Out-of-distribution Detection and Generation using Soft Brownian Offset Sampling and Autoencoders

Felix Möller, Diego Botache, Denis Huseljic, Florian Heidecker, Maarten Bieshaar, and Bernhard Sick
Intelligent Embedded Systems, University of Kassel, Germany
{fm | diego.botache | dhuseljic | florian.heidecker | mbieshaar | bsick}@uni-kassel.de
www.ies.uni-kassel.de

Abstract

Deep neural networks often suffer from overconfidence which can be partly remedied by improved out-of-distribution detection. For this purpose, we propose a novel approach that allows for the generation of out-of-distribution datasets based on a given in-distribution dataset. This new dataset can then be used to improve out-of-distribution detection for the given dataset and machine learning task at hand. The samples in this dataset are with respect to the feature space close to the in-distribution dataset and therefore realistic and plausible. Hence, this dataset can also be used to safeguard neural networks, i.e., to validate the generalization performance. Our approach first generates suitable representations of an in-distribution dataset using an autoencoder and then transforms them using our novel proposed Soft Brownian Offset method. After transformation, the decoder part of the autoencoder allows for the generation of these implicit out-of-distribution samples. This newly generated dataset then allows for mixing with other datasets and thus improved training of an out-of-distribution classifier, increasing its performance. Experimentally, we show that our approach is promising for time series using synthetic data. Using our new method, we also show in a quantitative case study that we can improve the out-of-distribution detection for the MNIST dataset. Finally, we provide another case study on the synthetic generation of out-of-distribution trajectories, which can be used to validate trajectory prediction algorithms for automated driving.

1. Introduction

Artificial intelligence (AI) is the key technology for perception in automated driving. In particular, deep neural networks (DNN) are widely used in relevant tasks such as object detection [37, 19] or trajectory prediction [4, 3]. Safeguarding neural networks in terms of Safe AI [22] is of tremendous importance for safe automated driving. Neural networks perform well when the distribution of the training and test data are sufficiently similar. However, if they are too dissimilar, they can suffer from overconfidence [26, 1, 17] which may even result in fatal events [42].

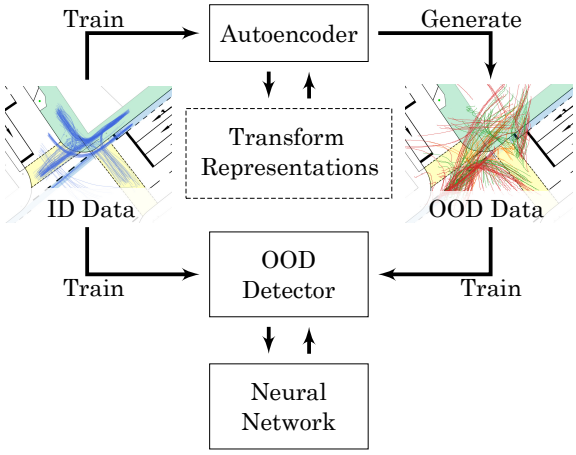

Figure 1. Schematic overview of our approach to create OOD samples used for training OOD detectors or to directly validate an AI model. First, ID samples are used to train an Autoencoder (AE). Afterwards, we generate OOD samples using Soft Brownian Offset sampling on the latent representation of the AE. The AE again decodes the OOD samples to form OOD samples in the original space. Finally, the novel OOD samples are used in conjunction with the original ID samples to train an OOD detector or to validate/ safeguard the AI model directly, i.e., neural model.

A central role in safeguarding AI-function for perception in automated driving is being taken by the so-called corner cases, which are rare but mostly highly critical and therefore relevant cases [19, 18]. The detection of such corner cases in machine learning (ML)

is often referred to as out-of-distribution (OOD) samples [43, 36, 28] and is an essential building block for safeguarding AI-methods. The AI methods can be validated and improved using these OOD samples. A complementary approach is the systematic artificial generation of corner cases and OOD samples, e.g., novel objects, critical (traffic) scenarios, or unusual trajectories of road users.

This article presents a novel data-driven method for the artificial generation of OOD samples, which can be used to safeguard AI models and train improved AI-based OOD detectors. At its core, the method comprises a new algorithm, referred to as Soft Brownian Offset sampling, to create OOD samples at the tails of the data distribution used for training the AI model. Hence, the samples are OOD but still close to the actual in-distribution (ID) samples and therefore likely to be realistic and close to the AI model's intended operation domain.

Gaussian Hyperspheric Offset is a common baseline to create OOD samples by sampling along a normally distributed hypersphere around the ID-data. Soft Brownian Offset Sampling is an algorithm based on the former method that allows to take an arbitrary point from a dataset and translate it to have a likely minimum distance to all other points from the dataset. The algorithm is no longer limited to sampling from a hypersphere around the ID data and is applied to the representations of an autoencoder (AE) trained to encode samples from a particular dataset. A schematic of the proposed approach is given in Fig. 1. Using the AE's decoder's output for the transformed representations allows for generating additional datasets that are implicitly OOD. These datasets can then be used to improve an OOD detector's training or validate the AI model's functionality.

1.1. Main Contributions

The main contributions of this article can be summarized as follows:

- We propose a novel algorithm, called Soft Browning Offset sampling, to create synthetic OOD samples (e.g., from the latent representation of an auto-encoder). We prove the applicability of the algorithm to different data modalities that are important for the perception of highly automated vehicles, i.e., images, time series, and trajectories.
- In a case study with cyclists trajectories, we show that the novel OOD generation method creates unusual but still realistic trajectories to be used for validation of AI-based behavior prediction methods (cf. [44])
- We show that using synthetically generated OOD samples originating from the new algorithm improves the performance of state-of-the-art OOD Detectors.
- We include an easy-to-use Python implementation of the proposed OOD generation methods.[1]

The remainder of this article is structured as follows: In Section 2, we review the current state of the art in the field of OOD-generation and detection. In Section 3, we detail the fundamentals of our approach and present our novel OOD sampling algorithm. Subsequently, we present experimental results in Section 4 and in conclude our findings in Section 5.

