IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received 15 October 2023, accepted 20 November 2023, date of publication 28 November 2023,
date of current version 5 December 2023.

Digital Object Identifier 10.1109/ACCESS.2023.3337658

RESEARCH ARTICLE

Out-of-Distribution Data Generation for Fault Detection and Diagnosis in Industrial Systems

JEFKINE KAFUNAH[1], PRIYANKA VERMA[1], (Senior Member, IEEE), MUHAMMAD INTIZAR ALI[2], AND JOHN G. BRESLIN[1], (Senior Member, IEEE)
[1]School of Engineering and the Data Science Institute, University of Galway, Galway, H91 TK33 Ireland
[2]School of Electronic Engineering, Dublin City University, Dublin 9, Ireland

Corresponding author: Jefkine Kafunah (jefkine.kafunah@insight-centre.org)

This work was supported in part by the Grant from the Science Foundation Ireland under Grant 16/RC/3918 (Confirm), and in part by the Grant from Science Foundation Ireland (SFI) under Grant 12/RC/2289_P2 (Insight).

ABSTRACT The emergence of Industry 4.0 has transformed modern-day factories into high-tech industrial sites through rapid automation and increased access to real-time data. Deep learning approaches possessing superior capabilities for intelligent, data-driven fault diagnosis have become critical in ensuring process safety and reliability in these industrial sites. However, such applications trained exclusively on in-distribution process data face challenges in the wake of previously unseen out-of-distribution (OOD) data in the real world. This paper addresses the challenge of out-of-distribution data detection for deep learning-based fault diagnosis models by generating synthetic data to simulate real-world anomalies not present in the training set. We propose Manifold Guided Sampling (MGS), a data-driven method for generating synthetic OOD samples from the in-distribution data-supporting manifold estimated through a deep generative model. Synthetic data from MGS enhances the model capacity for prediction uncertainty quantification, resulting in safe and reliable models for real-world industrial process monitoring. Furthermore, the MGS algorithm maintains the in-distribution data feature space as a reference point during data generation to ensure the resulting synthetic OOD data is realistic. We analyze the effectiveness of MGS through experiments conducted on the steel plates faults dataset and demonstrate that augmenting training data with synthetic data from MGS enhances the model performance in OOD detection tasks and provides robustness against dataset distributional shifts. The findings underscore the effectiveness of utilizing synthetic MGS-generated OOD data in scenarios where real-world OOD data is limited, enabling better generalization and more reliable fault detection in practical applications.

INDEX TERMS Deep generative models, fault diagnosis, process monitoring, safety-critical, out-of-distribution data, variational autoencoder, uncertainty estimation.

I. INTRODUCTION

Industry 4.0 (I4.0) has revolutionized modern-day factories through rapid automation and increased access to real-time data from complex industrial processes [1], [2], [3], [4], [5]. Central to the proliferation of industrial process datasets are multitudes of integrated sensors that gather data, resulting in large-scale, high-dimensional, and nonlinear historical process data. The compiled datasets are the in-distribution (ID) data representing some underlying industrial process.

The associate editor coordinating the review of this manuscript and approving it for publication was Guillermo Valencia-Palomo.

Recently, data-driven fault diagnosis (FD) models trained on large-scale industrial process datasets using deep learning (DL) techniques have demonstrated the ability to deliver actionable insights required to cope with the increasing demands around safety, efficiency, and production quality [6], [7], [8]. For DL-based FD models, the underlying assumption is that the training and testing data are independent and identically distributed.

However, during deployments in the real world, gradual changes over time result in data distributional shifts and the emergence of out-of-distribution (OOD) data [9], [10], [11], [12], [13]. In industrial applications, exposure of

data-collecting sensors to potentially harsh and variable environmental conditions, physical shock or damage, excess electrical noise, imprecise pre- and post-deployment sensor calibration, sensor drifts, process parameter variations, and changes in working conditions are some of the factors commonly associated with data distributional shifts and the emergence of OOD data [14], [15], [16], [17], [18], [19].

In recent years, DL algorithms have achieved state-of-the-art (SOTA) performance in a broad range of tasks [20], [21], [22], [23], leading to the integration of DL into safety-critical tasks such as biometric identification [24], medical diagnosis [25], and fully autonomous driving [26], [27], [28]. Similar adoptions in manufacturing are increasingly common, such as the uptake of deep neural networks (DNNs) in FD as the preferred process monitoring approach for complex industrial process data [8], [29]. Nonetheless, current SOTA DL models are known to generate inaccurate and overconfident predictions on OOD data, further degrading the performance of DL-based FD systems [30], [31], [32], [33], [34]. Improving capacity for OOD detection is crucial in safeguarding the DL-based FD models, especially for safety-related systems where the consequences of wrong predictions can be catastrophic, leading to the total shutdown of entire operations, while in other cases, injuries or the loss of life.

Data-driven DL-based FD models require a comprehensive dataset with broad coverage of operating conditions and fault scenarios to model the accurate system behavior during training. Therefore, data quality and availability directly affect the performance of DL-based FD models. Insufficient training data, also known as data scarcity, affects DL-based FD models by restricting the exploration of a comprehensive data feature space, thus limiting the model's ability to learn the most informative and discriminative features required for OOD detection tasks [35]. Data scarcity also relates to the long-tailed distribution problem or imbalanced dataset, common in safety-critical industrial systems where actual fault scenarios are rare and hard to simulate. The long-tailed distribution problem impacts the generalization of DL-based FD models as they tend to perform well on the dominant classes, unlike the less frequent classes [36], [37]. DL-based FD models with poor generalization tend to perform poorly in OOD detection tasks. The data scarcity problem underscores the need for additional training data to improve DL-based FD model generalization and capacity for OOD detection. Furthermore, despite the effectiveness of approaches such as Reverse KL-divergence Prior Networks (RKL-PNs) [34], [38], Aleatoric Epistemic uncertainty DNNs (AE-DNNs) [39], and Out-of-DIstribution detector for Neural networks (ODIN) [40] in OOD detection, all these approaches require access to realistic OOD data during training.

This paper addresses the challenges of training data quality and availability for data-driven DL-based FD systems by generating synthetic OOD data that simulate real-world anomalies not present in the training set. We aim to bridge the gap in training DL-based FD models using SOTA approaches such as RKL-PNs, AE-DNNs, and ODIN with additional insight from the synthetic OOD data. Our research aims to enhance the robustness of DL-based FD systems against OOD data in real-world deployment scenarios.

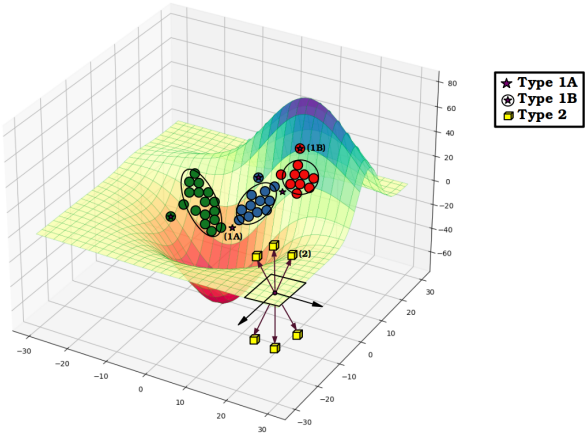

FIGURE 1. Types of OOD data. Type 1A: classwise samples on the intersecting regions of the manifold, Type 1B: classwise samples on low-probability regions of the manifold, and Type 2: classwise samples located in proximity regions outside the learned ID manifold.

We propose MGS, a data-driven method for generating synthetic OOD data based on a deep generative model. Implementation of MGS begins by training a variational autoencoder (VAE) to obtain a learned ID data-supporting manifold of the large-scale high-dimensional nonlinear historical process data. From the manifold-related hypotheses on high-dimensional data [41], [42], [43], [44], we observe that samples existing in (i) the classwise intersecting regions on the manifold, (ii) the classwise low-probability regions on the manifold, and or (iii) regions located outside the learned ID manifold; all represent regions from which we can obtain OOD latent variables (see Fig. 1). Therefore, decoding the OOD latent variables obtained from sampling the regions of interest on the manifold generates a combined set of OOD historical process data. MGS facilitates the generation of realistic OOD samples that augment the original ID training dataset. In particular, the availability of synthetic OOD data enables us to train DL-based FD applications using approaches similar to RKL-PNs and AE-DNNs. Throughout the training process, a dual loss function merges two objective functions using a convex combination that optimizes the ID samples on the one hand and OOD samples on the other. The resulting DL-based FD model offers the improved capacity to handle complex real-world industrial environments through enhanced performance on tasks such as OOD detection and uncertainty estimation.

Based on our approach, we summarise our main contributions as follows:

- We propose MGS, a data-driven method that generates synthetic OOD samples by leveraging an ID data-supporting manifold estimated through a deep generative model.

- We improve the quality of synthetic OOD data by learning disentangled OOD latent variables, performing targeted sampling from extremely low probability regions on the manifold, and expanding the range of angle choices for OOD latent variables outside the manifold.
- We demonstrate that incorporating synthetic OOD data from MGS improves the capacity for FD models to detect OOD input data and estimate predictive uncertainty, resulting in reliable FD models for real-world industrial process monitoring.

II. RELATED WORK

A. FAULT DIAGNOSIS METHODS

FD methods fall under four general categories: model-based, knowledge-based, data-driven, and hybrid approaches [45], [46]. Data-driven methods for FD have gained significant popularity and effectiveness in dynamic modern industrial environments, especially with the advancement of machine learning (ML) and artificial intelligence (AI) technologies. This work focuses mainly on data-driven DL-based FD methods that leverage large-scale industrial process datasets to learn patterns and relationships directly from data, making them more adaptable and proficient in detecting and diagnosing faults or anomalies.

