CycleGAN-Based Data Augmentation for Enhanced Remaining Useful Life Prediction under Unsupervised Domain Adaptation

Dorian Joubaud [1], Evgeny Zotov[1], Oguz Bektas[1], Sylvain Kubler [1], Yves LeTraon[1]

[1] *Interdisciplinary Centre for Security, Reliability and Trust (SnT), University of Luxembourg, Luxembourg*
dorian.joubaud@uni.lu, evgeny.zotov@uni.lu, oguz.bektas@uni.lu,
sylvain.kubler@uni.lu, yves.letraon@uni.lu

Abstract

Predictive maintenance is crucial for enhancing operational efficiency and reducing costs in Prognostics and Health Management (PHM). One of the key tasks in predictive maintenance is the estimation of Remaining Useful Life (RUL) of machinery. In practice, the data for different machines is not always accessible in sufficient quantity or quality, therefore the machine learning models trained on machines in one domain often perform poorly when applied to other domains due to covariate shifts. As a solution, Domain Adaptation (DA) aims to tackle domain shifts by extracting domain-invariant features. However, traditional methods often fail to adequately address the complexity and variability of real-world data. We propose to address this challenge, using a Wasserstein CycleGAN with Gradient Penalty (W-CycleGAN-GP) to learn mappings between domains and generate augmented data in the target domain from data in the source domain. We use our approach to generate realistic augmented data that bridge domain gap coupled with recent work on adversarial-based and correlation alignment-based DA models to improve the performance of RUL prediction models in target domains without having access to labeled data. The experimental results on the C-MAPSS dataset demonstrate a significant improvement in the RUL prediction score and accuracy within the target domain.

1. Introduction

The demand for reliability in complex systems has led to significant advances in Prognostics and Health Management (PHM). In this context, accurately estimating the Remaining Useful Life (RUL) of systems and their components is essential for robust predictive maintenance strategies. RUL prediction enables maintenance decisions that improve operational efficiency and reduce costs.

Machine learning models have been widely adopted for RUL prediction due to their ability to learn complex patterns from historical data. Data-driven models, in particular, offer advanced frameworks for RUL estimation. They can learn from historical data and identify patterns and features associated with system degradation. These models can effectively handle non-linear dynamic systems and provide high-accuracy predictions. However, there are still challenges persisting such as multidimensional data and the need for extensive datasets in different domains. These require further developments that can provide advanced ML techniques to improve prediction accuracy and system reliability.Also, in practice, the data available for different machines is often not accessible in sufficient quantity or quality. Additionally, there can be significant variations between the operating conditions or failure modes of different machines, even when they are of the same type. These variations lead to covariate shifts, where the distribution of the data in the source domain (where the model is trained) differs from that in the target domain (where the model is applied). As a result, machine learning models trained on data from one domain often perform poorly when applied to other domains.

Domain Adaptation (DA) techniques have been developed to address the challenge of covariate shifts by extracting domain-invariant features. The goal of DA is to learn a representation that is robust to changes in the data distribution between the source and target domains. The adversarial training enable the model to learn representations that are indistinguishable between the source and target domains. Despite progress in DA research, traditional methods often do not adequately address the complexity and variability of real-world data, particularly in the context of RUL prediction.

To address these challenges, we propose applying DA for data augmentation. Using a Wasserstein CycleGAN with Gradient Penalty (W-CycleGAN-GP), our objective is to learn mappings between the domains and generate augmented data in the target domain from data in the source domain. Taking advantage of recent advances in adversarial DA models in RUL, we augment the target domain dataset with realistic synthetic data. Our combined approach allows for improved performance of RUL prediction models in target domains without requiring access to labeled data in the target domain.

We validate our approach on the C-MAPSS dataset, a standard benchmark for RUL prediction tasks. The experimental results demonstrate a significant improvement in RUL prediction scores and accuracy within the target domain compared to unaugmented prediction, showcasing the effectiveness of our method in practical applications.

The rest of this paper is organized as follows: Section 2 reviews related work domain adaptation, data augmentation and RUL prediction. Section 3 describes the proposed W-CycleGAN-GP approach in detail, including the architecture and training procedure. Section 4 presents the experimental setup and results, while Section 5 discusses the findings and their implications. Finally, Section 6 concludes the paper and suggests directions for future research.

2. RELATED WORK

2.1. Domain Adaptation

Domain adaptation (DA) is a sub-field of transfer learning that aims to improve the performance of a model on a target domain in the presence of covariate shifts between source and target domains. DA has been applied to many domains, notably on structural health monitoring (SHM) where an ML model is trained on numerical simulation and tested on experimental data (Ozdagli & Koutsoukos, 2020). Unsupervised domain adaptation (UDA) is a specific type of DA where the target labels are not available during the training procedure. UDA is particularly relevant in RUL prediction, where labelled failure data some domains is unavailable or is prohibitively expensive and time-consuming to collect.

Initial works considering the problem are based on aligning data distributions in the source and target domains. (Long, Wang, Ding, Sun, & Yu, 2013), (Long, Wang, Ding, Sun, & Yu, 2014), and (Long, Cao, Wang, & Jordan, 2018) use maximum mean discrepancy to reduce differences in the marginal and/or conditional distributions. Correlation Alignment (CORAL) (Sun, Feng, & Saenko, 2017) minimizes domain shifts by aligning the second-order statistics (covariance matrices) of source and target distributions. With the success of deep neural network architectures and their ability to capture deep feature representations, adaptations of existing methods and the development of new approaches aiming to capture domain-invariant features from source and target data have been developed. (Sun & Saenko, 2016) adapted CORAL within a deep convolutional neural network (CNN) to learn a nonlinear transformation that aligns correlations of layer activations. (Ganin et al., 2016) paved the way to promising research with domain adversarial neural network (DANN). They use a domain discriminator with a gradient reversal layer (GRL) to make features from both domains indistinguishable, thereby learning domain-invariant features. Adversarial Discriminative Domain Adaptation (ADDA) separates the feature extraction and adversarial training phases, allowing for more flexible and stable training. ADDA aligns the feature distributions of the source and target domains by employing a discriminator that distinguishes between them (Tzeng, Hoffman, Saenko, & Darrell, 2017).