2. Related Work

2.1. Out-of-distribution Detection

In general, OOD detection can be described as the task of distinguishing between data that stems from one distribution and data that stems from another distribution that is sufficiently different. It can be formulated as a learning task in which we aim to separate samples originating of the in-distribution (ID) from those of the out-distribution (OOD) [7, 2].

A neural-network-based OOD detection baseline is presented by Hendrycks and Gimpel [20]. It does not require any retraining and utilizes the probabilities from softmax distribution to distinguish between in- and out-of-distribution samples. Another improved OOD detector is the ODIN detection, which uses temperature scaling on the trained network and small perturbations on inputs to separate ID and OOD samples based on the network's softmax score [30]. Chen et al. propose the ALOE algorithm that improves the robustness of state-of-the-art OOD detectors[7] by a novel robust training procedure incorporating both adversarially crafted ID and OOD samples. Density-based approaches aim to build a probabilistic model of the data and then subsequently use this model for OOD detection, e.g., [36]. OOD detection is strongly related to tasks such as anomaly, or novelty detection [16] in which the goal is to detect unknown and potentially anomalous patterns. Outlier Exposure [21] is a novel state-of-the-art deep anomaly detection, which uses a modified loss function to incorporate samples of an auxiliary dataset to better detect OOD samples. [28] propose a modification of loss functions and a novel training method to distinguish between OOD and ID data. Prior Networks [31, 32] use the aleatoric and epistemic uncertainty for OOD detection using a Dirichlet distribution. Another related method is proposed by

[1] https://pypi.org/project/sbo/

Huseljic et al. [23]. They utilize the properties of a Dirichlet-Categorical distribution and are able to measure and separate aleatoric and epistemic uncertainty. This is achieved by combining two objective functions – the first optimizing on ID samples, the second on OOD samples – into one by means of a convex combination. Furthermore, they suggest a naive technique to generate OOD samples by means of adapting the latent space of a generative adversarial network (GAN) [15]. Their model is then able to detect OOD samples while also allowing for a reliable estimation of the risk coming with a decision of the trained DNN. Both [23] and [28] require a separate set of (artificially generated) OOD samples to train OOD detectors.

2.2. Out-of-distribution Generation

In contrast to OOD detection, OOD generation is a relatively new field of research. Recently, the generation of training samples through deep generative models, e.g., GAN [15] or variational autoencoder (VAE) [25], attracted attention in the OOD community. Lee et al. [29] noticed that samples generated at the tails of data distributions can be exploited to improve OOD detection. They use these samples to fine-tune the output of a DNN. Moreover, the authors propose a new GAN objective, which allows the generation of samples in low-density regions of training distributions. As shown by Vernekar et al. [43], this generation procedure requires a DNN with already well-working estimation of predictive uncertainty. Furthermore, [43] show that their approach fails on a simple 3D-example indicating even more significant difficulties in higher dimensions. In contrast, Vernekar et al. utilize a conditional VAE [43] and define two types of OOD samples surrounding the latent encoding on a learned manifold. However, their approach is limited by the dimensionality of input images due to insufficient generative capabilities of VAE and the need for a Jacobian matrix defined over the entire data set leading to high computational cost. An alternative GAN-based approach for OOD sample generation is proposed by Sricharan and Srivastava and show that for effective OOD detection, the generated OOD samples should cover and be close to the low-density boundary of in-distribution [41]. Catching up the idea of these approaches, we propose multiple novel geometric transformations that alleviate generative models to generate OOD samples that are similar to but also sufficiently different from the ID.

3. Methodology

This section introduces concepts and strategies that serve as a basis for the proposed approaches. The basis of our approach is a compact, relatively low-dimensional but yet, expressive representation of the data. Based on this representation, we present three methods to generate OOD samples.

3.1. Feature Representation using Autoencoders

Many natural data sources show the property of presenting a low-dimensional, possibly noisy, manifold embedded within the higher dimensional observed data space [5]. Approaches such as the principal component analysis or autoencoders try to capture this property. We use this low-dimensional embedding as the basis for our generative modeling and the OOD sampling.

Autoencoders, originally proposed in [38], implement a dimensionality reduction with bottleneck layer (i.e., layer with significantly fewer neurons than in the input space). After training, AEs can be separated into two parts: the layers up to the bottleneck can be used as an encoder, and the remaining layers are used as a decoder. The loss function of vanilla AE measures the reconstruction error between the input and decoded sample. The activations of the bottleneck layer's neurons (also referred to as latent variables) comprise a new compact feature (latent) space. They are the starting point of our OOD sampling strategy, i.e., we generate OOD samples in the latent space using the algorithms proposed in the following and subsequently use the decoder to transform the generated data back into the original feature space. There are many extensions to the original autoencoder, e.g., regularized, sparse, denoising, contractive, and variational autoencoder (VAE) [14], which can all be combined with our OOD sampling strategies. Especially, the usage of VAE, which posses a relatively "smooth" latent space, is attractive as a basis for OOD sampling. Moreover, as it offers the direct capability to generate new samples, it provides a decent baseline for OOD sampling (cf. Gaussian Hyperspheric Offset Sampling).

At this point, we would like to emphasize that the approaches presented in this article are not necessarily dependent on a learned representation, e.g., from autoencoders, but can deal with generic low-dimensional feature representations as well.