He and He [47] introduce a method for diagnosing bearing faults using DL. The approach involves preprocessing sensor signals through a short-time Fourier transform (STFT) and, to detect bearing faults, constructs an optimized deep learning structure called a large memory storage retrieval (LAMSTAR) neural network using the resulting spectrum matrix. The LAMSTAR network uses Self-Organizing Maps (SOM) models to process the spectrum matrix that identifies subpatterns in input data for bearing fault diagnosis. Results suggest that the LAMSTAR network-based method performs better at 'normal' and relatively low input shaft speeds. Li et al. [48] present an approach for diagnosing motor bearing faults using neural networks and time/frequency-domain vibration analysis. Vibration simulation enables the design of various motor rolling bearing FD strategies. Results show that neural networks can interpret motor-bearing vibration signatures effectively. Jiang et al. [49] propose an improved deep recurrent neural network (DRNN) method, alleviating the need for manual feature extraction and selection for intelligent fault diagnosis. DRNN uses frequency spectrum sequences as inputs to reduce input size, improve robustness, and adopt an adaptive learning rate to enhance training performance. The DL-based FD models in [47], [49], and [48] are application-specific, relying on specialized feature extraction techniques.

Further, Wen et al. [50] propose a new Convolutional Neural Network (CNN) based on LeNet-5 for fault diagnosis. The proposed method converts signals into two-dimensional (2-D) images, allowing for better feature extraction and eliminating the effect of handcrafted features. The technique demonstrates improved prediction accuracy on popular datasets, including the motor bearing, self-priming centrifugal pump, and axial piston hydraulic pump datasets. Xia et al. [51] introduce a CNN-based method that utilizes sensor fusion to diagnose rotating machinery faults. The approach automatically extracts representative features through feature learning, eliminating the need for manual feature selection. The method applies sensor fusion at the data level for enhanced accuracy and reliability for various machinery types and faults with limited prior knowledge. We observe that FD models, depending on robust feature extraction, can be application-specific, requiring explicit knowledge of relations between process variables. Restricting OOD data during training to the application domain for application-specific models improves OOD detection results.

Xu et al. [52] propose a method for fault diagnosis using a deep transfer convolutional neural network framework. Time-domain signal data is transformed into images and used as input for a CNN-based LeNet-5 to automatically extract features and classify faults. Several offline CNNs are pretrained to improve real-time performance, and their shallow layers are transferred directly to the online CNN, significantly improving the real-time performance while achieving the desired diagnostic accuracy within a limited training time. Lu et al. [53] propose a DL-based FD model named DAFD to address cross-domain learning problems in FD. DAFD models trained in a particular source domain are adoptable in a different but related target domain. While [52] and [53] both utilize transfer learning, the latter emphasizing domain adaptation, there is a need for a dedicated strategy for dealing with OOD data. Our method seeks to address, among others, the problem of OOD detection, where synthetic samples emerge from the original target domain.

Qiao et al. [54] propose an adaptable, time-frequency dual-input model based on a CNN and long short-term memory (LSTM) network (TFWConvLSTM) to address the problem of bearing fault diagnosis under variable loads and different noise interferences. TFWConvLSTM utilizes a time-frequency dual-input structure to enhance feature extraction and adopts a CNN-LSTM structure to capture spatiotemporal characteristics. Additionally, the LSTM gate structure is employed to use temporal features and improve noise immunity fully. Zhao et al. [55] propose an end-to-end Batch-Normalization-Based LSTM (BN-based LSTM) neural network for fault diagnosis. Unlike traditional methods, BN-based LSTM trains the representation of raw input data and classifier simultaneously, utilizing the dynamic information of process data. In particular, BN-based LSTM implements batch normalization to reduce the internal covariate shift and accelerate the convergence of the LSTM network. Zhang et al. [56] propose a novel method based on gated recurrent unit neural networks for fault diagnosis of rotating machinery (FDGRU). The approach initially converts the one-dimensional time-series vibration signals into two-dimensional images, followed by applying the temporal information of the time-series to a Gated Recurrent

Unit (GRU) that learns representative features from constructed images. A multilayer perceptron (MLP) is finally employed to implement fault recognition. Zhao et al. [57] develop a new DL method, deep residual shrinkage networks (DRSNs), for FD tasks with highly noised vibration signals. Jia et al. [58] present a DNN-based intelligent method for diagnosing the faults of rotating machinery. The proposed DNN models trained on massive datasets are less dependent on human labor or prior knowledge about signal processing techniques and diagnostic expertise. We observe that the DL-based FD implementations mentioned above are deep networks with Softmax layers as the network output, resulting in overconfident model predictions for both ID and OOD samples.

B. OOD DATA GENERATION METHODS

OOD data generation is an important topic in ML as it is essential for creating safe and reliable models ready for deployment in the real world. In practice, the knowledge of OOD data distribution, a priori (during training), can reduce the tendency of DL-based FD models to make unsafe, false predictions with high confidence [59]. Generating high-quality and realistic data representative of the target distribution is one of the main challenges in OOD data generation. Currently, the main approaches for OOD data generation include data augmentation and deep generative modeling. This work focuses on synthetic data generation through deep generative modeling.

1) DATA AUGMENTATION

Inoue in [60] proposes SamplePairing, a technique for data augmentation that synthesizes a new image sample by overlapping a source image with another randomly picked from the training data through a process of pixel averaging. Zhang et al. [61] propose a simple data-agnostic augmentation routine known as *mixup* that constructs virtual training samples generated as the linear interpolation of two random samples from the training set and their labels. The *mixup* approach regularizes the neural network to favor simple linear behavior in between training examples. Further, Tokozume et al. propose Between-Class learning (BC learning) [62], an approach geared toward data augmentation for sound recognition networks. BC learning generates new data samples between class sounds by mixing two sounds belonging to different classes with a random ratio. Krizhevsky et al. [63], in their Alexnet implementation, employ variations of data augmentation techniques such as random cropping, flipping of extracted patches, and altering the intensity of RGB channels. Adaptions of the data augmentation techniques in Alexnet feature in subsequent submissions in the ImageNet Large Scale Visual Recognition Challenge (ILCVRC) [64]. Nonetheless, most of the data augmentation techniques by design focus on diversifying ID data to enhance the training set and prevent problems with overfitting and poor generalization.

2) DEEP GENERATIVE MODELING

Lee et al. in [65] propose a generative adversarial network (GAN) [66] with a modified objective function, allowing the GAN to generate OOD samples in the 'boundary' low-density regions of training distributions. During training, optimization happens jointly for two models where a confident classifier improves the proposed GAN and vice versa as training proceeds. However, the confident classifier is pre-trained on ID and OOD samples, creating an unrealistic scenario where the model has prior knowledge of OOD samples. Vernekar et al. in [67] demonstrate the inability of GANs to generate samples for a simple 3D dataset, suggesting the method will experience difficulties operating in higher dimensions. Further, Vernekar et al. [67] propose a method for generating two separate types of OOD samples from latent encodings derived from the learned manifold of a Conditional VAE (CVAE) [68]. The approach faces scaling-up challenges due to the high computing cost requirement when calculating the Jacobian over the entire dataset and capacity limitations related to the Gaussian distribution in the VAE. Motivated by the idea to relax the classic assumption of Gaussian distributed data, Mŏller et al. in [69] present Soft Brownian Offset (SBO) sampling, a method to create synthetic OOD samples at the tails of data distribution by applying transformations on the latent representations of deep generative models such as VAEs. SBO is also applicable to generic low-dimensional feature representations of the ID data. Nonetheless, the approach is limited to OOD sampling from only the low-density regions of the low-dimensional learned manifold.

Our work builds upon the concepts in [67], where we propose using Umbrella Sampling (US) [70] to access latent variables of the OOD data located in the low-density regions of the learned manifold by sampling extremely low-probability areas of the posterior distribution. Further, we utilize a class-based Jacobian, calculated from a limited sample size, resulting in efficiencies in computing cost.

III. PROPOSED MANIFOLD GUIDED SAMPLING METHOD

In this section, we present our proposed method. First, we outline the concepts of **manifold hypothesis** that form a basis for our proposed approach. Second, we provide a comprehensive overview of our methodology, including the sampling process and implementation for types 1A, 1B, and 2 OOD data. Finally, we outline the implementation of the MGS algorithm, along with the proposed pseudo-code.

A. MANIFOLDS AND HIGH-DIMENSIONAL DATA

One of the prominent characteristics of modern-day industrial datasets is the high dimensional data typically compiled in a large-scale nature. For high-dimensional data, the number of features is usually large and can easily exceed the number of observations in a dataset. Due to the challenges associated with learning in higher dimensions, [71], it is essential to identify low-dimensional subspaces of the data space containing meaningful information. A collection of

methodologies for analyzing high dimensional data based on geometrical and topological approaches support the following hypotheses:

- The **manifold hypothesis** states that real-world data presented in high-dimensional input space are more likely to concentrate on a much lower-dimensional sub-manifold embedded in the high-dimensional input space [41], [42], [43], [44].
- The **manifold hypothesis for classification** states that for multi-class data, different classes are likely to concentrate on different disjoint sub-manifolds separated by low-density regions in the input space [41], [72].

The manifold-related hypotheses are essential to many dimension reduction algorithms and other manifold-inspired algorithms [72].