Most DA approaches have been developed for classification tasks over computer vision data. Thus, these techniques need to be adapted for regression tasks and for sequential data, which are prevalent in the PHM context. Recent works aim to adapt DA techniques for RUL prediction. (da Costa, Akçay, Zhang, & Kaymak, 2020) use long-short term memory (LSTM) networks and adversarial training from DANN to extract domain-invariant features that can be used to predict RUL in the target domain. (Hu et al., 2022) introduce the Wasserstein distance-based weighted domain adversarial neural network (WD-WDANN) to handle RUL DA with different working conditions. Their approach only considers high-quality data in terms of transferability to avoid outliers that could penalize feature alignment. (Li, Li, Zuo, Zhu, & Shen, 2022) propose to push DA even further by aligning distributions both at the semantic and at the feature level using attention-based backbone architecture, which shows high quality results. Recently, (Nejjar, Geissmann, Zhao, Taal, & Fink, 2024) propose two novel adversarial-based DA approaches that adapt DANN to take into account the different operational conditions (e.g. operating speed, working temperature or environmental noise) that might be unequally represented in the source and target domain.

2.2. Data Augmentation

Data augmentation is a very popular technique to tackle data scarcity or as a regularisation technique in machine learning (ML). It consists to generate synthetic data from existing data. Generative adversarial network (GAN) is a very popular technique to perform data augmentation. This builds on a competitive minimax game between two neural networks: the generator, which learns to create artificial data samples, and the discriminator, which learns to detect fake data samples (Goodfellow et al., 2020). The distribution of data is generated by the generator network; therefore, it can provide a representation of the true data that can be directly used for simulating the underlying process. In the several years since its inception, the approach has been expanded in multiple directions. Research on different neural network architectures for both the generator and discriminator has introduced the deep convolutional model DCGAN, which is employed for unsupervised image feature extraction and picture generation (Radford, Metz, & Chintala, 2015). The absence of control over the outputs of the original GAN prompted researchers to develop methods for conditioning GAN models with additional inputs or outputs. Originally introduced in Conditional GAN (Mirza & Osindero, 2014), where image generation was effectively linked to input labels through supervised learning, subsequent research demonstrated an unsupervised conditional training method grounded in information theory with InfoGAN (Chen et al., 2016).

In recent times, GANs have imptoved the state of the art across various domains. For instance, in neural audio synthesis, WaveGAN has adapted the DCGAN model mentioned above for audio sequence generation (Donahue, McAuley, & Puckette, 2018), and GANSynth has been developed for the generation of high-quality audio spectrograms (Engel et al., 2019). A most notable breakthrough in simulation of realistic high-resolution (1024x1024) images of human faces has been succeded by the StyleGAN architecture that has adapted the style transfer methods for the GAN's generator network (Karras, Laine, & Aila, 2019). While most of GAN studies have focused on image-generation, the generation of time-domain signals with GANs is still a relatively unexplored area (Zotov & Kadirkamanathan, 2021).

In the context of UDA, data augmentation can also be used as with (Hsu, Zhang, & Glass, 2017) who developed an augmentation-based method do generate labelled data in the target domain for speech recognition suing a Variable Auto-Encoder (VAE) trained in an unsupervised way. (Volpi, Morerio, Savarese, & Murino, 2018) leverage the strength of Conditional-GAN to perform data augmentation in the feature space. In the same way (Wang, Meng, & Breckon, 2023) use feature augmentation for image classification to perform selective pseudo-labeling with a *norm-AE* generative model. Their objective is to take advantage of unlabeled data from target domain to learn a unified classifier for both source and target data. Taking advantage from GAN architecture, (Palladino, Slezak, & Ferrante, 2020) use Cycle-consistency GAN (CycleGAN) on magnetic resonance images to maintain distribution to generate relevant data and avoid distribution shift between two data domain. (Iacono & Khan, 2023) introduce a structure preserving CycleGAN which is able to maintain the structure of medical data when generating synthetic data. Both discussed approach show promising results and an increase of performance in the UDA approach studied.

In the context of PHM, time-series data extracted from sensors are prevalent. However, fewer works on CycleGAN and time-series can be found in the literature. (Saravanan, Luo, & Van Ngo, 2023) apply CycleGAN to time-series by transforming time-series into 2D images for anomaly detection. (Schockaert & Hoyez, 2020) develop a CycleGAN for multivariate time-series on an artificial blast furnace dataset. Another variant of CycleGAN uses Wasserstein distance to stabilise the training, (Luleci, Catbas, & Avci, 2023) for fault diagnosis bearing data generation. Finally, (Pu, Cabrera, Li, & de Oliveira, 2023) proposed an updated version using the sliced Wasserstein distance (SWD). SWD estimates the difference between two distribution with a lower computational cost compared to the Wasserstein distance by sorting 1D random projection on the projection sphere. They adapt their approach for both conditional and unconditional CycleGAN. They test their approach on computer vision data (MNIST dataset) and on industrial time-series data (robot fault diagnosis).

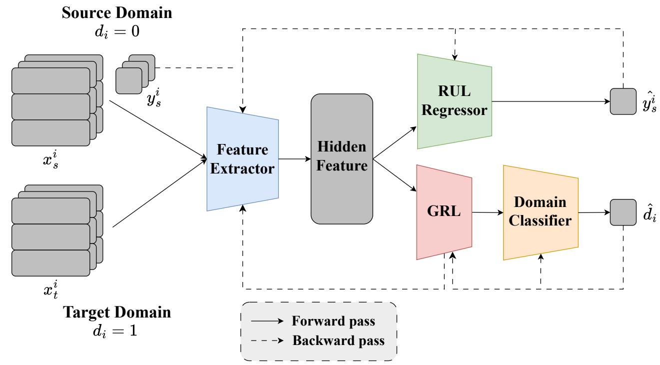


Figure 1. DANN architecture flowchart inspired from the work of (Nejjar et al., 2024). Each domain data are encoded into hidden features using feature extractor. Regressor predict the RUL for the source data. Domain classifier coupled with Gradient Reversal Layer (GRL) ensure that feature from both domains are indistinguishable.