3.2. Gaussian Hyperspheric Offset

The proposed method of Gaussian Hyperspheric Offset (GHO) projects a point $\boldsymbol{x} \in \mathbb{R}^D$ onto the surface of a hypersphere with the most likely radius of μ. This radius is normally distributed by $\boldsymbol{n} \sim \mathcal{N}(0, \boldsymbol{I}_D)$ and its standard deviation scales with σ. Uniform sampling on the surface of the hypersphere is achieved by scaling the normally distributed vector $\boldsymbol{s} \sim \mathcal{N}(0, \boldsymbol{I}_D)$ by the inverse of its length [39]:

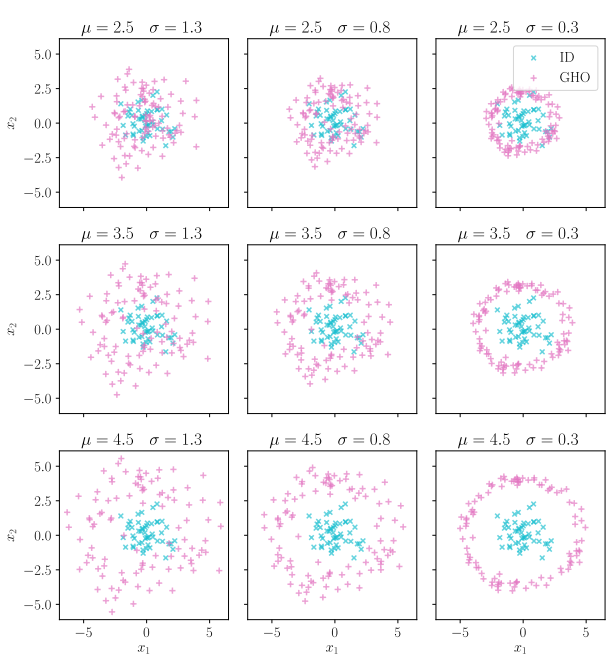

Figure 2. Exemplary choice of parameters for GHO: ID dataset was sampled from $\boldsymbol{X} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}_2)$ while OOD method's parameters are annotated. Varying both μ and σ allows for precise control of OOD samples' behaviour.

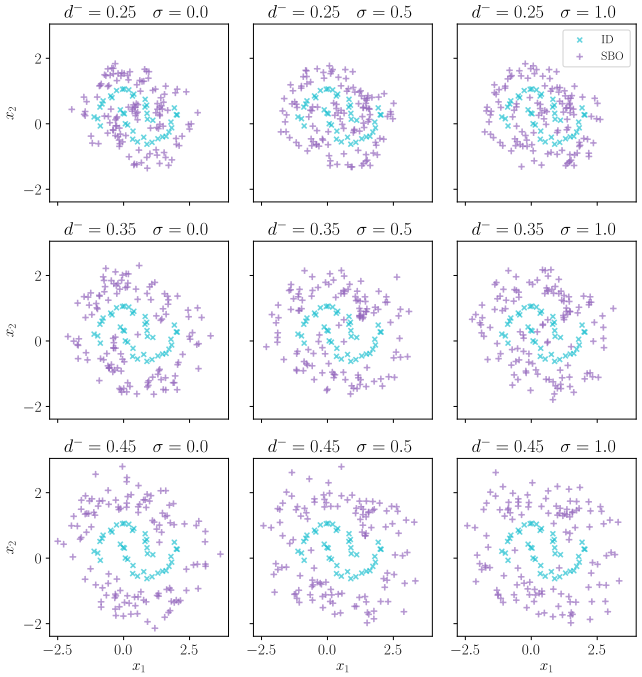

Figure 3. Exemplary choice of parameters for SBO: ID dataset was sampled using Scikit-learn's function *make_moons* [35] while SBO's parameters are annotated with varying d^- and σ while it holds that $d^+ = d^-$. The bottom row makes the difference of the softness parameter visible where it regulates whether OOD samples creep in on pockets ($\sigma > 0$) or not ($\sigma = 0$). Data resulting from application of GHO would not account for the pockets between the two shapes – whereas SBO's advantage is most prominently noticeable in the first row – but would instead create a circular shape as seen in Fig. 2. It is noteworthy that density transfers from ID to OOD originating in its uniform sampling of $\mathbf{y}$ (cf. Alg. 1).

$$\gamma(\mu, \sigma) = \mu \boldsymbol{s} |\boldsymbol{s}|^{-1} + \sigma \boldsymbol{n} \tag{1}$$

Fig. 2 shows the influence of the change of parameters of γ in an exemplary fashion for $\mathbb{R}^2$. One shortcoming of this method is its assumption to work on normally distributed data. While this may hold for specific applications our goal was to weaken these assumptions and propose a more general framework.

3.3. Soft Brownian Offset

Soft Brownian Offset (SBO) defines an iterative approach to translate a point $\boldsymbol{x} \in \boldsymbol{X}$ by a most likely distance d^- from a given dataset $\boldsymbol{X}$ (cf. Fig. 3). It is inspired by Brownian motion (BM) [12] and transfers the concept from its one-dimensional origin to $\mathbb{R}^D$ using hyperspheres as a topological basis. It shares a loose connection with Gaussian processes, since BM has properties of one within the Wiener process [34] which is also confirmed by Donsker's theorem, a functional extension to the central limit theorem [10, 11].