B. METHODOLOGY

Given a high-dimensional ID dataset, we begin by obtaining a transformation into a lower-dimensional data space, retaining the meaningful properties present in the original data. A deep generative model such as the VAE is suitable for this task as it can model a relatively smooth latent space. In practice, the VAE generates a reconstruction of the input $\tilde{\mathbf{x}}$, given the latent variable $\mathbf{z}$ through a decoder $p_\theta(\tilde{\mathbf{x}}|\mathbf{z})$ and the encoder $q_\phi(\mathbf{z}|\mathbf{x})$, representing the variational approximate posterior. From the ID high-dimensional input space, the VAE models a lower-dimensional manifold embedding where the high-density regions correspond to dense areas of the input space. For our implementation, we use the Total Correlation Variational Autoencoder (β-TCVAE) [73], a VAE model that learns disentangled latent representations from the input data. In particular, through β-TCVAEs, we obtain a more interpretable generative model capable of understanding the role of each latent dimension in the data generation process.

Vernekar et al. in [74] propose two categories of OOD samples: Type 1 (1A and 1B): OOD samples on the data manifold and Type 2: OOD samples outside the data manifold. In this work, we adopt similar OOD sample categorizations with adjustments towards feature robustness under the influence of outliers and resource management for improved computational costs.

For **Type 1: OOD samples on the data manifold**, the low-density regions in the input space corresponding to ID data boundary regions on the manifold represent areas consisting of OOD data. Following the **manifold hypotheses** (Sec. III-A), we observe that boundary regions on the manifold have densities that gradually decrease the further away you move from the dense areas.

We begin by obtaining $q_\phi(\mathbf{z}|\mathbf{x})$ from our trained β-TCVAE model, a uni-modal multivariate Gaussian with a diagonal covariance structure. $q_\phi(\mathbf{z}|\mathbf{x})$ represents the variational approximate posterior distribution from which we seek to retrieve the outliers representing classwise Type 1A: classwise samples on the intersecting regions of manifold and Type 1B: classwise samples on low-probability regions of the manifold, (See Fig. 1: Type 1A, 1B OOD). Sampling from low probability regions of a given classwise cluster distribution $\mathbf{Z}_{\mathrm{ID}_k} \sim q_\phi(\mathbf{z}_k|\mathbf{x}_k)$ retrieves the local class k outliers existing around the respective cluster region on the manifold.

To obtain samples in the low-density regions of the learned ID data-supporting manifold, we apply US, an algorithm that performs sampling on extremely low-probability regions of a posterior distribution, accurately down to approximately $15\,\sigma$ on the credible region. The US algorithm applies temperature stratification, a technique that flattens the distribution by defining various temperatures and biasing window functions, enabling the exploration of wider parameter ranges and low-probability areas of the posterior distribution. Exponential spacing of temperatures ensures equal exchange probabilities between windows.

From class k encoder mappings $\mathbf{Z}_{\mathrm{ID}_k}$, we can estimate the mean $\boldsymbol{\mu}_{\mathrm{ID}_k}$ and covariance $\boldsymbol{\Sigma}_{\mathrm{ID}_k}$ that define the structure of the aggregate class k posterior distribution. In our application, we use the minimum covariance determinant (MCD) method [75], [76], [77], a robust estimator of the mean and covariance matrix aimed at minimizing the influence of outliers. The US algorithm then enables us to sample the low-probability regions of the class k posterior distribution, a multivariate Gaussian with mean $\boldsymbol{\mu}_{\mathrm{ID}_k}$ and covariance $\boldsymbol{\Sigma}_{\mathrm{ID}_k}$ to obtain our outlier $\mathbf{z}_{\mathrm{OOD}_k}$. Additionally, we can increase the diversity of generated OOD samples through targeted adjustments of the disentangled latent variables. To this end, we introduce a latent noise vector variable ϵ with elements in the range $[-2.5, 2.5]$. We apply random transformations to individual elements of latent vector $\mathbf{z}$ through vector addition with ϵ and decode to obtain a more diverse set of OOD data.

For **Type 2: OOD samples outside the data manifold**, we observe that samples inhabiting regions of relative proximity yet isolated from the ID data-supporting manifold represent an additional category of OOD data. To this end, we adopt the method proposed by Vernekar et al. in [74], where a sample existing in a direction perpendicular to the tangent space of the sub-manifold at a point $\mathbf{x}_{\mathrm{ID}}$ corresponds to an OOD sample $\mathbf{x}^{\perp}_{\mathrm{OOD}}$ that falls outside the manifold (See Fig. 1: Type 2). In particular, consider a VAE that models a lower-dimensional data manifold from the high-dimensional ID data $\mathbf{X}_{\mathrm{ID}}$ through corresponding latent variables $\mathbf{Z}_{\mathrm{ID}}$. The encoder $q_\phi : \mathbf{X}_{\mathrm{ID}} \rightarrow \mathbf{Z}_{\mathrm{ID}}$ and decoder $p_\theta : \mathbf{Z}_{\mathrm{ID}} \rightarrow \tilde{\mathbf{X}}_{\mathrm{ID}}$ functions provide a mapping through which we can recreate input data in the form $\tilde{\mathbf{x}}_{\mathrm{ID}} = p_\theta(q_\phi(\mathbf{x}_{\mathrm{ID}}))$ and as a result, for a given point $\mathbf{x}_{\mathrm{ID}}$, the tangent space of the manifold is the column space of the Jacobian matrix:

$$\mathbf{J}(\mathbf{x}_{\mathrm{ID}}) = \left.\frac{\partial p_\theta(\mathbf{z})}{\partial \mathbf{z}}\right|_{\mathbf{z}=q_\phi(\mathbf{x}_{\mathrm{ID}})} \tag{1}$$

Notably, the basis vectors of the left null-space of the Jacobian denoted $\mathbf{null}(\mathbf{J}^\top(\mathbf{x}_{\mathrm{ID}}))$, span the space perpendicular to the sub-manifold at the point $\mathbf{x}_{\mathrm{ID}}$. The perpendicular vector $\mathbf{v}^\perp$ is thus obtained by randomly sampling the set of unit vectors $\mathbf{V}^\perp \sim \mathbf{null}(\mathbf{J}^\top(\mathbf{x}_{\mathrm{ID}}))$. However, a primary concern is the computational cost of the Jacobian matrix

over the entire dataset. In our implementation, we make the following adjustments: (i) From the **manifold hypothesis for classification**, we observe that different classes are likely to concentrate along different sub-manifolds. Therefore, we obtain a per-class average Jacobian including the corresponding left null space upon which we derive the class-specific perpendicular vectors $\mathbf{v}_k^{\perp}$. (ii) Based on the quality of the average Jacobians, we can reduce the required number of samples to achieve reliable results to a limited number of batches $b \in [10, 50]$. Further, we replace the perturbation (vector addition transformation) with a $d \times d$ rotation matrix, where k is the same dimension as $\tilde{\mathbf{x}}_{\mathrm{ID}_k} \in \mathbb{R}^d$. We then generate $\mathbf{x}_{\mathrm{OOD}_k}^{\perp}$ by rotating $\tilde{\mathbf{x}}_{\mathrm{ID}_k}$ by an angle γ, uniformly sampled from within the range $[45°, 90°]$ towards $\mathbf{v}_k^{\perp}$. The rotation matrix provides us with a broader spectrum of choices where the greater angle sizes γ yield $\mathbf{x}_{\mathrm{OOD}_k}^{\perp}$ samples more similar to $\mathbf{v}_k^{\perp}$.

1) MGS LEARNING ALGORITHM

We outline the proposed algorithmic approach for the MGS in 1, describing the procedure to obtain synthetic OOD data from a deep generative model. Inputs for Algorithm 1 include (i) a minimum acceptable threshold distance d^* from the ID data, (ii) a perturbation value ϵ for adjusting the latent variables, and (iii) a batch size b for the decoder Jacobian matrix. The MGS algorithm uses the following four main steps in its implementation:

In the first step, we train a β-TCVAE, obtaining the ID data supporting manifold with disentangled representations of the latent variables, an encoder $q_\phi(\mathbf{z}_k|\mathbf{x}_k)$ and decoder $p_\theta(\tilde{\mathbf{x}}_k|\mathbf{z}_k)$.

In the second step, use the MCD approach to obtain the mean $\boldsymbol{\mu}_{\mathrm{ID}_k}$ and covariance $\boldsymbol{\Sigma}_{\mathrm{ID}_k}$ from the classwise posterior distribution of the latent variables $\mathbf{z}_{\mathrm{ID}_k}$. Using the US approach, we perform targeted sampling on the low-probability regions of the class k posterior distribution to obtain $\mathbf{z}_{\mathrm{OOD}_k^1}$ for classwise types 1A, 1B latent variables. For type 2, we obtain the classwise tangent space of the manifold $\mathbf{J}(\mathbf{x}_{\mathrm{ID}_k})$ averaged over the batch of size b as illustrated in equation 1. The left null-space of the Jacobian denoted $\mathbf{null}(\mathbf{J}^\top(\mathbf{x}_{\mathrm{ID}_k}))$ gives the classwise latent variable $\mathbf{z}_{\mathrm{OOD}_k^2}$ for Type 2 OOD data.

In the third step, we compile $\mathbf{z}_{\mathrm{OOD}_k^1}$ and $\mathbf{z}_{\mathrm{OOD}_k^2}$ into unified collections of classwise latent variables $\mathbf{z}_{\mathrm{OOD}_k}$. We then perturb the classwise latent variables $\mathbf{z}_{\mathrm{OOD}_k}$ in the form $(\mathbf{z}_{\mathrm{OOD}_k} \times \epsilon)$ to obtain $\tilde{\mathbf{z}}_{\mathrm{OOD}_k}$.