3. DATA AUGMENTATION FOR UNSUPERVISED DOMAIN ADAPTATION

In this paper, we study the application of unsupervised domain adaptation (UDA) in the case of RUL prediction in PHM. This section presents the problem and the notations.

3.1. Problem Definition

Consider a source domain $\mathcal{S} = \{(\mathbf{x_s^i}, y_s^i)\}_{i=1}^{n_s}$ where $\mathbf{x_s^i} \in \mathcal{X}_\mathcal{S}$ denotes a multivariate time-series data collected from sensors and $y_s^i \in \mathcal{Y}_\mathcal{S}$ the remaining useful life continuous label and n_s is the number of available data. Similarly, a target domain $\mathcal{T} = \{(\mathbf{x_t^i}, y_t^i)\}_{i=1}^{n_t}$ where $\mathbf{x_t^i} \in \mathcal{X}_\mathcal{T}$ and $y_t^i \in \mathcal{Y}_\mathcal{T}$. We assume a covariate shift between domains where source and target share the same conditional distribution $p_s(y_s|\mathbf{x_s}) = p_t(y_t|\mathbf{x_t})$ but from different marginal distributions $p_s(\mathbf{x_s}) \neq p_t(\mathbf{x_t})$. Our objective is to learn a mapping function f^* that approximates the degradation of the target domain from the target domain sensor data $f^*(\mathbf{x_t^i}) \approx y_t^i$. During the learning phase, the training data only contain source data and labels $\{(\mathbf{x_s^i}, y_s^i)\}_{i=1}^{n_s}$ and target data $\{(\mathbf{x_t^i})\}_{i=1}^{n_t}$. This configuration is called unsupervised domain adaptation (UDA).

3.2. Unsupervised Domain Adaptation

To handle a configuration where no labels are available in the target domain, we leverage both traditional work with CORelation ALignment (CORAL) and recent work with Domain Adversarial Neural Networks (DANN).

3.2.1. Domain Adversarial Neural Network

Studies by (da Costa et al., 2020) and (Nejjar et al., 2024) combine DANN with LSTM and CNN to align deep features and perform domain adaptation between different machines for Remaining Useful Life (RUL) prediction. DANN aims to align the distributions of both domains while learning to predict the RUL of the source domain. It is composed of three parts: (i) a feature extractor, (ii) a regressor, and (iii) a domain classifier.

(i) Each input data point $x^i \in \mathcal{X}$ is given a label $d_i \in \{0, 1\}$ that represents its input domain. The feature extractor f_e takes input data $x^i \in \mathcal{X}$ belonging to one domain and extracts hidden features $h_i = f_e(x^i)$ from the input data.

(ii) The regressor R takes the hidden features h_i extracted from the data that belong to the source domain and predicts a singleton that estimates the RUL of the given input sequence $\hat{y^i} = R(h_i)$. The root mean square error (RMSE) is used as a loss to minimize the error between the actual RUL y^i and the predicted RUL $\hat{y^i}$.

(iii) Given the hidden features h_i, the discriminator D_d estimates the domain d_i to which the original input data belong $\hat{d_i} = D_d(h_i)$. Binary cross-entropy (BCE) is used as a loss to minimize the error between the actual domain d_i and the predicted domain $\hat{d_i}$:

$$BCE = -(d_i \log(\hat{d_i}) + (1 - d_i) \log(1 - \hat{d_i}))$$

While D learns to minimize the classification loss, f_e learns to maximize this loss thanks to the gradient reversal layer (GRL) that reverses the sign of the gradient during backward propagation. These opposite objectives describe an adversarial learning scenario where the objective for the feature extractor is to learn to extract a representation of the data such that data from the source and target domains are indistinguishable. Figure 1 show the different part of DANN approach.

3.2.2. CORelation ALignment

CORAL (Sun et al., 2017) is a simple and effective method for UDA that aims to transform the source domain features to match target domain's feature distribution. The transformation matrix is computed to minimize the Frobenius norm between the covariance matrices of the source and target domains, thus aligning their second-order statistics. In their work (da Costa et al., 2020) used a simple deep neural network on top of the aligned space to predict the RUL of both source and target domains. The model is trained using labels from the source domain and evaluated on the target domains.

3.3. CycleGAN

CycleGAN is an extension of GAN that aims to learn two mappings between two domains, $\mathcal{S}$ and $\mathcal{T}$. It relies on two adversarial neural network architectures, each composed of one generator and one discriminator, which compete against each other. The objective is to learn two generator models: $\mathcal{G}_{st} : \mathcal{S} \to \mathcal{T}$ and $\mathcal{G}_{ts} : \mathcal{T} \to \mathcal{S}$. These generators translate data from the source domain to the target domain and vice versa, while competing against adversarial discriminators $D_{\mathcal{S}}$ and $D_{\mathcal{T}}$. The discriminator models are trained to distinguish between real data from their respective domains and fake data generated by the generators.

During training, real data $\mathbf{x_s}$ from domain $\mathcal{S}$ is fed into $\mathcal{G}_{st}$ to generate fake data $\tilde{\mathbf{x}}_\mathbf{t}$ in domain $\mathcal{T}$:

$$\tilde{\mathbf{x}}_\mathbf{t} = \mathcal{G}_{st}(\mathbf{x_s}). \tag{1}$$

This fake data is then compared to real data from domain $\mathcal{T}$ by $D_{\mathcal{T}}$. Simultaneously, the fake data generated in domain $\mathcal{T}$ is passed through $\mathcal{G}_{ts}$ to reconstruct data $\approx{\mathbf{x}}_\mathbf{s}$ in the original domain $\mathcal{S}$.

$$\overset{\approx}{\mathbf{x}}_\mathbf{s} = \mathcal{G}_{ts}(\tilde{x}_t). \tag{2}$$

Similarly, $\mathcal{G}_{ts}$ takes data from domain $\mathcal{T}$ and generates data in domain $\mathcal{S}$. The adversarial training process involves comparing this generated data to real data from domain $\mathcal{S}$ using $D_{\mathcal{S}}$. The fake data is then re-transformed back to the original domain $\mathcal{T}$ using $\mathcal{G}_{st}$. The training objectives of CycleGAN contain the following losses.