As shown in Algorithm 1 the approach first uniformly selects a sample from the original dataset and then transforms iteratively until the transformed data point's minimum distance to the dataset $d^* = d_{\min}(\boldsymbol{y}, \boldsymbol{X})$ transgresses the boundary distance of d^-. Allowing for soft boundaries, the rejection likelihood $\rho \in [0, 1]$ decides on whether to stop early. It is defined as

$$\rho(d^*, d^-, \sigma) = \left(1 + \exp\left(\frac{d^* + d^-}{\sigma d^- \kappa}\right)\right)^{-1} \tag{2}$$

and is falling sigmoidically with an increasing minimum distance d^* and most likely distance d^-. To accomodate for a likelihood ρ sufficiently close to 1 for $d^* = 0$, $\kappa = 7$ is a proposed common choice while $\sigma \in [0, 1]$ defines the boundary softness.

3.4. Hard Brownian Offset

Hard Brownian Offset (HBO) can be recovered as a special case from Algorithm 1 using $\sigma = 0$. It then converts the most likely distance d^- to a guaranteed minimum distance through neglection of probabilistic break statements given in regular SBO. Disabling boundary

Algorithm 1: Brownian Offset

Data: ID samples $\boldsymbol{X}$ with individual samples $\boldsymbol{x} \in \mathbb{R}^D$ and $|\boldsymbol{X}| = N$, Most likely distance d^-, Offset distance d^+, Boundary softness σ
Result: OOD samples $\boldsymbol{Y}$ with individual samples $\boldsymbol{y} \in \mathbb{R}^D$ and $|\boldsymbol{Y}| = M$

```
for i ∈ {1, ..., M} do
    y ← Uniform(X);
    d* ← 0;
    while d* < d⁻ do
        y ← y + d⁺γ(1, 1);
        d* ← d_min(y, X);
        u ← Uniform([0, 1]);
        if u < ρ(d*, d⁻, σ) then
            break
        end
    end
    Y_i ← y;
end
return Y
```

softness leads to a strict demarcation which can be a desired feature depending on the application.

4. Experiments

Our experiments are three-fold: First, we demonstrate the approaches' capabilities to sample OOD time series. Second, we present a quantitative evaluation regarding the potential improvement for OOD detection on image data. We conclude our experimental evaluation with a case study on the synthetic generation of OOD samples for cyclist trajectories.

4.1. Time series

For time series, a synthetic ID baseline dataset of sine waves is created, where a single wave is given by

$$\mathrm{gt}(t) = \sin(2\pi \cdot f \cdot t) + \epsilon \tag{3}$$

with frequency $f \sim \mathcal{N}(0, 35)$, noise $\epsilon \sim \mathcal{N}(0, 1\mathrm{e}{-1})$, and time $t \in [0, \ldots, 125]$. The ID training dataset comprises 2000 time series sampled from this model (cf. Fig. 4) and is denoted as $\boldsymbol{X}_{id}$. All labels are set to $\boldsymbol{y}_{id} = 0$ (i.e., ID samples). Not only does this have the advantage of being hypothetically reducible to a single dimension in $\boldsymbol{Z}_{id}$ (cf. the spectral decomposition of the input signal) but also of knowing the underlying distribution for $\boldsymbol{Z}_{id}$ and inductively $\boldsymbol{X}_{id}$, which is a desired property to form a comparable OOD baseline.

Having created an ID dataset, $\boldsymbol{X}_{id}$ is then used to train a VAE matching the identity while minimizing reconstruction loss. The actual architectures for the VAE's encoder and decoder are chosen to be

$$\text{Encoder} \equiv L_{64} \times L_{48} \times L_{32} \times L_{20}, \tag{4}$$
$$\text{Decoder} \equiv L_{20} \times L_{32} \times L_{48} \times L_{120}, \tag{5}$$

where L indicates linear layers with number of neurons denoted in the index. We use rectified linear units as activation functions.

Having fully trained the VAE, the encoder is used to transform $\boldsymbol{X}_{id}$ to generate its learned latent representation $\hat{\boldsymbol{Z}}^{\mathrm{vae}}$. These serve as the basis for the application of the proposed approaches of GHO, HBO and SBO. In particular, we use these methods to generate OOD samples in the latent representation denoted as $\hat{\boldsymbol{Z}}^{\mathrm{vae}}_{\mathrm{GHO}}$, $\hat{\boldsymbol{Z}}^{\mathrm{vae}}_{\mathrm{HBO}}$ and $\hat{\boldsymbol{Z}}^{\mathrm{vae}}_{\mathrm{SBO}}$.

Having created these OOD samples, we use the VAE's decoder to reconstruct OOD samples in the input space. These are referred to $\boldsymbol{X}_{\mathrm{GHO}}$, $\boldsymbol{X}_{\mathrm{HBO}}$ and $\boldsymbol{X}_{\mathrm{SBO}}$, respectively. All of these datasets have labels of $\boldsymbol{y}_{ood} = 1$, i.e., they contain only OOD samples.

Table 1. Hyperparameter settings of the OOD sampling methods.

Method	Parameterization
GHO	$\mu = 9$ and $\sigma = 0.8$
HBO	$d^* = d^+ = 2$ and $\sigma_{\mathrm{HBO}} = 0$
SBO	$d^* = d^+ = 2$ and $\sigma_{\mathrm{SBO}} = 1$

The hyperparameter settings are depicted in Tab. 1. We generate 2000 samples using each method. Figure 4 shows a selection of produced results.

To further validate our OOD sampling methods, we construct an "optimal" OOD baseline strategy using the knowledge about the data generating process, i.e., spectral composition underlying the generation of sine curves. The creation of this synthetic OOD baseline dataset $\boldsymbol{X}_{\mathrm{O3D}}$ follows a process similar to $\boldsymbol{X}_{\mathrm{ID}}$, with the difference that the we only sample from the tails of the data generating distribution.

