HYBOOD: a hybrid generative model for out-of-distribution detection with corruption estimation

Giwoong Lee*, Jiseung Ahn*, Jeongyeol Choe

[1]IOPS
Yuseong-gu
Daejeon 35223 Republic of Korea
gwlee0524@i-ops.co.kr, jsahn@i-ops.co.kr, jychoe@i-ops.co.kr

Abstract

We propose **HYBOOD**, a hybrid out-of-distribution model based on normalizing flow followed by a simple linear classification model. In real-world settings, it is known that data corruption has a strong influence on model degradation; for example image quality like noise, blur and image geometry like translation, scaling, rotation. MNIST-C, CIFAR10-C are the general synthesized datasets to measure model robustness and corruption difficulty in terms of covariate and semantic shifts. HYBOOD shows that the separability between in-distribution, covariate shift, and semantic shift can be represented by generative distribution distance and log-scale density $\log(p(\boldsymbol{x}))$. We also find out the types of covariate shifts are ordered by corruption difficulty ranking (CDR) for the datasets. To the best of our knowledge, this is the first method to measure data corruption difficulty with generative models using Wasserstein Distance, Mutual Information and Minimal Description Length. In this paper, we pose interesting experimental results that the generative model tested on MNIST-C is most deteriorated by fog, impulse noise and stripe corruption types. This can be interpreted that those types are challenging corruptions to the generative model in uncertainty and complexity. By training in-distribution data only, HYBOOD achieves out-of-distribution detection performance for distinguishable covariate and semantic shifts, and quantifying covariate shift ranking.

Open-set problems require trustworthy models for up-to-date deep neural networks in that for example as an image classifier in a self-driving car the model can face a new input class (e.g. a passing wild animal on highway) not seen in the training time. Out-of-distribution (OOD) detection is crucial to solve such a scenario under broad taxonomy scope as in (Yang et al. 2024), where anomaly detection, novelty detection (e.g. detecting the new data different from all training data), and outlier detection are categorized by inductive and transductive learning tasks. OOD detection methods are extended from discriminative to generative models since deep generative models are effective unsupervised learning exactly aware of the underlying distribution of the training data via exact marginal likelihood. Unlike the popularity of the deep generative models, recent works have shown that deep generative models can assign higher likelihood to out-of-distribution data than the training data (Nalisnick et al. 2019c; Choi, Jang, and Alemi 2019).

As a causal explanation and solution to the problem, thorough methods of generative deep neural networks are proposed using such as estimating image complexity (which is the degree to distinguish meaningful contents from noise), likelihood ratio, or frequency-regularized learning (Nalisnick et al. 2019a; Serrà et al. 2020; Zhang, Goldstein, and Ranganath 2021; Havtorn et al. 2022; Cai and Li 2022).

As yet the detection of covariate and semantic shifts by deep generative models is less explored. (Bai et al. 2023) proposed a unified learning framework, SCONE, that is capable of simultaneously generalizing to covariate shift while robustly detecting semantic shift by training a mixture of in-distribution (ID), covariate shift and semantic shift as out-of-distribution. The model, however, trains both labeled in-distribution data and unlabeled wild data and detects OODs via energy margin. We had a speculation that generative models may be an alternative to the discriminative OOD detection since the generative models can generalize to cover covariate shifts with augmentation so covariate shift plays a pivotal role in addressing representation between in-distribution and semantic shift as our hypothesis schematically depicted in Figure 2. We also assumed that the shift-aware deep generative model's generalization via covariate shift might be affected by coverage levels due to the shift's complexity. Corruption datasets like MNIST-C and CIFAR10-C provide various corruption types that can be used in the difficulty measure of covariate shifts.

In this paper, we propose **HYBOOD**, a simply modified hybrid model based on the architecture of (Nalisnick et al. 2019b) to estimate corruption difficulty and detect out-of-distribution *shiftness* in Figure 1. To simplify the structure of the model, we replace a generalized linear model (GLM) with a compressed GLM, global average pooling and linear model (GAPLM). GAPLM takes latent features in the generative model and plays similar role as a penultimate layer described in (Zhou 2023; Liu et al. 2022). As explained in (Lin, Chen, and Yan 2014), global average pooling by the mlpconv layers enables better approximation to the confidence maps than GLMs. Normalizing-flow density distribu-

*These authors contributed equally.

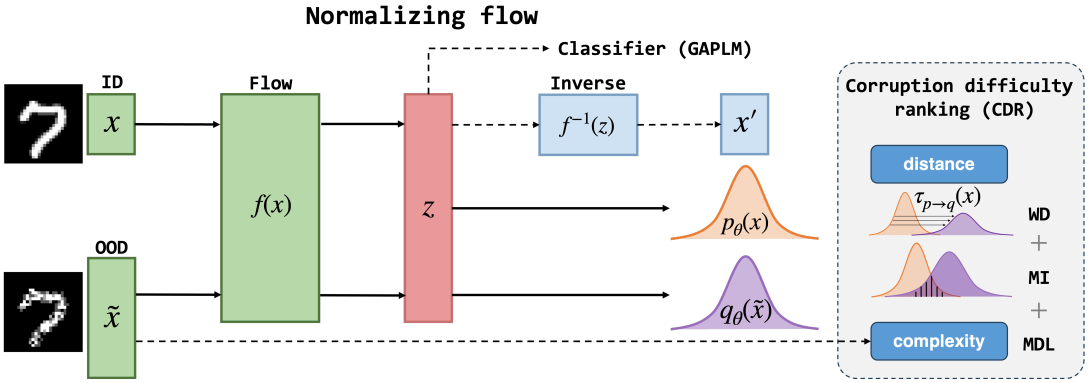


Figure 1: HYBOOD architecture. Based on the DIGLM, global average pooling from the latent space in the normalizing flow is used for selective classification. From the latent information, the model measures the distance between a training and a test sample using Wasserstein Distance (WD), Mutual Information (MI), and $\log(p_\theta(\boldsymbol{x}))$. Independently image complexity is calculated by Minimum Discription Length (MDL) conclusively as a measure of corruption difficulty and *OODness*.