Finally, in step four, decoding $\tilde{\mathbf{z}}_{\mathrm{OOD}_k}$ using the decoder $p_\theta(\tilde{\mathbf{x}}_k|\tilde{\mathbf{z}}_{\mathrm{OOD}_k})$ generates the preliminary classwise OOD dataset $\tilde{\mathbf{X}}_{\mathrm{OOD}_k}$. Based upon samples that fall within a minimum acceptable distance d^* from the original input dataset $d^* \leq d(\tilde{\mathbf{x}}_{\mathrm{OOD}_k}, \mathbf{X}_k)$, we uniformly select classwise samples $\tilde{\mathbf{x}}_{\mathrm{OOD}_k}$ and compile the final OOD dataset $\mathbf{X}_{\mathrm{OOD}_k}$.

Algorithm 1 MGS Learning Algorithm

Input: minimum distance d^* set as an acceptable threshold from the ID data, perturbation value ϵ for adjusting latent variables, batch size b for decoder Jacobian matrix.

Data: $\mathcal{D} = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^N$, set of N i.i.d. labeled samples from the training dataset.

```
# Step 1
Fit a β-TCVAE on dataset D to obtain encoder
  q_φ(z_k|x_k) and decoder p_θ(x̃_k|z_k)
for i ∈ {1, ..., N} do
    # Step 2
    Generate classwise posterior distribution
      z_ID_k ~ q_φ(z_k|x_k)
    For types 1A and 1B:
    (i) Obtain the mean and covariance from the
      classwise posterior distribution using MCD
    μ_ID_k, Σ_ID_k ← MCD(z_ID_k)
    (ii) Sample low-probability regions in class k
      posterior distribution using US
    z_OOD_k^1 ~ US(μ_ID_k, Σ_ID_k)
    For type 2:
    (i) Calculate the classwise Jacobian
    J(x_ID_k) ← ∂p_θ(z)/∂z |_{z=q_φ(x_ID_k)}
    (ii) Calculate classwise left null-space for batch
      size b
    z_OOD_k^2 ~ null(J^⊤(x_ID_k))

    # Step 3
    Compile unified collections of classwise latent
      variables
    z_OOD_k ← numpy.vstack(z_OOD_k^1, z_OOD_k^2)
    Perturb z_OOD_k
    z̃_OOD_k ← (z_OOD_k × ε)

    # Step 4
    Decode z̃_OOD_k to obtain
      x̃_OOD_k ~ p_θ(x̃_OOD_k|z̃_OOD_k)
    Δd = d_min(x̃_OOD_k, X_ID_k)
    if Δd ≤ d* then
        X̃_OOD_ik ← x̃_OOD_ik
    end
end
```

Output: $\tilde{\mathbf{X}}_{\mathrm{OOD}}$, the set of generated OOD samples

IV. EXPERIMENTAL RESULTS

A. CASE STUDY

We evaluate the effectiveness of our proposed synthetic OOD data generation method, MGS, using the Steel Plates Faults dataset [78]. We compare MGS against other synthetic OOD data generation methods: OOD Detection and Generation using Soft Brownian Offset Sampling (SBO) [69] and OOD Detection in Classifiers via Generation (CGen) [74], as baselines. Finally, we augment the raw ID data with

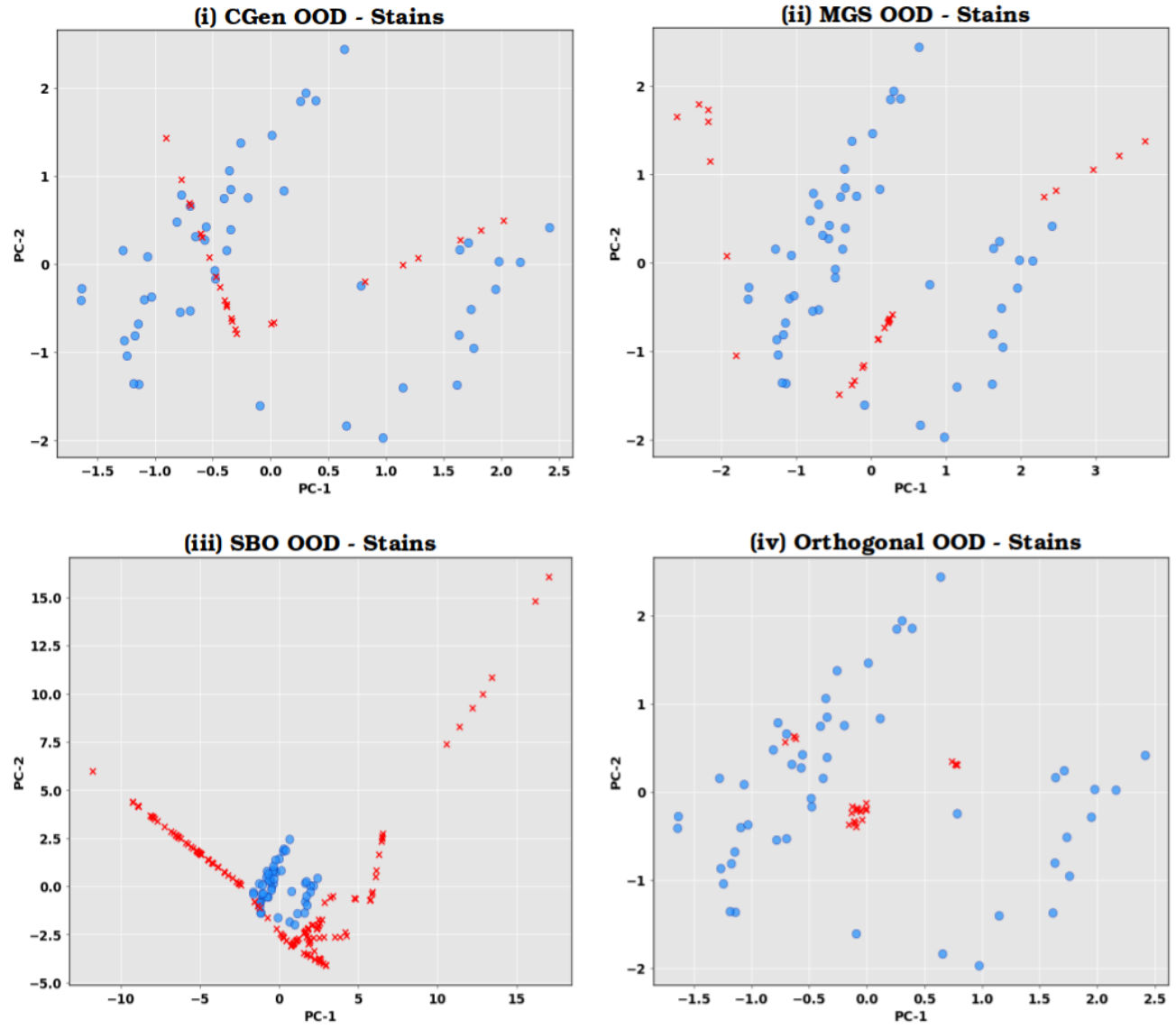

FIGURE 2. Two-dimensional plots for fault type Stains (OOD data - red and ID data - blue). Synthetic OOD data generation methods (i) CGen OOD, (ii) MGS OOD, (iii) Soft-Brownian Noise OOD (SBO OOD), and (iv) Type 2: Orthogonal projection OOD data. MGS-generated OOD intersects the least with ID data.

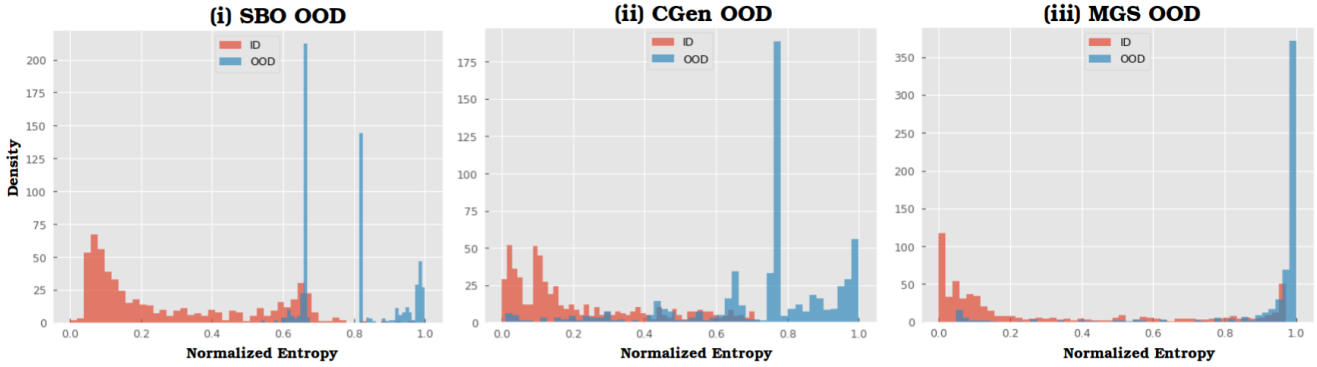

FIGURE 3. Predictive entropy density plots for ID and OOD data from AE-DNN trained on Steel Plates Faults dataset augmented using OOD from (i) SBO, (ii) CGen, and (iii) MGS methods. Entropy scores are normalized to fall within the range [0, 1]. MGS OOD improves AE-DNN capacity to detect OOD data using predictive entropy scores by achieving the best divergence between ID and OOD predictive entropy scores.

synthetic OOD data and train a DL-based FD application on the real-world industrial task of defects classification under OOD data uncertainty. Fig. 2 illustrates a selection of the results (fault type: stains) from OOD data generation methods applied to the Steel Plates Faults dataset. Comparatively, MGS-generated OOD intersects the least with ID data.