Next, we train an OOD detector in a supervised fashion. The baseline dataset for training an OOD detector is given by the union of $\boldsymbol{X}_{\mathrm{ID}}$ and $\boldsymbol{X}_{\mathrm{O3D}}$. We consider the OOD samples created by the baseline as "ground truth". We use our different datasets for training the OOD detector, i.e., the artificially created OOD sets $\boldsymbol{X}_{\mathrm{GHO}}$, $\boldsymbol{X}_{\mathrm{HBO}}$ and $\boldsymbol{X}_{\mathrm{SBO}}$ and compare it against the baseline OOD set $\boldsymbol{X}_{\mathrm{O3D}}$. These datasets are combined with the $\boldsymbol{X}_{\mathrm{ID}}$ to obtain the respective dataset for training the OOD detector. Training and test split ratio was chosen to be 9:1. Table 2 shows how the selection of datasets influences model metrics in this setting by comparing training data sets with their respectively

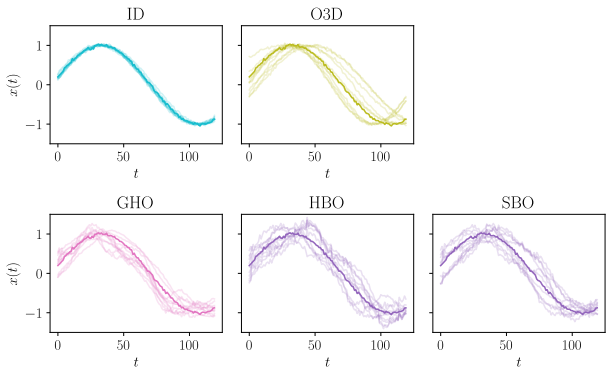

Figure 4. Archetype class of ID dataset (bold) and selected samples closest to it from selected dataset (transparent) according to DTW distance. O3D's inavailability in real settings stemming from its synthetic nature has to be emphasized.

trained discriminator's F_1 score. $\hat{F}_1$ is given by

$$\hat{F}_1(\boldsymbol{X}) = \frac{F_1(\boldsymbol{X})}{F_1(\boldsymbol{X}_{\text{ID}} \cup \boldsymbol{X}_{\text{O3D}})} - 1 \tag{6}$$

and describe the change of the score relative to the baseline dataset. We show the mean of 100 individual trial runs. We see that the OOD detector trained with our proposed OOD sampling methods performs comparable.

Table 2. Influence of data set selection on model metrics

$\boldsymbol{X}^{\text{train}} \in$	$\hat{F}_1$
$\boldsymbol{X}_{\text{ID}} \cup \boldsymbol{X}_{\text{O3D}}$	**0.00**
$\boldsymbol{X}_{\text{ID}} \cup \boldsymbol{X}_{\text{GHO}}$	-0.02
$\boldsymbol{X}_{\text{ID}} \cup \boldsymbol{X}_{\text{HBO}}$	-0.01
$\boldsymbol{X}_{\text{ID}} \cup \boldsymbol{X}_{\text{SBO}}$	-0.01

Table 3 describes the Wasserstein distance between two datasets using Dynamic Time Warping (DTW) [33] to measure distance between individual samples. This provides an intuition about the similarities of the generated OOD samples with respect to the methods used, the baseline, and the ID dataset. The distance matrix $\hat{\Delta}$ between datasets is defined row and column-wise as

$$W_d^{(\text{norm})}(P_i, P_j) = W_d(P_i, P_j) / \max W_d(P_x, P_y) \tag{7}$$

with $i, j, x, y \in \{\text{ID}, \text{O3D}, \text{GHO}, \text{SBO}, \text{HBO}\}$.
P_i and P_j are shown in the columns and rows respectively, where they are indicated by their corresponding dataset. Because of metrics' symmetry, only a triangular excerpt is shown.

Interpreting these results in light of the definition of OOD detection given earlier that we aim to separate samples originating of the ID from those of the ODD. In this context, it can be said that SBO seems most promising in OOD generation as it has the highest Wasserstein distance to not only all other generated datasets but also the original ID dataset. Moreover, we also note, that all presented methods have a higher Wasserstein distance to the ID dataset than the baseline $\boldsymbol{X}_{\text{O3D}}$. This hints that the generated samples are further away from the ID than the baseline.

Table 3. Normalized DTW Wasserstein distance between datasets

$W_d^{(\text{norm})}$	$\boldsymbol{X}_{\text{ID}}$	$\boldsymbol{X}_{\text{O3D}}$	$\boldsymbol{X}_{\text{GHO}}$	$\boldsymbol{X}_{\text{SBO}}$	$\boldsymbol{X}_{\text{HBO}}$
$\boldsymbol{X}_{\text{ID}}$	0	0.375	0.818	**0.835**	0.761
$\boldsymbol{X}_{\text{O3D}}$		0	0.976	**1.000**	0.948
$\boldsymbol{X}_{\text{GHO}}$			0	**0.230**	0.204
$\boldsymbol{X}_{\text{SBO}}$				0	0.026
$\boldsymbol{X}_{\text{HBO}}$					0

4.2. Image data

In this section, we transfer the concept to image data and use the same procedure as described above on the MNIST dataset to generate OOD images depicted in Fig. 5. Therefore, we train a VAE on the MNIST dataset and apply our OOD detection methods to the learned latent representation. Besides that, we also consider a different (but similar) dataset NotMNIST [6] (i.e., collection of fonts and glyphs), which we use to evaluate our methods' capability to provide suitable OOD samples to train OOD detectors.