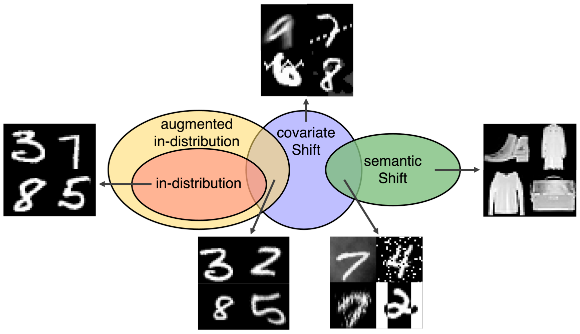


Figure 2: Schematic diagram of IDs (with augmented in-distribution) and OODs (covariate, and semantic shift) with samples for each distribution. A deep generative model trained on MNIST is generalized to augmented distribution, overlapped with covariate shift. Semantic shift is farthest away from in-distribution. However the relative distance may be changed for most difficult covariate shift types expressive as least or no intersection with in-distribution, for example, like the top low of Figure 5.

tion for given data with uncertainty measures could provide the plausible boundary to detect *OODness* how much the data is deviated from the training distribution. For example, the generative distributions trained on MNIST may mediate between in-distribution and semantic shift in the difficulty level of covariate corruption types as assumed in Figure 2. This is the key idea to estimate corruption difficulty and distinguish covariate shift and semantic shift on the hybrid generative model.

Motivated by (Ozair et al. 2019; Hendrycks and Dietterich 2019) that each corruption distribution can be represented by Wasserstein Distance and $\log p(\boldsymbol{x})$ as seen in Figure 3, the density distributions between ID, augmented ID, covariate, and semantic shift trained on MNIST-C were found to be justifiably matching our hypothesis of Figure 2.

Through our proposed method, we observe that the corruption types of MNIST-C and CIFAR10-C are sorted by the uncertainty measures highly consistent with other related works. Also we show that log density distributions are related to the corruption difficulty and accuracy we ranked, and each uncertainty measures tend to capture different corruption types.

Our contributions for hybrid out-of-distribution with corruption estimation are as follows:

1. *Corruption difficulty ranking*: we measure corruption difficulty for standard MNIST-C, CIFAR10-C using distribution metrics (e.g. Wasserstein Distance, Mutual Information) extracted from the latent space and image complexity metric (e.g. Minimal Discription Length) for covariate shift types.
2. *Covariate and semantic shiftness*: HYBOOD quantifies covariate and semantic shifts based on a hybrid deep generative model. Of the interesting points, the log density distributions of ID, covariate, and semantic shift are observed to be located according to corruption difficulty ranking. The more difficult the covariate shift is, the farther from the semantic shift for MNIST-C.
3. *Uncertainty*: we show the corruption difficulty ranking corresponds to aleatoric and epistemic uncertainty. CDR is an agnostic prior of model and data uncertainty based on normalizing flow model and image complexity estimation.

Related works

Out-of-distribution detection

As deep neural networks are used in many safety-critical cases, the reliability and trustworthiness of the decision of the models attract great attention. One of the model robustness techniques is out-of-distribution detection to decide whether a new input sample is in the training distribution or in a new unseen distribution. Out-of-distribution can be simply expressed as $q(y, \boldsymbol{x}) \neq p(y, \boldsymbol{x})$. where p is source (train) distribution and q is target (test) distribution.

Generally OOD detection can be categorized into classification-based, density-based, and distance-based methods (Yang et al. 2024). Breaking down out-of-distribution for detection stages, it is divided into training-time and inference-time OOD detections, in which inference-time methods are simple to adopt to various model architectures. But since the scoring functions are based on the output of a model, it is hard to anticipate to have a safety-aware learning objective. As a solution, OOD detection can suffice through out-of-model-scope detection in order to reject unsafe predictions during inference (Guérin et al. 2023).

Though discriminative model's prediction $p(y|\boldsymbol{x})$ is typically accurate on i.i.d test inputs, it can yield overconfidence in case of out-of-distribution inputs. Thus, density model $p(\boldsymbol{x})$ may be a support to decide when to trust $p(y|\boldsymbol{x})$ (Bishop 1993). (Nalisnick et al. 2019b) proposed a hybrid of generative and discriminative model using normalizing flows (NF) to compute exact density $p(\boldsymbol{x})$ and $p(y|\boldsymbol{x})$ in a single feed-forward pass.

Hybrid models

While normalizing flow models are powerful in many cases, NF in itself does not show as much performance as in out-of-distribution detection because the distance between ID and OOD on input space is relatively close and $p(\boldsymbol{x}|\mathcal{D})$ has no semantic information, which is related to target class information; for example in Colored-MNIST the target classes may be variable such as digits or colors depending on definition.

To overcome the restriction of OOD detection based on NF architecture, (Zhang et al. 2020) combines deep neural networks and NF, where gradients are updated alternatively between the flow model and the classifer. (Cao and Zhang 2022) uses spectral normalizing in which the gradients of the NF and DNN are propagated separatively instead of jointly at each training step. HYBOOD is the variant of the DIGLM (Deep Invertible Generalized Linear Model) of (Nalisnick et al. 2019b). The difference between HYBOOD and DIGLM is that HYBOOD instead applies global average pooling from the latent distribution to classify. The model consists of a compressed GLM stacked on top of an invertible generative model (Kingma and Dhariwal 2018). The exact joint distribution $p(y, \boldsymbol{x})$ can be computed by the neural hybrid model evaluating predictive accuracy and uncertainty altogether. The generative density $p(\boldsymbol{x})$ is used for out-of-distribution detection.