1) STEEL PLATES FAULTS DATASET

In the steel industry, intelligent fault diagnosis during steel plate production is essential for the timely identification of defects that directly influence the safety and performance of the final product. Notably, fault diagnosis in steel plate production is challenging due to the complex nature of defects owing to the dynamic production process and the quality of raw materials [79]. The steel-plate surface defect inspection system involves capturing video images of the steel plates on the rolling equipment, followed by image processing and analysis, detecting the area of the defect, extracting features from the defect area, and finally, defect classification [80].

The steel plates faults dataset consists of 1941 instances for classifying surface defects in stainless steel plates during industrial production. This is a labeled dataset where instances are classified into either of the seven distinct typologies of faults: Pastry, Z Scratch, K Scratch, Stains, Dirtiness, Bumps, and Other Faults. Each recorded instance consists of 27 attributes representing the geometric shape of the fault and its contour. For this dataset, we apply FD to diagnose the source of the fault from among the seven commonly occurring faults of the steel plates. The target class distribution reveals an imbalanced dataset.

B. EXPERIMENTAL SETUP

For synthetic OOD data generation, we utilize the β-TCVAE, a variant of the variational autoencoder that attempts to learn disentangled representations. We choose the β-TCVAE architecture as an encoder consisting of three fully-connected layers (27, 16, and 4 output features) and the decoder with three fully-connected layers (4, 16, and 27 output features). We train the β-TCVAE for 5000 epochs using the Adam optimizer [81] and a base learning rate of 0.1. We partition the data into a train/test split of 70%/30% and use a large batch size of 128. To achieve disentangled representations, we combine the mean squared error reconstruction loss with the special case β-TCVAE where for the ELBO-TC-Decomposition, we choose the following weights $\alpha = 1$ for index-code mutual information (MI), $\gamma = 1$ for dimension-wise KL and $\beta = 10$ for total correlation (TC). During the sampling stage, to obtain types 1A and 1B using the US algorithm, we select a series of higher temperatures $\{T_i\}_{i=0}^{L}$ that flatten the target distribution to allow for the exploration of wider ranges of parameters. In particular, we use the `linspace` NumPy function [82] to select 24 number evenly space between intervals from 1 to 30. For both MGS and CGen, we combine types 1A, 1B, and 2 OOD data in the ratio 70%/30%. Further, we implement the comparison method, SBO, using the hyperparameter setting $d^* = 0.45$, $d^* = 1$ and $\sigma_{\text{SBO}} = 1$.

We utilize a deep feedforward neural network (DFNN) for experiments on the DL-based FD models. The network architecture consists of four fully-connected layers (270, 216, 162, 108, 54, and 13 output features), with each layer followed by a rectified linear unit (ReLU) [83], a batch normalization layer [84], and a dropout layer [85]. We use four approaches to train our models with an aggregate dataset of ID and synthetic OOD data: (i) RKL-PNs [34], [38], (ii) AE-DNNs [39], (iii) exposing an ordinary DNN to OOD data during validation (ODNN-OOD), and (iv) training an ordinary DNN using only ID data (ODNN-ID). For the ordinary DNN, we utilize the softmax-cross entropy loss. We train the classifiers for 1000 epochs using the Adam optimizer [81] and a base learning rate of 0.1. Through a learning rate scheduler, the base learning rate adaptively changed to 0.01 at epoch 75 and 0.001 at epoch 90 during training. For the optimizer tuning, we ultimately settle on Adam with ϵ values of 10^{-4}. We partition the data into a train/test split of 70%/30% and use a large batch size of 128 for all experiments on the Steel Plates Faults dataset, an imbalanced dataset, hence increasing the chances of including samples from the minority classes in each batch during training.

Finally, to evaluate the robustness of models trained using synthetic OOD data, we infuse noise into the test data to simulate OOD data in the real-world industrial environment. We create three distinct test OOD datasets by introducing randomness through the following three methods: (i) Gaussian noise with a mean of 0 and a standard deviation of 1, (ii) Poisson noise with an influence parameter of 1, and (iii) Uniform noise within the range of -1 to 1.

1) EVALUATION METRICS

For the evaluation of models on predictive uncertainty and OOD detection, we choose the following metrics[1]:

Accuracy (ACC) $\uparrow$ measures the model performance as a percentage of correct predictions out of the total predictions made.

$$\text{ACC} = \frac{1}{N}\sum_{n=1}^{N} \mathbb{1}\left(y_n \neq \hat{y}_n\right) \tag{2}$$

Acc evaluates the model's generalization performance on a hold-out test set. The higher the accuracy score, the more accurate the model's prediction.

Expected Calibration Error (ECE) $\downarrow$ measures the consensus between classifiers' predicted probabilities (confidence) and empirical accuracy.

$$\text{ECE} = \sum_{j=1}^{J} \frac{|B_j|}{n} \left|\text{acc}\left(B_j\right) - \text{conf}\left(B_j\right)\right| \tag{3}$$

where n represents the number of samples and B_j is the bin j [59].

[1] Arrows next to the evaluation metric indicate which direction is better

TABLE 1. Accuracy, ECE, NLL, Brier, AUROC-OOD, and FPR95 test set results for RKL-PN, AE-DNN, ODNN-OOD, and ODNN-ID trained on Steel Plates Faults ID and synthetic OOD data generated from SBO, CGen, and MGS methods. Boldface values indicate better results per method.

Method	OOD Data	ACC ↑	ECE ↓	NLL ↓	Brier ↓	AUROC-OOD ↑	FPR95 ↓
RKL-PN	SBO	0.67	0.16	1.20	0.08	0.91	1.0
	CGen	0.72	**0.11**	**0.95**	**0.06**	0.89	0.98
	MGS	**0.73**	**0.11**	1.01	**0.06**	**0.93**	**0.89**
AE-DNN	SBO	0.70	0.16	1.19	0.08	0.87	1.0
	CGen	**0.71**	**0.05**	**0.99**	**0.06**	0.64	1.0
	MGS	0.69	0.08	1.06	0.07	**0.88**	**0.96**
ODNN-OOD	SBO	**0.77**	0.21	1.46	0.06	0.62	0.99
	CGen	0.75	0.21	1.43	0.06	0.81	0.98
	MGS	0.76	**0.20**	**1.33**	0.06	**0.82**	**0.97**
ODNN-ID	None	0.80	0.14	1.0	0.05	–	–

TABLE 2. ID/OOD aleatoric and epistemic test set results for RKL-PN, AE-DNN, ODNN-OOD, and ODNN-ID trained on Steel Plates Faults ID and synthetic OOD data generated from SBO, CGen, and MGS methods. Boldface values indicate better results per method.

Method	OOD Data	Alea Conf ↑	Epist Conf ↑	OOD Alea ↑	OOD Epist ↑
RKL-PN	SBO	0.7798	0.7795	0.8928	0.8976
	CGen	**0.8162**	**0.8154**	0.8889	0.8897
	MGS	0.7721	0.7720	**0.9137**	**0.9192**
AE-DNN	SBO	0.7734	0.7733	0.9601	0.9616
	CGen	**0.8551**	**0.8550**	0.8503	0.8492
	MGS	0.8138	0.8134	**0.9618**	**0.9643**
ODNN-OOD	SBO	0.8942	0.8942	0.5672	0.5672
	CGen	0.9225	0.9225	**0.7454**	**0.7465**
	MGS	**0.9317**	**0.9318**	0.7213	0.7220
ODNN-ID	None	0.9258	0.9257	0.50	0.50

TABLE 3. AUROC-OOD and FPR95 results for RKL-PN, AE-DNN, ODNN-OOD, and ODNN-ID models tested on noise-infused OOD from Gaussian, Uniform, and Poisson methods. Boldface values indicate better results per method.

		Gaussian		Uniform		Poisson	
Model	OOD Data	AUROC-OOD ↑	FPR95 ↓	AUROC-OOD ↑	FPR95 ↓	AUROC-OOD ↑	FPR95 ↓
RKL-PN	SBO	0.59	1.0	0.62	1.0	0.64	1.0
	CGen	0.77	0.98	**0.90**	**0.99**	**0.74**	0.96
	MGS	**0.78**	**0.96**	**0.90**	**0.99**	**0.74**	**0.91**
AE-DNN	SBO	0.63	1.0	0.52	1.0	0.61	1.0
	CGen	0.63	**0.96**	0.49	**0.99**	0.59	0.95
	MGS	**0.78**	**0.96**	**0.59**	**0.99**	**0.73**	**0.92**
ODNN-OOD	SBO	0.64	0.97	0.73	0.99	0.55	0.96
	CGen	**0.81**	**0.95**	**0.85**	**0.98**	**0.77**	**0.89**
	MGS	0.79	0.97	**0.85**	**0.98**	0.74	0.91
ODNN-ID	None	0.70	0.91	0.76	0.96	0.63	0.90

Brier Score (BS) ↓ measures the accuracy of predicted probabilities.

$$\text{BS} = \frac{1}{N}\sum_{i=1}^{N}\left(\hat{\boldsymbol{p}}_i - \boldsymbol{y}_i\right)^2 \tag{4}$$

computed as the mean squared error of predicted probabilities and true classes where $\hat{\boldsymbol{p}}$ is a vector of predicted probabilities and $\boldsymbol{y}$ is the one-hot encoded ground truth [86].