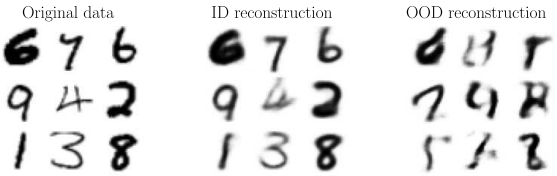

Figure 5. *From left to right*: Original images of the MNIST dataset; ID reconstructions from a VAE trained on MNIST; OOD reconstruction from the same VAE using SBO to transform ID samples within the latent space of the VAE. Plots use matching indices which explains partial similarity seen in the OOD reconstructions.

We train separate OOD detection networks using samples from all proposed methods. For OOD detection, we use the method presented by Huseljic et al. [23]. As a baseline, we use a OOD detection network trained without any synthetic OOD samples. All networks use the LeNet architecture [27]. To evaluate an OOD detectors' ability to detect OOD samples, we evaluate the *Area Under Receiver Operating Characteristic Curve* (AUROC), which is a measure indicating the ability to separate ID and OOD samples.

We compute it by considering a binary classification problem (MNIST vs. NotMNIST) where we use the estimated uncertainty from our model and determine whether a sample originates from either ID or OOD regions. More specifically, we compute the area under the graph that is obtained by plotting the true positive rate against the false-positive rate. In contrast to other measures, the AUROC is independent regarding the threshold and, therefore, is often used to evaluate methods used for OOD detection [31].

Table 4. Mean Results ($\pm 2\sigma$) over five repetitions for OOD detection on Mnist (ID) vs. NotMNIST (OOD).

	AUROC
no OOD	0.9685±0.0037
GHO	0.9888±0.0049
HBO	**0.9908±0.0028**
SBO	0.9843±0.0045

By using our synthetically generated OOD samples, we can see that the OOD detection capability of the neural network is greatly improved. Moreover, we see that HBO (as a special variant of SBO) scores best.

4.3. Cyclist Trajectories

This section presents a case study regarding the application of SBO for cyclist trajectories generation at an experimental research intersection [13]. The generation of uncommon and still not unrealistic trajectories is an essential building block for safeguarding, i.e., validating the generalization performance, of AI-based trajectory prediction methods in automated vehicles [3]. Furthermore, increasing the amount of data is essential to optimize the training process, as the number of abnormal instances in the training data is usually sparse. The cyclists' head positions in the following example are tracked in a two-dimensional coordinate system based on a triangulation technique [4] using two cameras permanently positioned at the intersection. The data consists of 1032 head trajectories, which we split into three different sets for training, evaluation, and testing. The training dataset with 715 trajectories is used to train a VAE to model the distribution of cyclists' movement at the experimental intersection using a low dimensional probabilistic latent representation. The input space of the VAE consists of trajectory segments with a length of 5 seconds, i.e., 250 two-dimensional (x- and y-axis) positions for a sampling rate of 50 Hz. The encoder comprises two hidden layers, i.e., the first hidden layer has 50 neurons, and subsequently, the bottleneck layer describes a two-dimensional latent space. The decoder presents a mirrored structure of the encoder. Moreover, for training, we use the Adam optimizer [24]. As reconstruction loss, we use the mean-squared error of the reconstructed positions. We depict the two-dimensional latent representation of the VAE in Fig. 6.

The generation of OOD instances with our SBO method is performed with an offset distance $d^+ = 0.1$ and a softness value of $\sigma = 0.5$. In this experimental scenario, we focus on showing the influence of the minimal distance parameter d_{min} for generating cyclist trajectories. We investigate for the SBO procedure three different values for the minimal distance d_{min} of 0.1, 0.5, and 1. The test dataset's reconstruction (i.e., ID data) results are presented in Fig. 7 in the blue region.

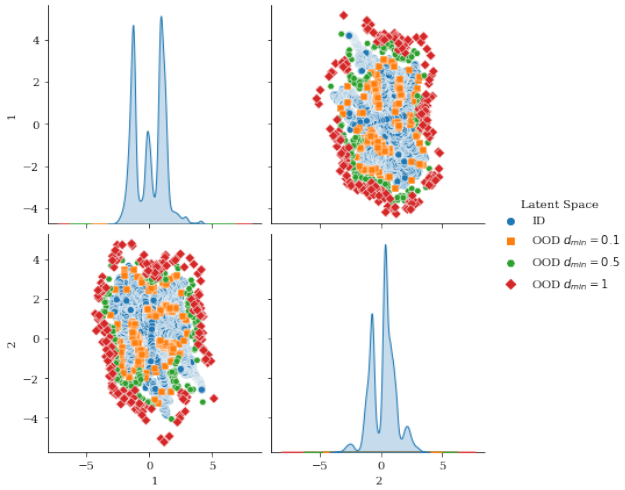

Figure 6. Two-dimensional latent space representation of trajectories (ID blue dots) using a VAE and OOD samples (OOD colored dots) generated with the SBO method using three different values for the parameter d_{min}.

The OOD trajectories generated within three different parameter values for d_{min} are shown in Fig. 7. We can appreciate a high correlation between the value of the parameter $d_{min} = 1$ and the appearance of the reconstructed red trajectories, as they are abnormal instances which are significantly different than the blue ones. Most likely, these trajectories are more unrealistic trajectories (e.g., cross building). The OOD instances created with the parameter values of $d_{min} = 0.1$ and $d_{min} = 0.5$ are more realistic. These reconstructions represent vital OOD samples, which are close to the ID representation.