Adversarial Losses: We leverage from the work of (Arjovsky & Bottou, 2017) who introduce the Wasserstein GAN (WGAN) that use the Wasserstein-1 distance and clip the weight for the discriminator optimization to stabilize the training.

$$\begin{aligned}\mathcal{L}_{\text{WGAN}}\left(G_{st}, D_t, \mathcal{S}, \mathcal{T}\right) = &\ \mathbb{E}_{x_t \sim p_t(x_t)}\left[D_{\mathcal{T}}(x_t)\right] \\ &- \mathbb{E}_{x_s \sim p_s(x_s)}\left[D_{\mathcal{T}}(G_{st}(x_s)\right],\end{aligned} \tag{3}$$

$$\begin{aligned}\mathcal{L}_{\text{WGAN}}\left(G_{ts}, D_s, \mathcal{S}, \mathcal{T}\right) = &\ \mathbb{E}_{x_s \sim p_s(x_s)}\left[D_{\mathcal{S}}(x_s)\right] \\ &- \mathbb{E}_{x_t \sim p_t(x_t)}\left[D_{\mathcal{S}}(G_{ts}(x_t)\right].\end{aligned} \tag{4}$$

Gradient Penalty Loss: To replace the weight clipping (Gulrajani, Ahmed, Arjovsky, Dumoulin, & Courville, 2017) introduce gradient penalty to WGAN. It consists of classifying uniformly samples along straight lines between pairs of points sampled from the real data distribution and fake gen-

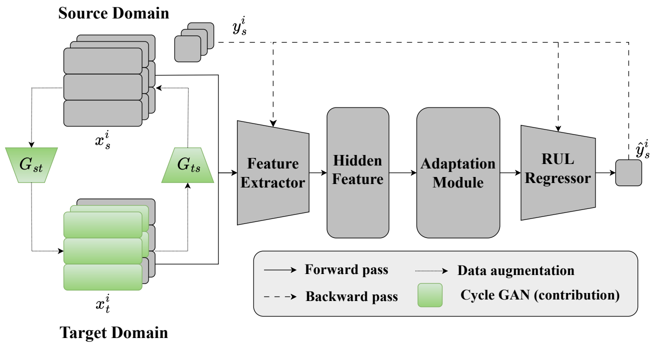


Figure 2. Proposed methodology flowchart inpired from (Nejjar et al., 2024). Target domain is augmented using CycleGAN (green). Each domain are encoded into hidden features using feature extractor. Regressor predicts the RUL for the source data. Domain adaptation module (such as DANN or CORAL) ensures that features of both domains are indistinguishable.

erated data distribution.

$$\mathcal{L}_{\text{GP}}\left(D_t, D_s\right) = \mathbb{E}_{\hat{x}_t \sim p_t(\hat{\boldsymbol{x}}_t)}\left[\left(\left\|\nabla_{\hat{x}_t} D_t\left(\hat{x}_t\right)\right\|_2 - 1\right)^2\right] + \mathbb{E}_{\hat{x}_s \sim p_s(\hat{\boldsymbol{x}}_s)}\left[\left(\left\|\nabla_{\hat{x}_s} D_s\left(\hat{x}_s\right)\right\|_2 - 1\right)^2\right]. \tag{5}$$

where p_s and p_t are the sampling distribution that uniformly samples along straight lines between pairs of points sampled from the real data distribution and the fake generated data distribution from source and target respectively.

Cycle Consistency Loss: Is one of the key component of CycleGAN architecture. The cycle consistency loss ensures that generated data from one domain can be recovered to its initial domain.

$$\mathcal{L}_{\text{cycle}}(G_{ts}, G_{st}) = \mathbb{E}_{x_s \sim p_s(x_s)}\left[\|G_{ts}(G_{st}(x_s)) - x_s\|_1\right] + \mathbb{E}_{x_t \sim p_t(x_t)}\left[\|G_{st}(G_{ts}(x_t)) - x_t\|_1\right]. \tag{6}$$

Identity Losses: Encourages the mappings to preserve inherent characteristics of the input data in the case where the data are close the respective output distribution. In other words, it act as regularisation for the CycleGAN enforcing $G_{st}(x_t) \approx x_t$ and $G_{ts}(x_s) \approx x_s$.

$$\mathcal{L}_{\text{cycle}}(G_{ts}, G_{st}) = \mathbb{E}_{x_t \sim p_t(x_t)}\left[\|G_{st}(x_t) - x_t\|_1\right] + \mathbb{E}_{x_s \sim p_s(x_s)}\left[\|G_{ts}(x_s) - x_s\|_1\right]. \tag{7}$$

3.4. Proposed approach

The proposed approach of this paper aims to leverage CycleGAN architecture to generate augmented target data from source data to overcome potential data scarcity in the target domain. The idea is to learn a transformation that transform source sensors time-series to target sensors time-series to complete the target domain.

The augmented dataset is then fed into a domain adaptation DA RUL prediction model. This model is designed to estimate the RUL in the target domain without having access to labeled data. By incorporating augmented data, we expect the model to overcome limitations posed by the initial dataset's scarcity, leading to improved predictive score. Figure 2 show a summary of our methodology

4. DATASETS

To evaluate our approach, we use the NASA Commercial Modular Aero-Propulsion System Simulation (C-MAPSS) datasets containing turbofan engine degradation data (Saxena & Goebel, 2008). It is composed of four datasets that include information from 24 sensors with different operating conditions and fault types. Table 1 gather the information of each dataset. These datasets are a key point in the literature for the development and testing of prognostic algorithms for aircraft engine health management and have been case studies for many studies. The signals in the data consist of multivariate and multidimensional time series data. These include sensor measurements and three operational settings: altitude, Mach number, and sea level temperature.