Data complexity and likelihood based generative models in OOD detection

(Serrà et al. 2020) suggests that image complexity affects excessive influence on the likelihood of generative models and solves the problem by using likelihood ratio similar to Bayesian model comparison. The visual complexity and data corruption are related to the likelihood vs. empirical KDE. For example, the SVHN street sign dataset, comparing to CIFAR10, has lower visual complexity resulting in higher likelihood.

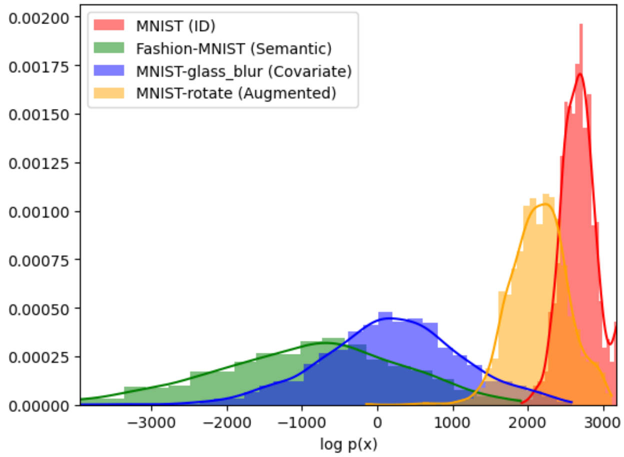


Figure 3: HYBOOD separates ID and OODs (covariate shift, and semantic shift) for MNIST, MNIST-C, Fashion-MNIST. These distributions are placed in a similar manner described in Figure 2 as generative densities. The glass blur corruption is ranked forth in CDR.

Model misestimation can be another problem in generative OOD models (Zhang, Goldstein, and Ranganath 2021). The authors use estimation error as a solution to the failure rather than the misalignment between likelihood-based OOD detection and out-of-distributions.

Covariate and semantic shifts

Suppose a training dataset is from a source distribution $p(\boldsymbol{x},\ y)$, which is used for a predictive model $p(y|\boldsymbol{x})$. If a test dataset has a target distribution $q(\boldsymbol{x},\ y)$ which is different from the source distribution, $p \neq q$, this is called distribution shift or dataset shift (Murphy 2023). **Covariate shift**, also called domain shift, represents that the distribution of features, $p(\boldsymbol{x})$, changes and $p(y|\boldsymbol{x})$ is fixed. It is common that same trees can have different visual images in each season on satellite imagery, and digits of different colors with class 1 are the same label in colored MNIST, and the backgrounds of pictures taken cows may be various grass plains. **Semantic shift** represents a kind of distribution shifts that the source domain is $p(\boldsymbol{x})p(y|\boldsymbol{x})$, whereas target domain is $q(\boldsymbol{x})q(y|\boldsymbol{x})$ which means the image and label are all changed from source to target.

Of distribution shifts, HYBOOD detects covariate and semantic shifts with corruption estimation of the covariate shift using corruption datasets. As out-of-distribution detection has a lot of attention in open-set applications, combined methods to deal with covariate and semantic shifts are proposed (Averly and Chao 2023).

In deep neural networks, test risk can be deduced first by decomposing the covariate shift into $q(\boldsymbol{x}, y) = q(\boldsymbol{x})q(y|\boldsymbol{x})$, and from $q(y|\boldsymbol{x}) = p(y|\boldsymbol{x})$, then

$$\underbrace{\min}_{w} \int dx\, q(\boldsymbol{x}) \int dy\, p(y|\boldsymbol{x}) l(f(\boldsymbol{x}, w), y)$$

where l is a loss function, f is a predictive model and w is weight parameters of f, which is different from the training risk (Huang, Li, and Smola 2021). It is necessary to find the density ratio, $\int d\boldsymbol{x} p(\boldsymbol{x}) \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} f(\boldsymbol{x})$, but the density estimation is difficult due to the tractability. Conditional class probability $r(y = 1|\boldsymbol{x}) = \frac{p(\boldsymbol{x})}{p(\boldsymbol{x})+q(\boldsymbol{x})}$, $\alpha = \frac{q(\boldsymbol{x})}{p(\boldsymbol{x})} = \frac{r(y=-1|\boldsymbol{x})}{r(y=1|\boldsymbol{x})}$ can be used to estimate to decide whether the data have covariate shift which is a corrected training with reweighting loss.

(Balakrishnan et al. 2020) addresses an observational method for face recognition measuring covariate bias like skin, gender, hair length, age and facial hair and shows that there exists a $q(\boldsymbol{x})$ such that $R[q, f] \geq R[p, f] + \sigma$ where $R[p, f] := E_{(x,y)}[l(y, f(x))]$. It implies that covariate-shitfted samples have larger mean and variance of loss than in-distribution ones (Huang, Li, and Smola 2021). (Yang, Zhang, and Russakovsky 2024) shows that covariate shift is more sensitive than semantic shift and presents a clean semantic dataset that minimizes the inferences of covariate shift. It is also known that neural networks do not generalize under covariate shift (e.g. IMAGENET-C dataset) (Hendrycks and Dietterich 2019). In (Benmalek et al. 2022), assuming a dataset where each image has additional attributes of the classes (e.g. "deflated" of "ball") enables to find out more complex semantic shifts. The authors define static shift such as simply a shift in relative frequency of classes between train and test sets.

Corruption dataset

The corruption estimation as a vulnerability (like impulse noise) for the corruption datasets can be helpful in trustworthy applications to protect image scam or fraud such as fake face recognition, and even in satellite imagery (Chen et al. 2021).