AUROC−OOD ↑ measures the Area Under the Receiver Operating Characteristic Curve for OOD data by posing the problem set as a binary classification with the OOD data considered the positive class.

$$\text{AUROC-OOD} = \mathbb{E}_{\mathbf{x}_{\text{OOD}},\mathbf{x}_{\text{ID}}}\Big[\mathbb{I}\big(\text{unc}(\mathbf{x}_{\text{OOD}}) \geq \text{unc}\,(\mathbf{x}_{\text{ID}})\big) \mid y_{\text{OOD}} = +1, y_{\text{ID}} = -1\Big] \tag{5}$$

where unc(·) represents the uncertainty measure [87], [88].

False Positive rate (FPR) at N% True Positive Rate (TPR) − (FPRN) ↓ measures the probability of a model misclassifying an out-of-domain input as in-domain given $N\%$ of the ID samples are correctly classified [74].

Confidence Calibration measures the correlation between confidence and correctness of model predictions. For a selected threshold, the metric is provided by the area under the precision-recall curve (AUPRC) [89] as follows:

- **Aleatoric Confidence (Alea. Conf.)** ↑ obtained using maximum class probability $\max_k \hat{\boldsymbol{p}}_k$ as the threshold and a binary set of labels where 1 corresponds to correct predictions while 0 to incorrect predictions.
- **Epistemic Confidence (Epist. Conf.)** ↑ we use the empirical variance of the predicted class $\hat{\boldsymbol{p}}_k$, as the threshold against a binary set of labels where 1 corresponds to correct predictions while 0 to incorrect predictions.

OOD Detection measures the models' ability to detect OOD samples. For a selected threshold, the metric is provided by the area under the precision-recall curve (AUPRC) [89] as follows:

- **Aleatoric OOD Detection (OOD Alea.)** ↑ obtained using maximum class probability $\max_k \hat{\boldsymbol{p}}_k$ as the threshold and a binary set of labels where 1 corresponds to in-domain data while 0 to out-of-domain data.
- **Epistemic OOD Detection (OOD Epist.)** ↑ we use the empirical variance of the predicted class $\hat{\boldsymbol{p}}_k$, as the threshold against a binary set of labels where 1 corresponds to in-domain data while 0 to out-of-domain data.

V. RESULTS AND DISCUSSION

First, we analyze the impact of synthetic OOD data on DL-based FD model performance in OOD data detection tasks. Table 1 compares OOD data from MGS, CGen, and SBO methods across classifiers trained using RKL-PNs, AE-DNN, ODNN-OOD, and ODNN-ID approaches. In the classifiers category, RKL-PNs and AE-DNN, with the loss function combining objectives for both the ID and OOD data, achieve higher AUROC-OOD scores, revealing an improved capacity for OOD data detection tasks. In particular, the RKL-PNs classifier, trained using synthetic OOD data from the MGS method, attained the overall best score of AUROC-OOD 0.93 and FPR95 score of 0.89, revealing a relatively lower misclassification probability of OOD inputs from models trained using synthetic OOD data. Furthermore, MGS outperforms both SBO and CGen approaches regarding AUROC-OOD and FPR95 scores across various classifier types, indicating the superior quality of MGS-generated synthetic OOD data. These results suggest that augmenting the Steel Plates Faults dataset with MGS-generated OOD data enhances the FD model capacity for OOD detection tasks in real-world scenarios where the OOD data may emerge.

Ordinary classifiers (ODNN-ID) trained using softmax cross-entropy obtain higher model accuracy, ECE, NLL, and Brier scores due to training exclusively on ID data. Nonetheless, incorporating OOD data during validation in the ODNN-OOD classifiers enhances the model performance on OOD detection tasks. MGS-generated synthetic OOD data achieves the best within classifier scores of 0.82 for AUROC-OOD and 0.97 for FPR95.

Table 2 presents the results from experiments investigating (i) the correlation between confidence and correctness of model predictions and (ii) measures the models' capacity for OOD detection. Fundamentally, aleatoric and epistemic confidence metrics (Alea. Conf and Epist. Conf) seek to establish the likelihood of correct predictions given high confidence. For the RKL-PN and AE-DNN classifiers, the CGen-generated OOD data achieves the best confidence scores. Nonetheless, the ODNN-OOD classifier using MGS-generated OOD data during validation has the best overall confidence scores at 0.9317 aleatoric and 0.9318 epistemic, indicating model predictions that are more likely to be correct given the increase in confidence. For the OOD detection tasks, investigations reveal that using the MGS-generated OOD data enhances the performance of RKL-PN and AE-DNN classifiers, evidenced by the significant improvements over the other OOD generation approaches. In particular, augmenting the Steel Plates Faults dataset with MGS-generated OOD data for the AE-DNN classifier achieves the best overall scores at 0.9618 OOD aleatoric and 0.9643 OOD epistemic. The CGen-generated OOD data achieves the best scores for the ODNN-OOD, while failure to use any OOD data during training yields the poorest scores of 0.50 for both OOD aleatoric and OOD epistemic, further demonstrating the significance of synthetic OOD data in the training of safety-related FD applications.

Fig. 3 illustrates the predictive entropy density plots of ID and OOD data from AE-DNN trained on the Steel Plates Faults dataset augmented using OOD data from SBO, CGen, and MGS methods. Augmenting ID data using MGS-generated OOD yields the best divergence in predictive entropies, with OOD samples predominantly obtaining high entropies while ID obtaining low entropies. Notably, distinguishing between ID and OOD data is essential for DL-based FD systems deployed in safety-related industrial environments, in this case, implementable through a thresholding-based system. The distinction in predictive uncertainties between ID and OOD samples highlights the benefits of using MGS-generated OOD data to enhance the capacity for OOD detection tasks.

Table 3 presents the results from experiments evaluating the DL-based FD model robustness against a collection of noise-infused OOD data. We observe that training classifiers using a combination of ID and synthetic OOD data achieves superior model performance in detecting noise-infused OOD data. In particular, augmenting the Steel Plates Faults dataset with MGS-generated OOD data during training enhances the performance of RKL-PN and AE-DNN classifiers, as evidenced by the AUROC-OOD and FPR95 scores. ODNN-OOD classifiers with access to MGS and CGen synthetic OOD data during training outperform ODNN-ID classifiers with the observation that CGen-generated OOD data achieves the best improvement across the ODNN category.

VI. CONCLUSION

This paper proposes Manifold Guided Sampling (MGS), a data-driven method for generating synthetic out-of-distribution (OOD) data based on deep generative networks. In particular, MGS leverages an in-distribution (ID) data-supporting manifold of large-scale industrial process data and a combination of strategic manifold sampling techniques to create realistic OOD data. Through MGS, we address the challenges of training data quality and availability for data-driven deep learning-based fault diagnosis systems by generating synthetic OOD data that simulate real-world anomalies not present in the training set. We demonstrate the impact of augmenting ID data with synthetic OOD data during training for models, with results that suggest the synthetic data improves the model capacity for OOD detection and provides robustness to the effects of distributional shifts.

Furthermore, MGS samples low-probability regions of the manifold and is more efficient in terms of compute resources due to the utilization of smaller batch sizes when generating the tangent space of the manifold. It maintains the in-distribution data feature space as a reference point during data generation and applies a similarity distance constraint to ensure the resulting synthetic data is realistic. Our results show the best distinction between ID and OOD data, which is crucial for systems deployed in safety-related industrial environments. In future work, we aim to investigate the effectiveness of our approach on time-series datasets and high-resolution sensor data such as large-scale multimodal camera-LiDAR datasets.

REFERENCES

[1] K.-D. Thoben, S. Wiesner, and T. Wuest, "'Industrie 4.0' and smart manufacturing—A review of research issues and application examples," *Int. J. Autom. Technol.*, vol. 11, pp. 4–19, Jan. 2017.

[2] P. O'Donovan, K. Bruton, and D. T. J. O'Sullivan, "Case study: The implementation of a data-driven industrial analytics methodology and platform for smart manufacturing," *Int. J. Prognostics Health Manage.*, vol. 7, no. 3, pp. 1–22, Nov. 2020.

[3] J. Davis, T. Edgar, R. Graybill, P. Korambath, B. Schott, D. Swink, J. Wang, and J. Wetzel, "Smart manufacturing," *Annu. Rev. Chem. Biomol. Eng.*, vol. 6, no. 1, pp. 141–160, Jul. 2015.

[4] Jonathan G. Koomey, H. Scott Matthews, and Eric Williams, "Smart everything: Will intelligent systems reduce resource use? *Annu. Rev. Environ. Resour.*, vol. 38, no. 1, pp. 311–343, 2013.

[5] D. M. Tilbury, "Cyber-physical manufacturing systems," *Annu. Rev. Control, Robot., Auto. Syst.*, vol. 2, no. 1, pp. 427–443, May 2019.

[6] K. Hadad, M. Pourahmadi, and H. Majidi-Maraghi, "Fault diagnosis and classification based on wavelet transform and neural network," *Prog. Nucl. Energy*, vol. 53, no. 1, pp. 41–47, Jan. 2011.

[7] L. Wen, L. Gao, and X. Li, "A new deep transfer learning based on sparse auto-encoder for fault diagnosis," *IEEE Trans. Syst. Man, Cybern. Syst.*, vol. 49, no. 1, pp. 136–144, Jan. 2019.

[8] Y.-J. Park, S.-K.-S. Fan, and C.-Y. Hsu, "A review on fault detection and process diagnostics (don't short) in industrial processes," *Processes*, vol. 8, no. 9, p. 1123, Sep. 2020.

[9] A. Storkey, "When training and test sets are different: Characterizing learning transfer," *Dataset Shift Mach. Learn.*, vol. 30, pp. 3–28, Feb. 2009.

[10] M. Sugiyama and K.-R. Müller, "Model selection under covariate shift," in *Proc. Int. Conf. Artif. Neural Netw.*, 2005, pp. 235–240.