5. Conclusion & Outlook

This article proposed GHO, HBO, and SBO, three strategies for OOD generation. Moreover, we proved the successful application of the OOD sampling methods with synthetic time series data, images, and cyclists' trajectories. We demonstrated that the methods could improve OOD detection and serve as a basis for

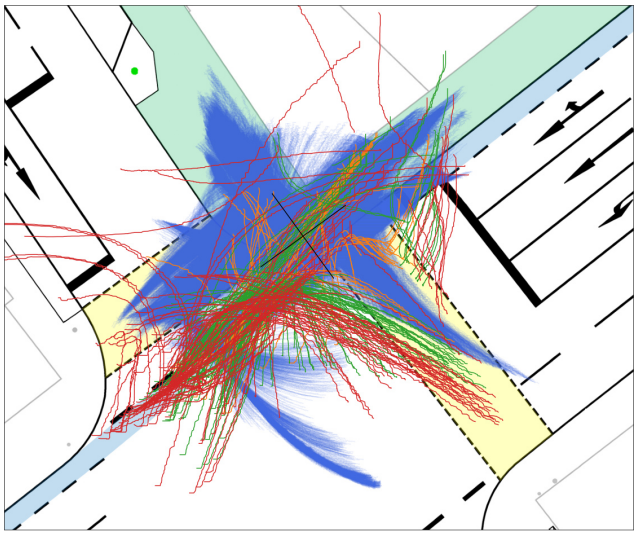

Figure 7. Top view of the experimental intersection with ID and OOD trajectories. The blue trajectories represent the ID data, and the trajectories in orange, green, and red represent the synthetically generated OOD samples.

safeguarding AI-based trajectory prediction methods in the automated driving domain.

The results obtained so far for OOD generation are extremely promising and an important starting point for future research in the field. As it turned out in our experiments with time series, one of the main critical points for successful OOD sampling is a proper representation and probabilistic model of the data. With this respect, we aim to investigate more sophisticated representation learning techniques such as SIREN [40] or VAE-LSTM [9] to model the data. In this context, one focus is the investigation of modeling techniques that better represent similarities between samplings in original and latent space (e.g., geodesic) [8]. Another line of investigation is the generation of cyclist and pedestrian body trajectories (i.e., body poses) using our SBO method. Here, we also aim to include additional conditions for the generation of more realistic OOD trajectories. For example, we aim to describe pedestrians' motion patterns by a discrete set of motion states (e.g., walking, moving, and turning) and automatically generate appropriate abnormal but still realistic body pose trajectories that can be used to safeguard AI-based prediction models.

Acknowledgment

This work results from the project AIMEE (01IS19061) funded by the German Federal Ministry of Education and Research (BMBF), the project KI Data Tooling (19A20001O) funded by the German Federal Ministry for Economic Affairs and Energy (BMWI), and the project DeCoInt[2] supported by the German Research Foundation (DFG) within the priority program SPP 1835: "Kooperativ interagierende Automobile", grant number SI 674/11-2.

References

[1] Dario Amodei, Chris Olah, Jacob Steinhardt, Paul Christiano, John Schulman, and Dan Mané. Concrete Problems in AI Safety. *arXiv preprint arXiv:1606.06565*, 2016.

[2] Roman V. Belavkin. Relation Between the Kantorovich– Wasserstein Metric and the Kullback–Leibler Divergence. *Springer Proceedings in Mathematics & Statistics*, page 363–373, 2018.

[3] Maarten Bieshaar. *Cooperative Intention Detection using Machine Learning–Advanced Cyclist Protection in the Context of Automated Driving.* Intelligent Embedded Systems. Kassel University Press, 2021. (Dissertation, University of Kassel, Department Electrical Engineering/ Computer Science).

[4] Maarten Bieshaar, Günther Reitberger, Stefan Zernetsch, Bernhard Sick, Erich Fuchs, and Konrad Doll. Detecting Intentions of Vulnerable Road Users Based on Collective Intelligence. In *Proc. of AAET*, pages 67–87, Braunschweig, Germany, 2017.

[5] Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics).* Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[6] Yaroslav Bulatov. NOT-MNIST Data Set, 2011. https://yaroslavvb.blogspot.com/2011/09/notmnist-dataset.html, accessed 21.03.2021.

[7] Jiefeng Chen, Yixuan Li, Xi Wu, Yingyu Liang, and Somesh Jha. Robust Out-of-distribution Detection for Neural Networks. *arXiv preprint arXiv:2003.09711*, 2020.

[8] Nutan Chen, Alexej Klushyn, Francesco Ferroni, Justin Bayer, and Patrick van der Smagt. Learning Flat Latent Manifolds with VAEs. *arXiv preprint arXiv:2002.04881*, 2020.

[9] Run-Qing Chen, Guang-Hui Shi, Wan-Lei Zhao, and Chang-Hui Liang. Sequential VAE-LSTM for Anomaly Detection on Time Series. *arXiv preprint arXiv:1910.03818*, 2019.

[10] Monroe D. Donsker. An Invariance Principle for Certain Probability Limit Theorems. *Memoirs of the AMS*, 6, 1951.

[11] Monroe D. Donsker. Justification and extension of doob's heuristic approach to the kolmogorov–smirnov theorems. *Annals of Mathematical Statistics*, 23(2):277–281, 1952.

[12] Albert Einstein. *Investigations on the Theory of Brownian Movement*. Dover, 1956.

[13] Michael Goldhammer, Elias Strigel, Daniel Meissner, Ulrich Brunsmann, Konrad Doll, and Klaus Dietmayer. Cooperative Multi Sensor Network for Traffic Safety Applications at Intersections. In *Proc. of ITSC*, pages 1178–1183, Anchorage, AK, 2012.

[14] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT Press, 2016.

[15] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Proc. of NeurIPS*, pages 2672–2680, 2014.