To measure model robustness, corruption datasets like IMAGENET-C, MNIST-C are used in computer vision (Hendrycks and Dietterich 2019; Mu and Gilmer 2019). The IMAGENET-C, MNIST-C datasets consist of 19, 16 corruption types. Originated from IMAGENET-C and CIFAR10-C, MNIST-C is tailored to MNIST to measure out-of-distribution model performance.

(Kong et al. 2023) provides 18 corruptions including three categories for out-of-distribution depth estimation: (1) weather and lighting conditions, (2) sensor failures and movement, and (3) data processing anomalies.

Corruption design MNIST-C is a collection of images with the four corruption principles: (1) non-triviality, (2) semantic invariance, (3) realism, and (4) breadth. Corruptions are designed to degrade the accuracy of models. Computer-vision models can be failed by the corruptions, whereas human vision perceives the labels without failure. Since image corruption happens in real-world settings, sensors, environments, and physical factors are included in the dataset by considering redundancy with other corruption datasets.

Method

As shown in Figure 1, in the first stage HYBOOD learns the in-distribution data using GLOW-based normalizing flow (Kingma and Dhariwal 2018), and the out-of-distribution detection stage is preceded to measure corruption difficulty for covariate shifts. We formulate the training model architecture and key corruption metrics in this section.

HYBOOD model

Followed by the DIGLM architecture (Nalisnick et al. 2019b), HYBOOD consists of a GLOW-based generative normalizing flow and a linear model by global average pooling. GLOW is 1x1 convolutions with $\sum_l \sum_d s_{l,d}(\boldsymbol{x}; \phi) + h_l w_l \log |\det W_l|$ for $\log \left|\frac{\partial \boldsymbol{f}_{\boldsymbol{\phi}}}{\partial \boldsymbol{x}_n}\right|$ as in Equation 1 where $s_{l,d}(\boldsymbol{x}; \phi)$ is a scaling operator used in *ActNorm* and *Affine Coupling Layer*, h_l, w_l are size of height and width in layer l and W_l is weight parameters in layer l.

We define a model of the joint distribution $p(\boldsymbol{x}, y)$ with feature and label pair $(\boldsymbol{x}_n, y_n)$ by GAPLM on the output of a normalizing flow:

$$\begin{aligned} p(y_n, \boldsymbol{x}_n; \boldsymbol{\theta}) &= p(y_n|\boldsymbol{x}_n; \boldsymbol{w}, \boldsymbol{\phi})\ \ p(\boldsymbol{x}_n; \boldsymbol{\phi}) \\ &= p(y_n|f(\boldsymbol{x}_n; \boldsymbol{\phi}); \boldsymbol{w})\ \ p_z(f(\boldsymbol{x}_n; \boldsymbol{\phi}))\ \ \left|\frac{\partial \boldsymbol{f}_{\boldsymbol{\phi}}}{\partial \boldsymbol{x}_n}\right| \end{aligned} \tag{1}$$

In GAPLM, $\boldsymbol{z}' = \text{GAP}(\boldsymbol{z})$ from the latent feature $\boldsymbol{z}$, acts as a penultimate layer to the classifier where GAP (Szegedy et al. 2015) means global average pooling.

$$p(y_n|z') = g(z'; w)$$

It is classification probability where g is a classification function with a few MLP layers.

The learnable parameters are $\boldsymbol{\theta} = \{\boldsymbol{\phi}, \boldsymbol{w}\}$ where $\boldsymbol{\phi}$ is parameterized in the predictive and generative components, and $\boldsymbol{w}$ is only in the predictive component.

Corruption estimation with uncertainty

(Hendrycks and Dietterich 2019) scores IMAGENET-C across five corruption severity levels as a classifier's robustness and corruption difficulty. Corruption Error (CE) is a standardized aggregate performance measure that calculates the deviation of error rate from the model trained on a clean dataset. Different from CE, CDR has no additional test stage at each level of severity which directly measures distributional distances from the HYBOOD, and image complexity from the input data in view of model and data uncertainty at once (See Analysis section in more detail).

MI, WD, and MDL The mutual information between two different random variables X and $\hat{X}$ is defined as follows:

$$\begin{aligned}\mathbb{I}(X;\hat{X}) &\triangleq D_{\mathbb{KL}}\left(p(x,\hat{x}), p(x)p(\hat{x})\right)\\ &= \sum_{\hat{x}\in\hat{X}}\sum_{x\in X} p(x,\hat{x})\log\frac{p(x,\hat{x})}{p(x)p(\hat{x})}\end{aligned}$$

which can be intrepreted as how similar the test input is with the training data in terms of image corruption level (training data as a criterion) depicted as the intersection between two distributions in Figure 1.

We apply Wasserstein distance to measure as how much out-of-distribution moves from in-distribution as a transport.

$$W_p(P,Q) = \Big(\inf_{\gamma\in J(P,Q)}\int ||x-\hat{x}||^p d\gamma(X,\hat{X})\Big)^{\frac{1}{p}}$$

where $J(P,Q)$ denotes all joint distributions γ for $(X,\hat{X})$ that have marginals P and Q. Here each random variable represents $\boldsymbol{z}_{train}$ and $\boldsymbol{z}_{test}$. This metric tends to capture the noise types in corruption data (See Analysis section).

As MI is a dependency measure between two random variables, it is zero when two distributions are independent and goes to $H(X)$, entropy of X, when two distributions are identical. In representation learning perspective, (Ozair et al. 2019) provides Wasserstein dependency measure (WDM) as a posterior, a modified version of mutual information where KL divergence is changed to Wassererstein distance. The combination of mutual information and Wasserstein distance in the latent space of the generative model that we propose to capture salient features as a measure of model uncertainty, corruption difficulty and out-of-distribution can be the prior in terms of KL divergence of WDM. This agrees to our motivation and hypothesis to use the metrics in normalizing flow.

$$\underbrace{W_p\big(p(x,\hat{x}), p(x)p(\hat{x})\big)}_{\text{WDM (posterior)}} \quad \xleftarrow[\text{agnostic}]{D_{\mathbb{KL}}(p(x),p(\hat{x}))} \quad \underbrace{W_p\big(p(x), p(\hat{x})\big)}_{\text{prior}}$$

where each distribution stands for $p(\boldsymbol{x}) \sim p_{\boldsymbol{\theta}}(\boldsymbol{z})$ as a model uncertainty from the normalizing flow in Eq. 1 in that $p_{\boldsymbol{\theta}}(\boldsymbol{z}) \sim f(\boldsymbol{x};\boldsymbol{\phi})$ is included both discriminative and generative components in the model.