[11] M. Sugiyama and A. J. Storkey, "Mixture regression for covariate shift," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 19, 2006, pp. 1337–1344.

[12] M. Sugiyama and K.-R. Müller, "Input-dependent estimation of generalization error under covariate shift," *Statist. Decisions*, vol. 23, no. 4, pp. 249–279, Jan. 2005.

[13] H. Shimodaira, "Improving predictive inference under covariate shift by weighting the log-likelihood function," *J. Stat. Planning Inference*, vol. 90, no. 2, pp. 227–244, Oct. 2000.

[14] Y. Tian, J. Wang, Z. Qi, C. Yue, P. Wang, and S. Yoon, "Calibration method for sensor drifting bias in data center cooling system using Bayesian inference coupling with autoencoder," *J. Building Eng.*, vol. 67, May 2023, Art. no. 105961.

[15] T. Guo, X. Tan, L. Yang, Z. Liang, B. Zhang, and L. Zhang, "Domain adaptive subspace transfer model for sensor drift compensation in biologically inspired electronic nose," *Expert Syst. Appl.*, vol. 208, Dec. 2022, Art. no. 118237.

[16] K. C. Rose, C. G. McBride, and V. W. Moriarty, "Creating and managing data from high-frequency environmental sensors," in *Encyclopedia Inland Waters*, 2nd ed., T. Mehner and K. Tockner, Eds. Amsterdam, The Netherlands: Elsevier, 2022, pp. 549–569.

[17] M. Pereira and B. Glisic, "Detection and quantification of temperature sensor drift using probabilistic neural networks," *Expert Syst. Appl.*, vol. 213, Mar. 2023, Art. no. 118884.

[18] J. Fleischer, G. Lanza, M. Schlipf, J. Kotschenreuther, and J. Peters, "Process parameter analysis on surface roughness and process forces in micro cutting," in *Proc. 4M 2nd Int. Conf. Multi-Mater. Micro Manuf.*, W. Menz, S. Dimov, and B. Fillon, Eds. Amsterdam, The Netherlands: Elsevier, 2006, pp. 289–292.

[19] S. Basu and A. K. Debnath, "Intelligent control system," in *Power Plant Instrumentation and Control Handbook*, 2nd ed., S. Basu and A. K. Debnath, Eds. Boston, MA, USA: Academic, 2019, pp. 477–631.

[20] G. Hinton, L. Deng, D. Yu, G. E. Dahl, A.-R. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, T. N. Sainath, and B. Kingsbury, "Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups," *IEEE Signal Process. Mag.*, vol. 29, no. 6, pp. 82–97, Nov. 2012.

[21] J. L. Ba, J. R. Kiros, and G. E. Hinton, "Layer normalization," 2016, *arXiv:1607.06450*.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 5998–6008.

[23] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," *Commun. ACM*, vol. 60, no. 6, pp. 84–90, May 2017.

[24] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: A unified embedding for face recognition and clustering," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2015, pp. 815–823.

[25] J. De Fauw et al., "Clinically applicable deep learning for diagnosis and referral in retinal disease," *Nature Med.*, vol. 24, no. 9, pp. 1342–1350, Sep. 2018.

[26] Z. Chen and X. Huang, "End-to-end learning for lane keeping of self-driving cars," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2017, pp. 1856–1860.

[27] Z. Yang, Y. Zhang, J. Yu, J. Cai, and J. Luo, "End-to-end multi-modal multi-task vehicle control for self-driving cars with visual perceptions," in *Proc. 24th Int. Conf. Pattern Recognit. (ICPR)*, Aug. 2018, pp. 2289–2294.

[28] A. Kendall, J. Hawke, D. Janz, P. Mazur, D. Reda, J.-M. Allen, V.-D. Lam, A. Bewley, and A. Shah, "Learning to drive in a day," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 8248–8254.

[29] C. K. Lau, K. Ghosh, M. A. Hussain, and C. R. Che Hassan, "Fault diagnosis of Tennessee Eastman process with multi-scale PCA and ANFIS," *Chemometric Intell. Lab. Syst.*, vol. 120, pp. 1–14, Jan. 2013.

[30] Balaji Lakshminarayanan, Alexander Pritzel, and Charles Blundell, "Simple and scalable predictive uncertainty estimation using deep ensembles," in *Proc. 31st Int. Conf. Neural Inf. Process. Syst.*, Red Hook, NY, USA: Curran Associates, 2017, pp. 6405–6416.

[31] Y. Gal, "Uncertainty in deep learning," Ph.D. thesis, Dept. Eng., Univ. Cambridge, Cambridge, U.K.:, 2016.

[32] S. Depeweg, J. M. Hernández-Lobato, F. Doshi-Velez, and S. Udluft, "Decomposition of uncertainty in Bayesian deep learning for efficient and risk-sensitive learning," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 1184–1193.

[33] S. Depeweg, J. M. Hernández-Lobato, F. Doshi-Velez, and S. Udluft, "Decomposition of uncertainty for active learning and reliable reinforcement learning in stochastic systems," *Stat*, vol. 1050, p. 11, Nov. 2017.

[34] A. Malinin and M. Gales, "Reverse kl-divergence training of prior networks: Improved uncertainty and adversarial robustness," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 14547–14558.

[35] L. Alzubaidi, J. Zhang, A. J. Humaidi, A. Al-Dujaili, Y. Duan, O. Al-Shamma, J. Santamaría, M. A. Fadhel, M. Al-Amidie, and L. Farhan, "Review of deep learning: Concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 1, pp. 1–74, Mar. 2021.

[36] Y.-X. Wang, D. Ramanan, and M. Hebert, "Learning to model the tail," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 30, 2017, pp. 7029–7039.

[37] X. Zhu, C. Vondrick, C. C. Fowlkes, and D. Ramanan, "Do we need more training data? *Int. J. Comput. Vis.*, vol. 119, no. 1, pp. 76–92, 2016.

[38] A. Malinin and M. Gales, "Predictive uncertainty estimation via prior networks," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 7047–7058.

[39] D. Huseljic, B. Sick, M. Herde, and D. Kottke, "Separation of aleatoric and epistemic uncertainty in deterministic deep neural networks," in *Proc. 25th Int. Conf. Pattern Recognit. (ICPR)*, Jan. 2021, pp. 9172–9179.

[40] S. Liang, Y. Li, and R. Srikant, "Enhancing the reliability of out-of-distribution image detection in neural networks," 2017, *arXiv:1706.02690*.

[41] C. Fefferman, S. Mitter, and H. Narayanan, "Testing the manifold hypothesis," *J. Amer. Math. Soc.*, vol. 29, no. 4, pp. 983–1049, Feb. 2016.

