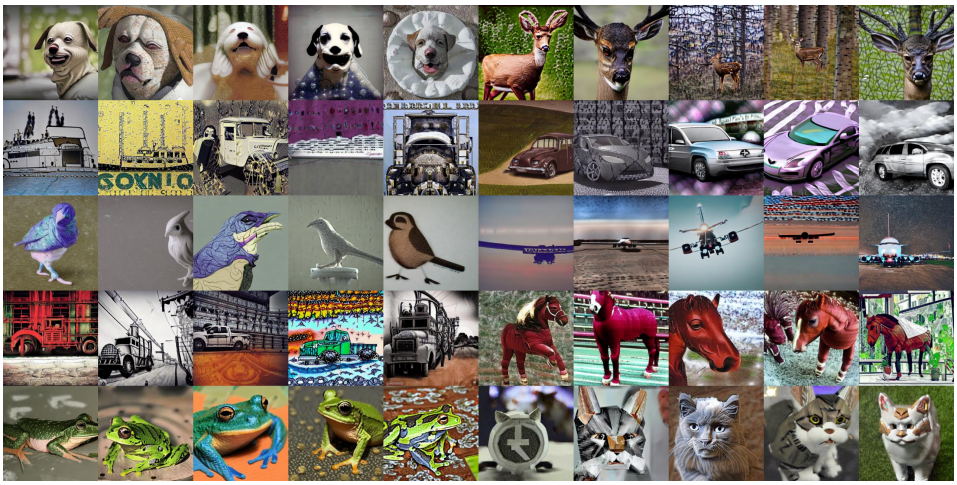


Figure 5: Experiments on adding Gaussian noise to text embeddings, which presents the generated outliers for CIFAR benchmarks. The corresponding ID classes from top left to bottom right are **dog**, **deer**, **ship**, **automobile**, **bird**, **airplane**, **truck**, **horse**, **frog**, and **cat**, respectively.

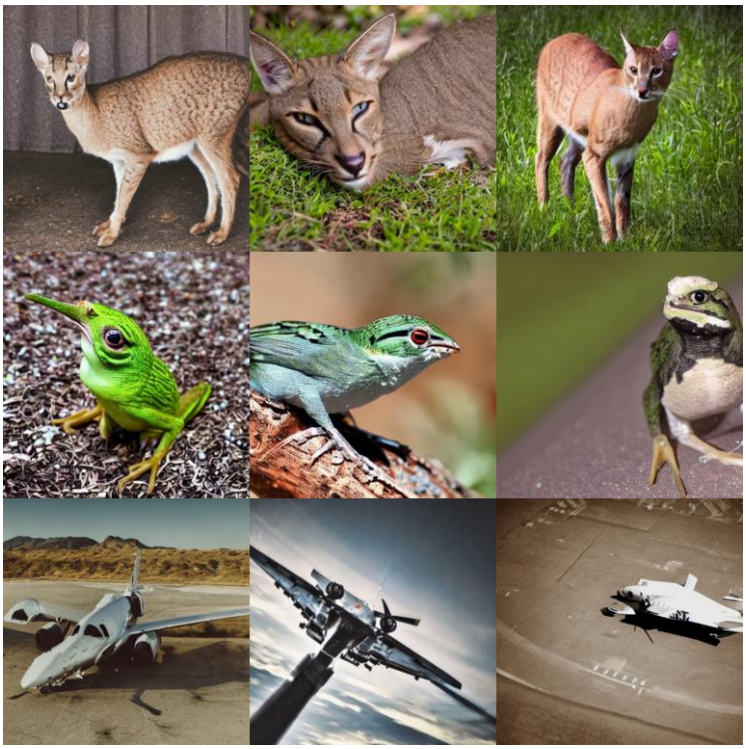


Figure 6: Experiments with visual space interpolation to generate outliers. From top to bottom are, the interpolation results of **cat** and **deer**, **frog** and **bird**, as well as **airplane** and **automobile**, respectively.

around intermediate values of the interpolated weights β. However, this strategy is not effective as a reliable outlier synthesis strategy.

Table 3: OOD detection results for ImageNet benchmark. The baseline with * added represents the representative outlier exposure methods. And the baseline with † added represents the representative outlier generation methods. Bold numbers are superior performances.