To measure aleatoric uncertainty with image complexity, we introduce Minimum Discription Length (MDL) score described in (Mahon and Lukasiewicz 2023) that represents how hard an input is to attain to the desired quality. MDL is an implicit inductive bias for complexity measure.

Corruption difficulty ranking (CDR) (Kendall and Gal 2017) introduces data (or aleatoric) uncertainty and model (or epistemic) uncertainty in computer vision. Normalizing flow based uncertainty is also introduced using ensemble (Berry and Meger 2023). We set up a corruption difficulty metric that corruption datasets can be quantified as a combination of intrinsic data complexity and extrinsic unawareness degree from the model.

By unifying the mutual information, Wasserstein distance, Minimum discription length, a *corruption difficulty ranking (CDR)* is proposed. Minimum discription length (MDL) calculates image complexity through hierarchical clustering of patches (Mahon and Lukasiewicz 2023; Dwivedi et al. 2023):

$$\text{CDR} = \alpha\text{WD}_{rank} + \beta\text{MI}_{rank} + \gamma\text{MDL}_{rank}$$

where $\text{WD}_{rank}, \text{MI}_{rank}, \text{MDL}_{rank}$ mean ranking values of WD, MI, MDL resepctively and α,β,γ are weight coefficients. For example, in Table 1, fog is the fifth, fourth and first rankings in WD, MI and MDL respectively. Thus, the score of fog is $5+4+1=10$ that the lowest score matches the most difficult corruption when letting α,β,γ as 1. The most corrupted images correspond to top rankings in CDR that the impulse noise and fog corruption types are, for example in MNIST-C, in the top-ranked group with respect to mutual information (See Table 1).

Experiments & Analysis

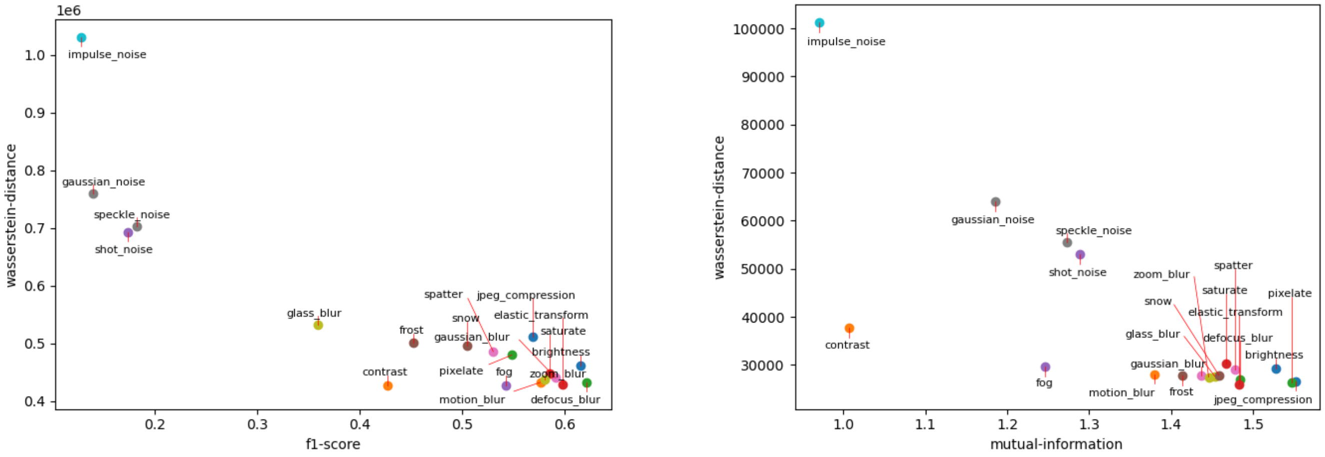
We have experimented two sets of training (ID) data and OOD data which contain (corrupted) covariate shifts and semantic shifts individually: {MNIST, MNIST-C, Fashion-MNIST}, {CIFAR10, CIFAR10-C, SVHN}. In the following subsections we analyze the generative density distributions and CDR as a uncertainty metrics. Finally, the comparison of HYBOOD network architecture is given in the ablation study.

Datasets

- **MNIST** (Deng 2012): Used as training data.
- **MNIST-C** (Mu and Gilmer 2019): Used as covariate shift (corruption) data containing various kinds of corruptions i.e. brightness, canny edges, fog, dotted lines, glass blur, etc.
- **Fashion-MNIST** (Xiao, Rasul, and Vollgraf 2017): Used as semantic shift data.
- **CIFAR10** (Krizhevsky 2009): Used as training data.
- **CIFAR10-C** (Hendrycks and Dietterich 2019): Used as covariate shift (corruption) data containing various kinds of corruptions brightness, contrast, elastic transform, impulse noise, snow, zoom blur, etc.
- **SVHN** (uni 2022): Used as semantic shift data.

Training

We heuristically set the hyperparameters of HYBOOD to optimize the model for OOD's separability and difficulty measure. In a similar way to (Nalisnick et al. 2019b), we set λ, the trade-off weight between $p(y|x)$ and $p(x)$, as $0.75, 0.30$ in MNIST, CIFAR10 respectively. We use Adam optimizer (Kingma and Ba 2017) with a learning rate 0.0001 in both and weight decay $0.001, 0.005$ in MNIST, CIFAR10 respectively.