[16] Christian Gruhl and Bernhard Sick. Novelty detection with CANDIES: a holistic technique based on probabilistic models. *IJMLC*, 9(6):927–945, 2018.

[17] Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On Calibration of Modern Neural Networks. *arXiv preprint arXiv:1706.04599*, 2017.

[18] Florian Heidecker, Jasmin Breitenstein, Kevin Rösch, Jonas Löhdefink, Maarten Bieshaar, Christoph Stiller, Tim Fingscheidt, and Bernhard Sick. An Application-Driven Conceptualization of Corner Cases for Perception in Highly Automated Driving. *arXiv preprint arXiv:2103.03678*, 2021.

[19] Florian Heidecker, Abdul Hannan, Maarten Bieshaar, and Bernhard Sick. Towards Corner Case Detection by Modeling the Uncertainty of Instance Segmentation Networks. In *Proc. of ICPR, Workshop IADS*, pages 1–8, Milan, Italy, 2021.

[20] Dan Hendrycks and Kevin Gimpel. A Baseline for Detecting Misclassified and Out-of-Distribution Examples in Neural Networks. In *Proc. of ICLR*, Toulon, France, 2017.

[21] Dan Hendrycks, Mantas Mazeika, and Thomas Dietterich. Deep Anomaly Detection with Outlier Exposure. In *Proc. of ICLR*, New Orleans, LA, 2019.

[22] Sebastian Houben, Stephanie Abrecht, Maram Akila, Andreas Bär, Felix Brockherde, and et al. Inspect, Understand, Overcome: A Survey of Practical Methods for AI Safety. Technical report, Safe AI for Automated Driving, 2020.

[23] Denis Huseljic, Bernhard Sick, Marek Herde, and Daniel Kottke. Separation of Aleatoric and Epistemic Uncertainty in Deterministic Deep Neural Networks. In *Proc. of ICPR*, 2020.

[24] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. In *Proc. of ICLR*, San Diego, CA, 2015.

[25] Diederik P. Kingma and Max Welling. Auto-Encoding Variational Bayes. In *Proc. of ICLR*, 2013.

[26] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell. Simple and Scalable Predictive Uncertainty Estimation using Deep Ensembles. *arXiv preprint arXiv:1612.01474*, 2016.

[27] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proc. of IEEE*, 86(11):2278–2324, 1998.

[28] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training Confidence-calibrated Classifiers for Detecting Out-of-Distribution Samples. In *Proc. of ICLR*, Vancouver, BC, 2018.

[29] Kimin Lee, Honglak Lee, Kibok Lee, and Jinwoo Shin. Training Confidence-calibrated Classifiers for Detecting Out-of-Distribution Samples. In *Proc. of ICLR*, 2018.

[30] Shiyu Liang, Yixuan Li, and R. Srikant. Enhancing The Reliability of Out-of-distribution Image Detection in Neural Networks. In *Proc. of ICLR*, Vancouver, BC, 2018.

[31] Andrey Malinin and Mark Gales. Predictive Uncertainty Estimation via Prior Networks. In *Proc. of NeurIPS*, pages 7047–7058, 2018.

[32] Andrey Malinin and Mark Gales. Reverse KL-Divergence Training of Prior Networks: Improved Uncertainty and Adversarial Robustness. In *Proc. of NeurIPS*, pages 14547–14558, 2019.

[33] Meinard Müller. *Dynamic Time Warping*, pages 69–84. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007.

[34] C. Park. Presentations of Gaussian Processes by Wiener Processes. *Pac. J. Math.*, 94(2):407–415, 1981.

[35] Fabian Pedregosa, Gaël Varoquaux, Alexandre Gramfort, Vincent Michel, Bertrand Thirion, and et al. Scikit-learn: Machine Learning in Python. *JMLR*, 12:2825–2830, 2011.

[36] Jie Ren, Peter J. Liu, Emily Fertig, Jasper Snoek, Ryan Poplin, Mark A. DePristo, Joshua V. Dillon, and Balaji Lakshminarayanan. Likelihood Ratios

for Out-of-Distribution Detection. *arXiv preprint arXiv:1906.02845*, 2019.

[37] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards Real-Time Object Detection with Region Proposal Networks. *TPAMI*, 39(6):1137–1149, 2017.

[38] David E. Rumelhart and James L. McClelland. *Learning Internal Representations by Error Propagation*, pages 318–362. MIT Press, 1987.

[39] Cory Simon. Generating uniformly distributed numbers on a sphere, 2015. http://corysimon.github.io/articles/uniformdistn-on-sphere/, accessed 19.03.2021.

[40] Vincent Sitzmann, Julien N.P. Martel, Alexander W. Bergman, David B. Lindell, and Gordon Wetzstein. Implicit Neural Representations with Periodic Activation Functions. *arXiv preprint arXiv:2006.09661*, 2020.

[41] Kumar Sricharan and Ashok Srivastava. Building robust classifiers through generation of confident out of distribution examples. *arXiv preprint arXiv:1812.00239*, 2018.

[42] National Highway Traffic Safety Administration U.S. Department of Transportation. PE 16-007 Technical Report, 2017.

[43] Sachin Vernekar, Ashish Gaurav, Vahdat Abdelzad, Taylor Denouden, Rick Salay, and Krzysztof Czarnecki. Out-of-distribution Detection in Classifiers via Generation. *arXiv preprint arXiv:1910.04241*, 2019.

[44] Stefan Zernetsch, Hannes Reichert, Viktor Kress, Konrad Doll, and Bernhard Sick. Trajectory Forecasts with Uncertainties of Vulnerable Road Users by Means of Neural Networks. In *Proc. of IV*, pages 810–815, Paris, France, 2019.