Method	Textures		Places365		iNaturalist		SUN		**Average**	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
Using ID data only										
MSP	66.58	80.03	74.15	78.97	72.72	77.19	78.70	75.15	73.04	77.84
Free Energy	52.84	86.36	70.64	81.67	73.98	75.97	76.92	78.08	68.60	80.52
ASH	15.93	96.00	63.08	82.43	52.05	83.67	71.68	77.71	50.68	85.35
Mahalanobis	40.52	91.41	97.10	53.11	96.15	53.62	96.95	52.74	82.68	62.72
KNN	26.54	93.49	78.64	76.82	75.78	69.51	74.30	78.85	63.82	79.66
Using ID data and auxiliary OOD data										
ConfGAN†	68.74	78.74	77.40	77.24	72.67	78.29	80.73	73.88	74.88	77.03
VOS†	94.83	57.69	98.72	38.50	87.75	65.65	70.20	83.62	87.87	61.36
NPOS†	56.10	84.37	78.23	76.91	74.74	77.43	83.09	73.73	73.04	78.11
OE*	57.34	82.97	7.92	98.04	73.87	76.94	52.60	77.31	52.60	83.81
Energy-OE*	42.46	88.27	1.88	99.49	73.81	78.34	69.45	79.54	46.90	86.41
ATOM*	60.20	90.60	7.07	98.25	74.30	77.00	55.87	75.80	49.36	85.41
DOE*	35.11	92.15	0.72	99.79	72.55	78.00	59.06	85.67	41.86	88.90
POEM*	40.80	89.78	**0.26**	**99.70**	73.23	68.83	65.45	82.08	44.93	85.10
DOG	**21.29**	**95.53**	42.73	91.15	**37.30**	**89.68**	**39.11**	**89.67**	**35.11**	**91.51**

B.5 VISUALIZATION OF NEAR-OOD GENERATION OF TEXT

In this section, we conduct an experiment to translate the task of locating near-OOD data into text space. We generate the near-OOD data by selecting near-synonyms of the current category text as anchors on the text side. The results are presented in Figure B.5. Specifically, we choose the top-k ($k = 1000$) synonyms based on the current classes for text-conditional generation. This strategy generates outliers by searching for similar embeddings in the text space in order to find appropriate anchors. However, since visual images contain rich background information, the near-OOD anchors searched by class words in the text space may be offset from the visual space.

C MORE EVALUATIONS

C.1 IMAGENET EVALUATIONS

We also conduct experiments on the ImageNet benchmarks, demonstrating the effectiveness of our DOG when facing this very challenging OOD detection task. Due to the large semantic space and complex image patterns, OOD detection on the ImageNet dataset is a challenging task (Huang & Li, 2021). However, similar to the CIFAR benchmarks, our DOG method also demonstrates the best detection performance among all the baseline methods considered.

C.2 MORE ABLATION EVALUATIONS

Ablation on k in process of selecting topk candidate set. We conducted experiments to explore the effect of the value of the candidate word set k on OOD detection performance. The result is presented in Figure 9.

A new pipeline as a kind of OE provides surrogate OOD data. We regard DOG as a new pipeline for outlier exposure providing the generation of surrogate OOD data and combining with existing outlier exposure methods. We selected the conventional and widely concerned outlier exposure method OE (Hendrycks et al., 2018) and Energy-OE (Liu et al., 2020), as well as the method POEM (Ming et al., 2022b) which implements SOTA on both CIFAR10 and CIFAR100 benchmarks for experiments.

D EXPERIMENTAL ENVIRONMENT

All experiments were conducted using four 3090Ti GPUs.

Table 4: Results of the combination of our DOG and existing OE methods on CIFAR benchmarks.

Method	SVHN		LSUN		iSUN		Textures		Places365		**Average**	
	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑	FPR95 ↓	AUROC ↑
CIFAR-100												
OE	46.73	90.54	16.30	96.98	47.97	88.43	50.39	88.27	54.30	87.11	43.14	90.27
OE + DOG	44.50	85.46	34.66	92.29	5.39	98.58	43.81	91.20	48.59	89.23	**35.39**	**91.35**
Energy-OE	35.34	94.74	16.27	97.25	33.21	93.25	46.13	90.62	50.45	90.04	36.28	93.18
Energy-OE + DOG	24.50	95.07	41.39	91.09	50.16	88.65	18.07	94.93	16.60	96.34	**30.14**	**93.22**
POEM	22.27	96.28	13.66	97.52	42.46	91.97	45.94	90.42	49.50	90.21	34.77	93.28
POEM + DOG	41.85	91.79	35.75	92.75	26.85	92.96	19.80	95.63	23.90	93.52	**29.62**	**93.33**