(a) CIFAR10-C WD vs F1

(b) CIFAR10-C WD vs MI

Figure 4: Accuracy vs. MI and MI vs. WD for CIFAR10-C. (a) plot of Wasserstein Distance and F1-score, and (b) plot of Wasserstein Distance and Mutual Information.

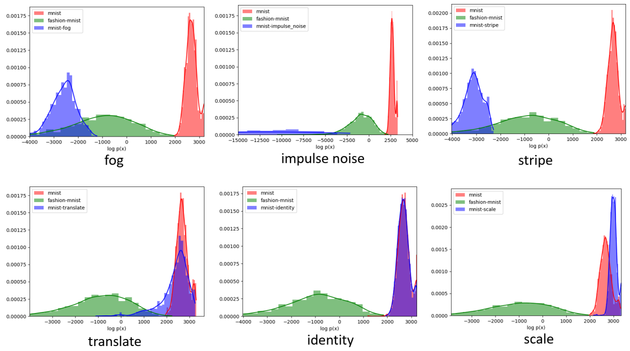


Figure 5: $\log p(\boldsymbol{x})$ plot for MNIST-C. The first row shows the top-3 rankings and the second row shows the bottom-3 rankings from Table 1. The red, blue, green color distributions represent ID, covariate shift, and semantic shift respectively where ID is MNIST and semantic shift is Fashion-MNIST.

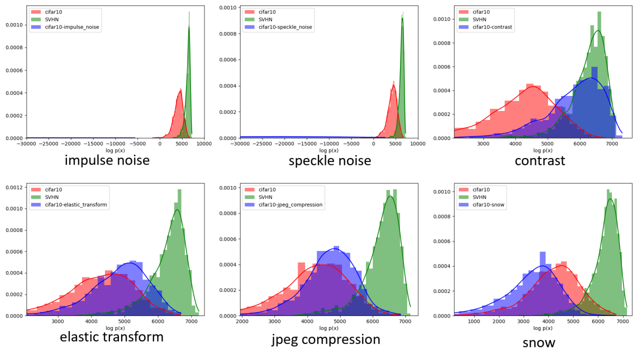


Figure 6: $\log p(\boldsymbol{x})$ plot for CIFAR10-C. The first row shows the top-3 rankings and the second row shows the bottom-3 rankings from Table 2. The red, blue, green color distributions represent ID, covariate shift, and semantic shift respectively where ID is MNIST and semantic shift is SVHN.

Density distribution of covariate, semantic shifts

Figure 5 of $\log p(\boldsymbol{x})$ are ordered by CDR of Tables 1 on MNIST. Very interestingly the in-disribution pushes aside covariate shift and semantic shift from the center for the most uncorrupted type (e.g. scale); in other words, *ID is located relatively in-between for the easiest corruptions.* On the contrary, the covariate shift distributions of hard corruptions are far away from in-distribution like fog, impulse, stripe as seen in the first row of Figure 5. Furthermore, the covariate shifts further away from ID than semantic shift are interpreted as very hard corruptions requiring a lot of generalization for the generative models. From the diagram of Figure 2, this case may be at the highest level of ambiguity that *the semantic shift distribution resides relatively in-between for the hardest corruptions* contrary to the easiest case mentioned above. Also similar as (Nalisnick et al. 2019b), semantic shifts regarded as OODs have a separable aspect on HYBOOD's log density for all corruption types; specifically, distinctly separable for simple MNIST-C and partially separable for more-semantic CIFAR10-C.

Corruption Difficulty Ranking

Tables 1 and 2 show CDR of the corruption data for MNIST and CIFAR10 respectively. Each corruption difficulty metric functions to rank different corruption characteristics that WD tends to rank noise type in highest, MI to rank noise and quality, and MDL to rank quality for CIFAR10-C; WD to rank noise, MI to rank content, and MDL to rank quality for MNIST-C. In overall, CDR is a unified corruption measure to quantify corruption types of image quality, noise, and contents (or blur, noise, digital and weather).

Table 1: Corruption Difficulty Ranking for MNIST-C. The corruption types are orderd by *Corruption Difficulty Ranking*. The arrow directions (↑, ↓) in the parenthesis of the three metrics represent that the more corrupted, the higher for WD, MDL and the lower for MI. Each value in the table describes the score and rank of a corresponding corruption type. *IQ, RN, Con, Geo* stand for the corruption categories of image quality, random noise, content and geometry respectively.

MNIST-C Ranking			
Corruption	WD(↑)	MI(↓)	MDL(↑)
fog(*IQ*)	2.925E+04 (5)	0.31 (4)	17.39 (**1**)
impulse noise(*RN*)	1.034E+05 (**1**)	0.19 (3)	15.07 (9)
stripe(*Con*)	5.861E+04 (2)	0.03 (**1**)	14.78 (10)
glass blur(*IQ*)	1.968E+04 (6)	0.64 (6)	16.88 (2)
brightness(*IQ*)	4.946E+04 (3)	0.16 (2)	14.27 (13)
motion blur(*IQ*)	1.699E+04 (7)	1.10 (8)	16.08 (4)
spatter(*Con*)	3.040E+04 (4)	0.96 (7)	15.61 (8)
zigzag(*Con*)	1.213E+04 (9)	1.34 (9)	15.75 (6)
canny edges(*Con*)	1.622E+04 (8)	0.48 (5)	13.02 (14)
dotted line(*Con*)	1.101E+04 (10)	1.54 (11)	15.61 (7)
shear(*Geo*)	8.766E+03 (13)	1.87 (14)	16.24 (3)
rotate(*Geo*)	8.058E+03 (14)	1.63 (12)	16.01 (5)
shot noise(*RN*)	1.004E+04 (11)	1.49 (10)	12.70 (15)
translate(*Geo*)	8.894E+03 (12)	1.71 (13)	14.30 (12)
identity(*Geo*)	6.673E+03 (16)	2.09 (16)	14.48 (11)
scale(*Geo*)	7.627E+03 (15)	1.88 (15)	12.29 (16)

CDR also reflects model's classification performance for the corruption types. Figure: 4a shows that top-ranking corruptions are in the lowest F1-score group and vice versa. (See the MNIST-C results in supplementary material.)