[42] L Cayton, "Algorithms for manifold learning," Univ. California, Oakland, CA, USA, Tech. Rep. cs2008-0923, 2005.
[43] S. Rifai, Y. N. Dauphin, P. Vincent, Y. Bengio, and X. Müller, "The manifold tangent classifier," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, 2011, pp. 2294–2302.
[44] H. Narayanan and S. Mitter, "Sample complexity of testing the manifold hypothesis," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 23, 2010, pp. 1786–1794.
[45] Z. Gao, C. Cecati, and S. X. Ding, "A survey of fault diagnosis and fault-tolerant techniques—Part I: Fault diagnosis with model-based and signal-based approaches," *IEEE Trans. Ind. Electron.*, vol. 62, no. 6, pp. 3757–3767, Jun. 2015.
[46] S. Yin, S. X. Ding, X. Xie, and H. Luo, "A review on basic data-driven approaches for industrial process monitoring," *IEEE Trans. Ind. Electron.*, vol. 61, no. 11, pp. 6418–6428, Nov. 2014.
[47] M. He and D. He, "Deep learning based approach for bearing fault diagnosis," *IEEE Trans. Ind. Appl.*, vol. 53, no. 3, pp. 3057–3065, May 2017.
[48] B. Li, M.-Y. Chow, Y. Tipsuwan, and J. C. Hung, "Neural-network-based motor rolling bearing fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 47, no. 5, pp. 1060–1069, Oct. 2000.
[49] H. Jiang, X. Li, H. Shao, and K. Zhao, "Intelligent fault diagnosis of rolling bearings using an improved deep recurrent neural network," *Meas. Sci. Technol.*, vol. 29, no. 6, Jun. 2018, Art. no. 065107.
[50] L. Wen, X. Li, L. Gao, and Y. Zhang, "A new convolutional neural network-based data-driven fault diagnosis method," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5990–5998, Jul. 2018.
[51] M. Xia, T. Li, L. Xu, L. Liu, and C. W. de Silva, "Fault diagnosis for rotating machinery using multiple sensors and convolutional neural networks," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 1, pp. 101–110, Feb. 2018.
[52] G. Xu, M. Liu, Z. Jiang, W. Shen, and C. Huang, "Online fault diagnosis method based on transfer convolutional neural networks," *IEEE Trans. Instrum. Meas.*, vol. 69, no. 2, pp. 509–520, Feb. 2020.
[53] W. Lu, B. Liang, Y. Cheng, D. Meng, J. Yang, and T. Zhang, "Deep model based domain adaptation for fault diagnosis," *IEEE Trans. Ind. Electron.*, vol. 64, no. 3, pp. 2296–2305, Mar. 2017.
[54] M. Qiao, S. Yan, X. Tang, and C. Xu, "Deep convolutional and LSTM recurrent neural networks for rolling bearing fault diagnosis under strong noises and variable loads," *IEEE Access*, vol. 8, pp. 66257–66269, 2020.
[55] H. Zhao, S. Sun, and B. Jin, "Sequential fault diagnosis based on LSTM neural network," *IEEE Access*, vol. 6, pp. 12929–12939, 2018.
[56] Y. Zhang, T. Zhou, X. Huang, L. Cao, and Q. Zhou, "Fault diagnosis of rotating machinery based on recurrent neural networks," *Measurement*, vol. 171, Feb. 2021, Art. no. 108774.
[57] M. Zhao, S. Zhong, X. Fu, B. Tang, and M. Pecht, "Deep residual shrinkage networks for fault diagnosis," *IEEE Trans. Ind. Informat.*, vol. 16, no. 7, pp. 4681–4690, Jul. 2020.
[58] F. Jia, Y. Lei, J. Lin, X. Zhou, and N. Lu, "Deep neural networks: A promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data," *Mech. Syst. Signal Process.*, vols. 72–73, pp. 303–315, May 2016.
[59] C. Guo, G. Pleiss, Y. Sun, and K. Q. Weinberger, "On calibration of modern neural networks," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 1321–1330.
[60] H. Inoue, "Data augmentation by pairing samples for images classification," 2018, *arXiv:1801.02929*.
[61] H. Zhang, M. Cissé, Y. N. Dauphin, and D. Lopez-Paz, "mixup: Beyond empirical risk minimization," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, Apr. 2018. [Online]. Available: https://openreview.net/group?id=ICLR.cc/2018/Conference#accepted-poster-papers
[62] Y. Tokozume, Y. Ushiku, and T. Harada, "Learning from between-class examples for deep sound recognition," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, Apr. Jan. 2018. [Online]. Available: https://openreview.net/group?id=ICLR.cc/2018/Conference#accepted-poster-papers
[63] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *Proc. Adv. Neural Inf. Process. Syst., 26th Annu. Conf. Neural Inf. Process. Syst.*, P. L. Bartlett, F. C. N. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds. Lake Tahoe, NV, USA, 2012, pp. 1106–1114.
[64] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, "ImageNet large scale visual recognition challenge," *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, Dec. 2015.
[65] K. Lee, H. Lee, K. Lee, and J. Shin, "Training confidence-calibrated classifiers for detecting out-of-distribution samples," in *Proc. 6th Int. Conf. Learn. Represent. (ICLR)*, Vancouver, BC, Canada, Apr. 2018. [Online]. Available: https://openreview.net/group?id=ICLR.cc/2018/Conference#accepted-poster-papers
[66] I. Goodfellow, "Generative adversarial networks," *Commun. ACM*, vol. 63, no. 11, pp. 139–144, 2020.
[67] S. Vernekar, A. Gaurav, V. Abdelzad, T. Denouden, R. Salay, and K. Czarnecki, "Out-of-distribution detection in classifiers via generation," 2019, *arXiv:1910.04241*.
[68] K. Sohn, H. Lee, and X. Yan, "Learning structured output representation using deep conditional generative models," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 28, 2015, pp. 3483–3491.
[69] F. Möller, D. Botache, D. Huseljic, F. Heidecker, M. Bieshaar, and B. Sick, "Out-of-distribution detection and generation using soft Brownian offset sampling and autoencoders," in *Proc. IEEE/CVF Conf. Comput. Vis. Pattern Recognit. Workshops (CVPRW)*, Jun. 2021, pp. 46–55.
[70] C. Matthews, J. Weare, A. Kravtsov, and E. Jennings, "Umbrella sampling: A powerful method to sample tails of distributions," *Monthly Notices Roy. Astronomical Soc.*, vol. 480, no. 3, pp. 4069–4079, Nov. 2018.
[71] I. M. Johnstone and D. M. Titterington, "Statistical challenges of high-dimensional data," *Philos. Trans. Roy. Soc. A, Math., Phys. Eng. Sci.*, vol. 367, no. 906, pp. 4237–4253, 2009.
[72] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 35, no. 8, pp. 1798–1828, Aug. 2013.
[73] R. T. Chen, X. Li, R. B. Grosse, and D. K. Duvenaud, "Isolating sources of disentanglement in variational autoencoders," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 2615–2625.
[74] S. Vernekar, A. Gaurav, V. Abdelzad, T. Denouden, R. Salay, and K. Czarnecki, "Out-of-distribution detection in classifiers via generation," in *Proc. 33rd Conf. Neural Inf. Process. Syst. (NeurIPS), Saf. Robustness Decis. Making Workshop*, 2019. [Online]. Available: https://arxiv.org/abs/1910.04241
[75] P. J. Rousseeuw, "Multivariate estimation with high breakdown point," *Math. Statist. Appl.*, vol. 8, nos. 283–297, p. 37, 1985.
[76] P. J. Rousseeuw, "Least median of squares regression," *J. Amer. Statist. Assoc.*, vol. 79, no. 388, pp. 871–880, 1984.
[77] M. Hubert and M. Debruyne, "Minimum covariance determinant," *WIREs Comput. Statist.*, vol. 2, no. 1, pp. 36–43, Jan. 2010.
[78] D. Dua and C. Graff, "UCI machine learning repository: Steel plates faults data set," School Inf. Comput. Sci., Univ. California, Irvine, CA, 2017.
[79] L. Yang, X. Huang, Y. Ren, and Y. Huang, "Steel plate surface defect detection based on dataset enhancement and lightweight convolution neural network," *Machines*, vol. 10, no. 7, p. 523, Jun. 2022.
[80] N. Chen, J. Sun, X. Wang, Y. Huang, Y. Li, and C. Guo, "Research on surface defect detection and grinding path planning of steel plate based on machine vision," in *Proc. 14th IEEE Conf. Ind. Electron. Appl. (ICIEA)*, Jun. 2019, pp. 1748–1753.
[81] D. P. Kingma and J. L. Ba, "Adam: A method for stochastic optimization," in *Proc. 3rd Int. Conf. Learn. Represent. (ICLR) Conf. Track*, 2015, pp. 1–15.
[82] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, and R. Kern, "Array programming with numpy," *Nature*, vol. 585, no. 7825, pp. 357–362, 2020.
[83] V. Nair and G. E. Hinton, "Rectified linear units improve restricted Boltzmann machines," in *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, Haifa, Israel, 2010, pp. 807–814.
[84] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, vol. 37, Jul. 2015, pp. 448–456.
[85] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A simple way to prevent neural networks from overfitting," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, Jan. 2014.
[86] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly Weather Rev.*, vol. 78, no. 1, pp. 1–3, Jan. 1950.
[87] D. Zhu, X. Wu, and T. Yang, "Benchmarking deep AUROC optimization: Loss functions and algorithmic choices," 2022, *arXiv:2203.14177*.
[88] T. Fawcett, "An introduction to ROC analysis," *Pattern Recognit. Lett.*, vol. 27, no. 8, pp. 861–874, Jun. 2006.
[89] B. Charpentier, D. Zügner, and S. Günnemann, "Posterior network: Uncertainty estimation without ood samples via density-based pseudo-counts," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, 2020, pp. 1356–1367.

JEFKINE KAFUNAH received the B.Sc. degree in actuarial science from the Jomo Kenyatta University of Agriculture and Technology, Kenya, in 2008. He is currently pursuing the Ph.D. degree in computer science with the College of Science and Engineering, University of Galway, Ireland. His research interests include data analytics, data-driven process monitoring, fault diagnosis and prognosis, industrial cyber-physical systems, and AI/ML.

PRIYANKA VERMA (Senior Member, IEEE) received the Ph.D. degree from the Atal Bihari Vajpayee-Indian Institute of Information Technology and Management, India. She completed the Leadership and Innovation Program from the Massachusetts Institute of Technology, USA. She was an Assistant Professor with the Maulana Azad National Institute of Technology Bhopal. She is currently a Marie Skłodowska-Curie Postdoctoral Research Fellow with the University of Galway, where she is also an Adjunct Lecturer. She has been a Visiting Research Scholar with Anglia Ruskin University, Chelmsford, U.K. She has coauthored many publications in updated journals and conferences. Her research interests include cyber security, the IIoT, smart manufacturing, AI/ML, cloud, and edge computing. She is one of the Brand Ambassador for promoting cyber security in Ireland. She received many awards on national and international levels. She is a conference speaker and has given expert lectures in many countries.

MUHAMMAD INTIZAR ALI received the Ph.D. degree (Hons.) in database and artificial intelligence from the School of Informatics, Vienna University of Technology, Austria, in 2011. He is currently an Assistant Professor with the School of Electronic Engineering, Dublin City University. He is actively involved in various EU-funded and industry-funded projects aimed at providing the IoT-enabled adaptive intelligence for smart applications. His research interests include semantic web, data analytics, the internet of things (IoT), linked data, federated query processing, stream query processing, and optimal query processing over large-scale distributed data sources. He is a PC member of various journals, international conferences, and workshops.

JOHN G. BRESLIN (Senior Member, IEEE) received the bachelor's and Ph.D. degrees in electronic engineering from the University of Galway, in 1994 and 2002, respectively. He is a Professor of electronic engineering with the University of Galway, where he is also the Director of the TechInnovate and AgInnovate Entrepreneurship Programs. Associated with two SFI Research Centres, he is a Co-Principal Investigator with Insight (Data Analytics) and a Funded Investigator with VistaMilk (AgTech). He has coauthored around 300 publications, including the books ''The Social Semantic Web,'' the ''Social Semantic Web Mining,'' and the ''Old Ireland in Colour'' Trilogy. He co-created the SIOC framework, implemented in hundreds of applications (by Yahoo, Boeing, Vodafone, etc.) on at least 65,000 websites with 35 million data instances. He is the Co-Founder of the PorterShed, boards.ie, and adverts.ie.