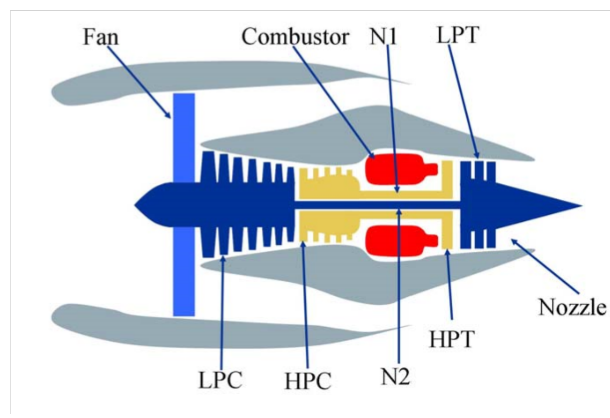


Figure 3. Simplified C-MAPSS diagram (Saxena et al., 2008)

The datasets are divided into training and testing trajectories for RUL predictions. Training trajectories is formed of run-to-failure data for developing lifetime prediction algorithms. On the other hand, test trajectories were cut before failure to validate these algorithms. How the test trajectories will behave will be predicted by the developers. The sub data setd includes different operational regimes and failure modes. Noise and initial wear-performace levels are included to better represent the real-life operational conditons and uncertainties (Saxena & Goebel, 2008).

The C-MAPSS simulation includes a fan speed controller, high-limit regulators, limiting regulators, acceleration and deceleration limiters, and a power management system. All are integrated similar to real engine controllers (see Figure 3). The atmospheric model in the simulation can operate at altitudes up to 40,000 ft, Mach numbers up to 0.90, and temperatures from -60 to 103°F.

Each data set contains various parameters recorded at different operating settings and regimes, such as total temperature at the fan inlet, pressure at the fan inlet, physical fan speed, and motor pressure ratio. While some datasets FD002 and FD004 contain more than one operational regime, FD001 and FD003 have a single operational status.

Data	FD001	FD002	FD003	FD004
Engines: Train	100	260	100	249
Engines: Test	100	259	100	248
Op. Conditions	1	6	1	6
Fault Modes	1	1	2	2

Table 1. C-MAPSS datasets descriptions (Saxena & Goebel, 2008)

To setup our experiment, we employ one dataset as source and another as target. We expect that datasets with more training data and a more operational conditions (FD002 and FD004) will generalize better to target domains compared to datasets with fewer data and fewer operational conditions (FD001 and FD003). In fact, when datasets with more operational conditions are used as unlabeled target domains, it can present challenges for adversarial learning, making it harder for the model to adapt effectively.

5. EXPERIMENTS

5.1. Experimental Settings

The training procedure involves selecting a source and a target domain. In this study, we used all C-CMAPSS dataset (FD001, FD002, FD003, FD004) dataset. The degradation trajectories are multivariate time-series measurement from 21 sensors and 3 operational settings. Among the selected dataset, 7 sensors have constant value. Since the constant values are different depending on the dataset, we keep the constant sensors to better learn the intrinsic characteristics of source and target. All sensor time-series are pre-processed as well as the RUL values. Each dataset is normalised in $[-1; 1]$ using z-normalisation. We employ sliding windows of length 50 on all trajectory time series data to generate source and target domain observation. Similarly to (Li et al., 2022), we employ a piece-wise linear degradation with initial constant values of 130 cycles before the linear degradation to model the RUL in the datasets.

5.2. Model Architecture

In this paper, we study the benefit of using CycleGAN augmentation for unlabeled data in the target domain. To this end we assess the performance of two state of the art UDA model (DANN and CORAL) for RUL prediction with and without augmentation.

In particular, the DANN model configuration is a simple deep neural network architecture to ensure fair comparison. The feature extractor is composed of three 1D convolutional layers with 128, 64, and 32 neurons, using ReLU activation. Each layer has a filter size of ten with a stride of 1. The discriminator contains three fully connected layers with 128, 64, and 1 neuron, as well as one dropout layer with a probability of 0.1. It uses ReLU activation except for the last layer, where we use sigmoid activation to perform domain prediction.

The regressor is used in both DANN and CORAL configurations. It operates on the output from the DANN feature extractor and on the transformed source space produced by CORAL. It contains three fully connected layers with 128, 64, and 1 neuron, using ReLU activation. During the training batch size is set at 256 when datasets FD001 or FD003 is used either as source or target, else batch size is set at 512.

The CycleGAN architecture used is based on the work of (Pu et al., 2023). The generator consists of five 1D convolutional layers, with a residual block inserted after every two convolutional layers. This is followed by five transposed convolutional layers, each also followed by a residual block every two layers. To enhance the diversity of the generated data, a learnable parameter is introduced in the transposed convolution block. This parameter scales a random noise added each time a transposed convolution is executed, promoting more varied outputs from the generator. The discriminator, on the other hand, comprises five stacked convolutional layers. The output of these layers is flattened and fed into a fully connected layer with a sigmoid activation function to perform domain classification.

5.3. Performance Metrics

To estimate the performance of our model, we employ both commonly used metrics for RUL prediction: (i) the Root Mean Square Error (RMSE)

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\hat{y^i} - y^i)^2} \tag{8}$$

and (ii) a non-symmetrical scoring function that penalize over estimation:

$$Score = \sum_{i=1}^{n} e^{\alpha(|\hat{y^i} - y^i|)} \tag{9}$$

where

$$\alpha = \begin{cases} 1/10 & \text{if} \quad \hat{y^i} - y^i > 0 \\ 1/13 & \text{else} \end{cases} \tag{10}$$

and $\hat{y^i}$ is the predicted RUL and y^i the actual RUL for a given window trajectory

To measure the domain shift between domains, we employ the Wasserstein distance (or Kantorovich–Rubinstein metric) that measures the amount of information required to transform one distribution into another.

$$W(\mathcal{S}, \mathcal{T}) = \inf_{\gamma \in \Pi(\mathcal{S}, \mathcal{T})} \int_{\Omega^2} D(x_s, x_t)\, d\gamma(x_s, x_t) \tag{11}$$

where $\Pi(\mathcal{S}, \mathcal{T})$ is the joint distribution of source and target, D a distance and $\gamma(x_s, x_t)$ represents the amount of "information" transported from x_s in $\mathcal{S}$ to $x_t e$ in $\mathcal{T}$.