In comparison to the performance in (Mu and Gilmer 2019), Conv3(GAN) model shows a similar test accuracy for the lowest CDR such as impulse noise of MNIST-C. Additionally comparing to the results by Corruption Error (CE) in (Hendrycks and Dietterich 2019), some interesting points are found. For IMAGENET-C result by AlexNet's CE, the corruption types of the highest scores are impulse noise, shot noise, gaussian noise, while those with the lowest scores are brightness, jpeg, elastic transformation. *For the CIFAR10-C results by HYBOOD's CDR, impulse noise and gaussian noise rank among the highest scores despite the different corruption data, while jpeg and elastic transform rank among the lowest scores.*

Ablation Study

We modified the generalized linear models (GLMs) of the base selective classifier to GAPLM as (Lin, Chen, and Yan 2014) describes that global average pooling by mlpconv layers provides a better confidence map than GLMs. Because of the simplicity of HYBOOD sub-architecture, we compare GAPLM to a high-level flatten linear layer. In Table 3, HYBOOD shows similar F1 by 0.30% on MNIST and a linear model shows better F1 by 2.29% on CIFAR10 in the order of 10^{-4}. As a result, the cost of trainable parameters in the classification stage is decreased from 15,360 to 240 comparing GAPLM to a linear model as global average pooling linear classifier takes averaged latent features generated from normalizing flow. Despite the downscaled latent features, the performance loss is relatively small. Additionally, we compare DIGLM (Nalisnick et al. 2019b) and proposed method in supplementary material.

Table 2: Corruption Difficulty Ranking for CIFAR10-C (in the same way as in Table 1)

CIFAR10-C Ranking			
Corruption	WD(↑)	MI(↓)	MDL(↑)
impulse noise(*RN*)	1.012E+05 (**1**)	0.97 (**1**)	20.13 (7)
speckle noise(*RN*)	5.551E+04 (3)	1.27 (5)	19.94 (10)
contrast(*IQ*)	3.766E+04 (5)	1.01 (2)	19.83 (13)
gaussian noise(*RN*)	6.396E+04 (2)	1.19 (3)	19.66 (15)
motion blur(*IQ*)	2.811E+04 (10)	1.38 (7)	20.30 (4)
zoom blur(*IQ*)	2.750E+04 (15)	1.45 (10)	20.42 (**1**)
fog(*IQ*)	2.973E+04 (7)	1.25 (4)	19.47 (16)
shot noise(*RN*)	5.302E+04 (4)	1.29 (6)	19.41 (17)
gaussian blur(*IQ*)	2.773E+04 (13)	1.44 (9)	20.06 (8)
saturate(*IQ*)	3.022E+04 (6)	1.47 (13)	19.91 (12)
glass blur(*IQ*)	2.772E+04 (14)	1.45 (11)	20.01 (9)
spatter(*Con*)	2.909E+04 (9)	1.48 (14)	19.93 (11)
defocus blur(*IQ*)	2.690E+04 (16)	1.48 (16)	20.37 (3)
pixelate(*IQ*)	2.639E+04 (18)	1.55 (18)	20.38 (2)
brightness(*IQ*)	2.926E+04 (8)	1.53 (17)	19.70 (14)
frost(*IQ*)	2.788E+04 (12)	1.41 (8)	18.75 (19)
elastic transform(*Geo*)	2.588E+04 (19)	1.48 (15)	20.13 (6)
jpeg compression(*IQ*)	2.660E+04 (17)	1.55 (19)	20.24 (5)
snow(*Con*)	2.789E+04 (11)	1.46 (12)	19.13 (18)

Table 3: Performance comparison of HYBOOD trained on MNIST and CIFAR10. Linear HYBOOD shows better classification performance, whereas GAPLM HYBOOD shows cost-effective performance.

HYBOOD Performance				
	MNIST		CIFAR10	
	GAPLM	Linear	GAPLM	Linear
F1	0.9740	**0.9769**	0.6778	**0.6933**
Precision	0.9751	0.9771	0.6775	0.6931
Recall	0.9738	0.9769	0.6782	0.6937

Conclusion

Using a simple and unified hybrid normalizing flow architecture, HYBOOD estimates covariate corruption and out-of-distribution detection with generative density. As we have shown in the results of corruption difficulty ranking, impulse noise is a paramount corruption type in computer-vision models. Robust deep generative models against real-world sensor defects or intentionally corrupted data would have impact on trustworthy AI research.

References

Averly, R.; and Chao, W.-L. 2023. Unified Out-Of-Distribution Detection: A Model-Specific Perspective. arXiv:2304.06813.

Bai, H.; Canal, G.; Du, X.; Kwon, J.; Nowak, R.; and Li, Y. 2023. Feed Two Birds with One Scone: Exploiting Wild Data for Both Out-of-Distribution Generalization and Detection. arXiv:2306.09158.

Balakrishnan, G.; Xiong, Y.; Xia, W.; and Perona, P. 2020. Towards causal benchmarking of bias in face analysis algorithms. arXiv:2007.06570.

Benmalek, R. Y.; Chhabria, S.; Pinheiro, P. O.; Cardie, C.; and Belongie, S. 2022. Learning to Adapt to Semantic Shift.

Berry, L.; and Meger, D. 2023. Normalizing Flow Ensembles for Rich Aleatoric and Epistemic Uncertainty Modeling. arXiv:2302.01312.