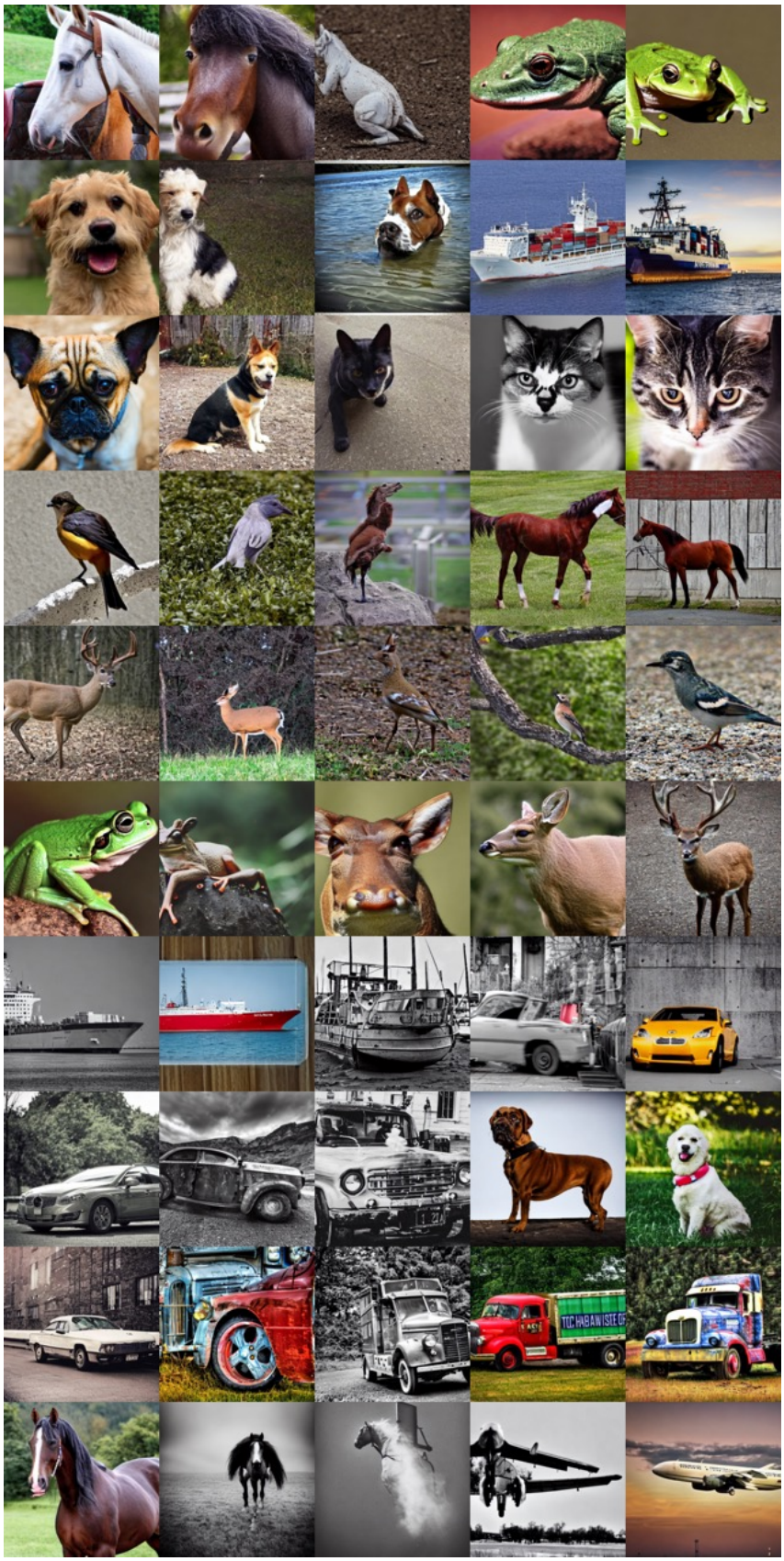
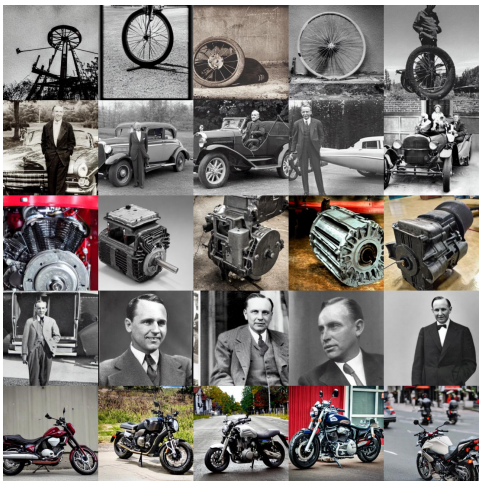


Figure 7: Experiments with text embedding interpolation to generate outliers. From left to right the parameter of interpolation β is $\{0.1, 0.3, ..., 0.9\}$.



(a) Outliers for class **automobile**

(b) Outliers for class **bird**

(c) Outliers for class **frog**

(d) Outliers for class **truck**

Figure 8: Visualization results for partial outliers of the CIFAR benchmarks.

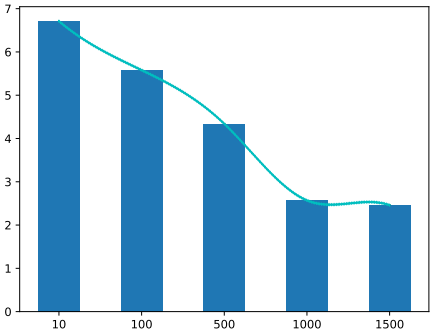


Figure 9: FPR95 values corresponding to different values of parameter k for CIFAR benchmarks.