Using this metric, we can estimate the domain shift between the original target domain data and the augmented target data generated from the source, providing an estimation of the quality of the augmented data.

5.4. Experimental Results

5.4.1. Data Augmentation Quality Evaluation

To evaluate the quality of our augmented data we use the Wasserstein distance to evaluate domain shift between the original target data and the augmented one. Measure betwen source/target and target/target augmented can be found in Table 2.

The domain shift between domains with the same number of operational conditions is minimal (e.g., FD002/FD004: 0.01 and FD001/FD003: 0.04), which makes it challenging for our CycleGAN to effectively bridge the gap between these domains. However, in all other scenarios (complex-to-simpler or simpler-to-complex), the shift between the target domain and the augmented target domain is significantly reduced compared to the shift between the source and target domains, indicating that our CycleGAN successfully transforms the source domain data, thereby reducing the domain gap.

5.4.2. Data Augmentation for Domain Adaptation

In our experiments, we evaluate our approach by estimating RUL using DANN and CORAL with and without augmented data in multiple transfer scenarios that involve different operational conditions and fault types. Both RMSE and score of all our experiment can be found in Table 3 and Table 4 respectively.

For scenario with lower domain shift (FD001/FD003 to FD002/FD004), the performance of the augmented configurations closely matches that of the non-augmented ones, with no significant improvement in RMSE or score (e.g., similar RMSE before and after augmentation on FD001 $\rightarrow$ FD002 for both DANN and CORAL).

Source $\rightarrow$ Target	$W(\mathcal{S}, \mathcal{T})$	$W(\mathcal{T}, \mathcal{T}_{aug})$
FD001 $\rightarrow$ FD002	0.35	$\mathbf{0.20}$
FD001 $\rightarrow$ FD003	$\mathbf{0.04}$	0.09
FD001 $\rightarrow$ FD004	0.36	$\mathbf{0.15}$
FD002 $\rightarrow$ FD001	0.35	$\mathbf{0.23}$
FD002 $\rightarrow$ FD003	0.33	$\mathbf{0.24}$
FD002 $\rightarrow$ FD004	$\mathbf{0.01}$	0.11
FD003 $\rightarrow$ FD001	$\mathbf{0.04}$	0.08
FD003 $\rightarrow$ FD002	0.33	$\mathbf{0.14}$
FD003 $\rightarrow$ FD004	0.33	$\mathbf{0.19}$
FD004 $\rightarrow$ FD001	0.36	$\mathbf{0.16}$
FD004 $\rightarrow$ FD002	$\mathbf{0.01}$	0.11
FD004 $\rightarrow$ FD003	0.33	$\mathbf{0.10}$

Table 2. Domain shift evaluation between source and target domain as well as between target and augmented target. Lower distance, in bold, correspond to lower domain shift

In contrast, for simpler transfers (FD002/FD004 to FD001/FD003), augmentation leads to substantial improvements, particularly with DANN. For instance, in the FD002 $\rightarrow$ FD001 scenario, the RMSE improves from 33.0 to 27.1, and the score from $5,706$ to $1,877$.

When the domain shift involves similar operational conditions (FD001/FD003 and FD002/FD004), augmentation also proves highly effective. Similar to the simpler transfer of operational conditions, a reduction in fault type complexity from more complex to simpler domains leads to significant improvements. For example, in the FD004 $\rightarrow$ FD002 transfer with DANN, augmentation reduces the RMSE from 40.0 to 23.7 and the score from $18,885$ to $2,102$. Similarly, in the FD003 $\rightarrow$ FD001 transfer with DANN, the RMSE improves from 29.9 to 23.7 and the score from $8,660$ to $1,729$.

Overall, our augmentation technique yields superior results with the DANN configuration compared to the CORAL configuration. This disparity in performance might be attributed to DANN's use of adversarial training to learn domain-invariant features, enhancing its capability to handle complex, non-linear domain shifts. The augmented data produced by CycleGAN enriches the target domain, offering DANN a wider range of examples to align the source and target distributions more effectively. Conversely, CORAL focuses solely on matching second-order statistics (covariances), which may not capture the intricate relationships between the domains, thus limiting its potential to use the augmented data effectively.

6. CONCLUSION

In this paper, we proposed a CycleGAN-based data augmentation technique to enhance the performance of Remaining Useful Life (RUL) prediction under unsupervised domain

Source → Target	Source only	DANN	DANN w/ Aug	CORAL	CORAL w/ Aug
FD001 → FD002	70.4 (±4.2)	**55.6** (±1.2)	56.0 (±2.3)	64.0 (±4.8)	**64.0** (±1.9)
FD001 → FD003	50.2 (±2.7)	39.8 (±1.5)	**37.5** (±1.3)	41.2 (±1.7)	**39.2** (±1.0)
FD001 → FD004	69.2 (±8.4)	**46.6** (±2.9)	49.2 (±1.7)	**55.0** (±6.0)	55.5 (±5.3)
FD002 → FD001	45.2 (±8.8)	33.0 (±5.8)	**27.1** (±3.7)	**41.1** (±3.3)	42.5 (±1.9)
FD002 → FD003	142.2 (±12.3)	44.5 (±4.0)	**42.8** (±2.7)	42.3 (±8.2)	**40.3** (±5.0)
FD002 → FD004	47.4 (±3.9)	43.6 (±1.3)	**42.2** (±0.5)	55.3 (±6.5)	**55.2** (±3.0)
FD003 → FD001	43.6 (±5.2)	29.2 (±2.9)	**23.7** (±1.9)	**46.9** (±0.8)	44.4 (±1.6)
FD003 → FD002	69.6 (±3.9)	**57.1** (±0.6)	57.8 (±1.4)	57.0 (±3.3)	**55.7** (±4.1)
FD003 → FD004	70.5 (±5.1)	46.5 (±0.6)	**46.2** (±0.4)	57.5 (±1.3)	**55.4** (±1.8)
FD004 → FD001	191.3 (±36.4)	41.7 (±13.7)	**41.1** (±7.1)	70.6 (±6.7)	**68.7** (±5.4)
FD004 → FD002	36.4 (±0.4)	40.0 (±6.9)	**23.7** (±0.9)	46.9 (±3.1)	**46.3** (±2.9)
FD004 → FD003	147.8 (±17.0)	40.6 (±2.8)	**38.8** (±4.4)	48.5 (±1.8)	**46.7** (±1.9)