Bishop, C. M. 1993. Novelty Detection and Neural Network Validation. In Gielen, S.; and Kappen, B., eds., *ICANN '93*, 789–794. London: Springer London. ISBN 978-1-4471-2063-6.

Cai, M.; and Li, Y. 2022. Out-of-distribution Detection via Frequency-regularized Generative Models. arXiv:2208.09083.

Cao, S.; and Zhang, Z. 2022. Deep Hybrid Models for Out-of-Distribution Detection. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 4723–4733.

Chen, H.-S.; Zhang, K.; Hu, S.; You, S.; and Kuo, C. C. J. 2021. Geo-DefakeHop: High-Performance Geographic Fake Image Detection. arXiv:2110.09795.

Choi, H.; Jang, E.; and Alemi, A. A. 2019. WAIC, but Why? Generative Ensembles for Robust Anomaly Detection. arXiv:1810.01392.

Deng, L. 2012. The mnist database of handwritten digit images for machine learning research. *IEEE Signal Processing Magazine*, 29(6): 141–142.

Dwivedi, R.; Singh, C.; Yu, B.; and Wainwright, M. J. 2023. Revisiting minimum description length complexity in overparameterized models. arXiv:2006.10189.

Guérin, J.; Delmas, K.; Ferreira, R. S.; and Guiochet, J. 2023. Out-Of-Distribution Detection Is Not All You Need. arXiv:2211.16158.

Havtorn, J. D.; Frellsen, J.; Hauberg, S.; and Maaløe, L. 2022. Hierarchical VAEs Know What They Don't Know. arXiv:2102.08248.

Hendrycks, D.; and Dietterich, T. 2019. Benchmarking Neural Network Robustness to Common Corruptions and Perturbations. arXiv:1903.12261.

Huang, Q.; Li, M.; and Smola, A. 2021. Practical Machine Learning (CS 329P).

Kendall, A.; and Gal, Y. 2017. What Uncertainties Do We Need in Bayesian Deep Learning for Computer Vision? In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Kingma, D. P.; and Ba, J. 2017. Adam: A Method for Stochastic Optimization. arXiv:1412.6980.

Kingma, D. P.; and Dhariwal, P. 2018. Glow: Generative Flow with Invertible 1x1 Convolutions. arXiv:1807.03039.

Kong, L.; Xie, S.; Hu, H.; Ng, L. X.; Cottereau, B. R.; and Ooi, W. T. 2023. RoboDepth: Robust Out-of-Distribution Depth Estimation under Corruptions. arXiv:2310.15171.

Krizhevsky, A. 2009. Learning multiple layers of features from tiny images. Technical report.

Lin, M.; Chen, Q.; and Yan, S. 2014. Network In Network. arXiv:1312.4400.

Liu, J. Z.; Padhy, S.; Ren, J.; Lin, Z.; Wen, Y.; Jerfel, G.; Nado, Z.; Snoek, J.; Tran, D.; and Lakshminarayanan, B. 2022. A Simple Approach to Improve Single-Model Deep Uncertainty via Distance-Awareness. arXiv:2205.00403.

Mahon, L.; and Lukasiewicz, T. 2023. Minimum Description Length Clustering to Measure Meaningful Image Complexity. arXiv:2306.14937.

Mu, N.; and Gilmer, J. 2019. MNIST-C: A Robustness Benchmark for Computer Vision. arXiv:1906.02337.

Murphy, K. P. 2023. *Probabilistic Machine Learning: Advanced Topics*. MIT Press.

Nalisnick, E.; Matsukawa, A.; Teh, Y. W.; Gorur, D.; and Lakshminarayanan, B. 2019a. Do Deep Generative Models Know What They Don't Know? arXiv:1810.09136.

Nalisnick, E.; Matsukawa, A.; Teh, Y. W.; Gorur, D.; and Lakshminarayanan, B. 2019b. Hybrid Models with Deep and Invertible Features. arXiv:1902.02767.

Nalisnick, E.; Matsukawa, A.; Teh, Y. W.; and Lakshminarayanan, B. 2019c. Detecting Out-of-Distribution Inputs to Deep Generative Models Using Typicality. arXiv:1906.02994.

Ozair, S.; Lynch, C.; Bengio, Y.; van den Oord, A.; Levine, S.; and Sermanet, P. 2019. Wasserstein Dependency Measure for Representation Learning. arXiv:1903.11780.

Serrà, J.; Álvarez, D.; Gómez, V.; Slizovskaia, O.; Núñez, J. F.; and Luque, J. 2020. Input complexity and out-of-distribution detection with likelihood-based generative models. arXiv:1909.11480.

Szegedy, C.; Liu, W.; Jia, Y.; Sermanet, P.; Reed, S.; Anguelov, D.; Erhan, D.; Vanhoucke, V.; and Rabinovich, A. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, 1–9. IEEE.

uni, P. 2022. SVHN Dataset. https://universe.roboflow.com/peking-uni/svhn-rktm0. Visited on 2024-07-25.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. .

Yang, J.; Zhou, K.; Li, Y.; and Liu, Z. 2024. Generalized Out-of-Distribution Detection: A Survey. arXiv:2110.11334.

Yang, W.; Zhang, B.; and Russakovsky, O. 2024. ImageNet-OOD: Deciphering Modern Out-of-Distribution Detection Algorithms. arXiv:2310.01755.

Zhang, H.; Li, A.; Guo, J.; and Guo, Y. 2020. Hybrid Models for Open Set Recognition. arXiv:2003.12506.

Zhang, L. H.; Goldstein, M.; and Ranganath, R. 2021. Understanding Failures in Out-of-Distribution Detection with Deep Generative Models. arXiv:2107.06908.

Zhou, Y. 2023. Rethinking Reconstruction Autoencoder-Based Out-of-Distribution Detection. arXiv:2203.02194.