Table 3. RMSE comparison of RUL prediction models applied on the target domain for five approaches: Source Only, DANN, DANN with CycleGAN-based data augmentation (DANN w/ Aug), CORAL and CORAL with CycleGAN-based data augmentation (CORAL w/ Aug). The results demonstrate that CycleGAN-based augmentation achieves help Adaptation models to reach best performance in many scenarios. Best scenario between with and without augmentation for both DANN and CORAL are in bold.

Source → Target	Source only	DANN	DANN w/ Aug	CORAL	CORAL w/ Aug
FD001 → FD002	$> 10^6$ ($\pm > 10^6$)	$\mathbf{104,814}$ ($\pm 14,981$)	$176,110$ ($\pm 20,858$)	$\mathbf{158,285}$ ($\mathbf{\pm 25,847}$)	$178,359$ ($\pm 23,847$)
FD001 → FD003	$159,850$ ($\pm 39,744$)	$21,053$ ($\pm 5,275$)	$\mathbf{12,529}$ ($\pm 3,897$)	$24,277$ ($\pm 1,523$)	$\mathbf{23,918}$ ($\mathbf{\pm 1,694}$)
FD001 → FD004	$> 10^6$ ($\pm > 10^6$)	$\mathbf{30,165}$ ($\pm 20,921$)	$43,845$ ($\pm 14,875$)	$\mathbf{114,442}$ ($\mathbf{\pm 24,379}$)	$116,206$ ($\pm 25,837$)
FD002 → FD001	$> 10^6$ ($\pm > 10^6$)	$5,706$ ($\pm 2,755$)	$\mathbf{1,877}$ (± 588)	$\mathbf{15,689}$ ($\mathbf{\pm 2,844}$)	$16,510$ ($\pm 3,674$)
FD002 → FD003	$> 10^6$ ($\pm > 10^6$)	$\mathbf{44,270}$ ($\pm 20,245$)	$55,842$ ($\pm 39,348$)	$39,444$ ($\pm 4,118$)	$\mathbf{34,644}$ ($\mathbf{\pm 4,544}$)
FD002 → FD004	$> 10^6$ ($\pm > 10^6$)	$78,277$ ($\pm 33,020$)	$\mathbf{42,283}$ ($\pm 14,761$)	$302,624$ ($\pm 33,198$)	$\mathbf{293,892}$ ($\mathbf{\pm 89,302}$)
FD003 → FD001	$15,555$ ($\pm 11,790$)	$8,660$ ($\pm 3,586$)	$\mathbf{1,729}$ (± 701)	$\mathbf{83,567}$ ($\mathbf{\pm 14,017}$)	$85,266$ ($\pm 11,271$)
FD003 → FD002	$> 10^6$ ($\pm > 10^6$)	$\mathbf{90,110}$ ($\pm 15,470$)	$105,995$ ($\pm 38,768$)	$108,813$ ($\pm 16,341$)	$\mathbf{75,287}$ ($\mathbf{\pm 12,100}$)
FD003 → FD004	$> 10^6$ ($\pm > 10^6$)	$60,062$ ($\pm 10,915$)	$\mathbf{43,845}$ ($\pm 14,875$)	$138,395$ ($\pm 19,403$)	$\mathbf{55,724}$ ($\mathbf{\pm 21,311}$)
FD004 → FD001	$> 10^6$ ($\pm > 10^6$)	$26,908$ ($\pm 24,168$)	$\mathbf{24,081}$ ($\pm 20,875$)	$207,659$ ($\pm 56,933$)	$\mathbf{112,753}$ ($\mathbf{\pm 9,227}$)
FD004 → FD002	$3,174$ (± 125)	$18,885$ ($\pm 8,182$)	$\mathbf{2,102}$ (± 360)	$107,919$ ($\pm 9,481$)	$\mathbf{94,439}$ ($\mathbf{\pm 8,119}$)
FD004 → FD003	$> 10^6$ ($\pm > 10^6$)	$21,337$ ($\pm 7,408$)	$\mathbf{18,024}$ ($\pm 5,459$)	$135,745$ ($\pm 18,369$)	$\mathbf{86,446}$ ($\mathbf{\pm 7,172}$)

Table 4. Score comparison of RUL prediction models applied on the target domain for five approaches: Source Only, DANN, DANN with CycleGAN-based data augmentation (DANN w/ Aug), CORAL and CORAL with CycleGAN-based data augmentation (CORAL w/ Aug). The results demonstrate that CycleGAN-based augmentation achieves help Adaptation models to reach best performance in many scenarios. Best scenario between with and without augmentation for both DANN and CORAL are in bold.

adaptation. Predictive maintenance is critical for improving operational efficiency and reducing costs, yet models often struggle with domain shifts due to variations in operational conditions and data scarcity in the target domain. To address this, we employed a Wasserstein CycleGAN with Gradient Penalty (W-CycleGAN-GP) to generate realistic augmented data in the target domain from source domain data. This augmented data bridges the domain gap, allowing Unsupervised Domain Adaptation (UDA) models (DANN and CORAL) to align domains more effectively.

We validated our approach using the C-MAPSS dataset. The experimental results demonstrated that our proposed method bridge domain gap and significantly improves the accuracy of the RUL prediction in most scenarios. UDA models augmented with CycleGAN-generated data achieved lower Root Mean Square Error (RMSE) and a better Score compared to both the source-only model and the standard non-augmented model. The proposed method proved to be effective in handling complex domain shifts, especially in transfers from more complex to simpler domains, where it significantly improved both RMSE and score(e.g., in the FD004 → FD002 transfer with DANN, RMSE was reduced from 40.0 to 23.7 and the score from 18,885 to 2,102). This underscores the value of diverse and realistic augmented data in enhancing the generalization capabilities of UDA models. Future work will focus on further optimizing the CycleGAN and DANN architectures and exploring their applicability to other datasets from in various industries.

ACKNOWLEDGEMENT

This research was funded in whole by the Luxembourg National Research Fund (FNR), grant reference UPTIME4.0 and project reference 16756339. For the purpose of open access, the author has applied a Creative Commons Attribution 4.0 International (CC BY 4.0) license to any Author Accepted Manuscript version arising from this submission.

REFERENCES

Arjovsky, M., & Bottou, L. (2017). Towards Principled Methods for Training Generative Adversarial Networks. , 1–17. Retrieved from http://arxiv.org/abs/1701.04862

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever, I., & Abbeel, P. (2016). Infogan: Interpretable representation learning by information maximizing generative adversarial nets. *Advances in neural information processing systems*, *29*.

da Costa, P. R. d. O., Akçay, A., Zhang, Y., & Kaymak, U. (2020). Remaining useful lifetime prediction via deep domain adaptation. *Reliability Engineering & System Safety*, *195*, 106682.

Donahue, C., McAuley, J., & Puckette, M. (2018). Adversarial audio synthesis. *arXiv preprint arXiv:1802.04208*.

Engel, J., Agrawal, K. K., Chen, S., Gulrajani, I., Donahue, C., & Roberts, A. (2019). Gansynth: Adversarial neural audio synthesis. *arXiv preprint arXiv:1902.08710*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., … Lempitsky, V. (2016). Domain-adversarial training of neural networks. *Journal of machine learning research*, *17*(59), 1–35.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., … Bengio, Y. (2020, oct). Generative adversarial networks. *Commun. ACM*, *63*(11), 139–144. Retrieved from https://doi.org/10.1145/3422622 doi: 10.1145/3422622

Gulrajani, I., Ahmed, F., Arjovsky, M., Dumoulin, V., & Courville, A. (2017). Improved training of wasserstein gans. , 5769–5779.

Hsu, W.-N., Zhang, Y., & Glass, J. (2017). Unsupervised domain adaptation for robust speech recognition via variational autoencoder-based data augmentation. , 16–23.

Hu, T., Guo, Y., Gu, L., Zhou, Y., Zhang, Z., & Zhou, Z. (2022). Remaining useful life estimation of bearings under different working conditions via wasserstein distance-based weighted domain adaptation. *Reliability Engineering & System Safety*, *224*, 108526.

Iacono, P., & Khan, N. (2023). Structure preserving cyclegan for unsupervised medical image domain adaptation. *arXiv preprint arXiv:2304.09164*.

Karras, T., Laine, S., & Aila, T. (2019). A style-based generator architecture for generative adversarial networks. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition* (pp. 4401–4410).

Li, X., Li, J., Zuo, L., Zhu, L., & Shen, H. T. (2022). Domain adaptive remaining useful life prediction with transformer. *IEEE Transactions on Instrumentation and Measurement*, *71*, 1-13. doi: 10.1109/TIM.2022.3200667

Long, M., Cao, Z., Wang, J., & Jordan, M. I. (2018). Conditional adversarial domain adaptation. *Advances in neural information processing systems*, *31*.

Long, M., Wang, J., Ding, G., Sun, J., & Yu, P. S. (2013). Transfer feature learning with joint distribution adaptation. , 2200–2207.

Long, M., Wang, J., Ding, G., Sun, J., & Yu, P. S. (2014). Transfer joint matching for unsupervised domain adaptation. , 1410–1417.

Luleci, F., Catbas, F. N., & Avci, O. (2023). Cyclegan for undamaged-to-damaged domain translation for structural health monitoring and damage detection. *Mechanical Systems and Signal Processing*, *197*, 110370.

Mirza, M., & Osindero, S. (2014). Conditional generative adversarial nets. *arXiv preprint arXiv:1411.1784*.

Nejjar, I., Geissmann, F., Zhao, M., Taal, C., & Fink, O. (2024). Domain adaptation via alignment of operation profile for remaining useful lifetime prediction. *Reliability Engineering & System Safety*, *242*, 109718.

Ozdagli, A., & Koutsoukos, X. (2020). Domain adaptation for structural health monitoring. , *12*(1), 9–9.

Palladino, J. A., Slezak, D. F., & Ferrante, E. (2020). Unsupervised domain adaptation via cyclegan for white matter hyperintensity segmentation in multicenter mr images. , *11583*, 1158302.

Pu, Z., Cabrera, D., Li, C., & de Oliveira, J. V. (2023). Sliced wasserstein cycle consistency generative adversarial networks for fault data augmentation of an industrial robot. *Expert Systems with Applications*, *222*, 119754.

Radford, A., Metz, L., & Chintala, S. (2015). Unsupervised representation learning with deep convolutional generative adversarial networks. *CoRR*, *abs/1511.06434*.

Saravanan, S. S., Luo, T., & Van Ngo, M. (2023). Tsi-gan: Unsupervised time series anomaly detection using convolutional cycle-consistent generative adversarial networks. , 39–54.

Saxena, A., & Goebel, K. (2008). Turbofan engine degradation simulation data set. *NASA ames prognostics data repository*, *18*, 878–887.

Saxena, A., Goebel, K., Simon, D., & Eklund, N. (2008). Damage propagation modeling for aircraft engine run-to-failure simulation. In *2008 international conference on prognostics and health management* (pp. 1–9).

Schockaert, C., & Hoyez, H. (2020). Mts-cyclegan: An adversarial-based deep mapping learning network for multivariate time series domain adaptation applied to the ironmaking industry. *arXiv preprint arXiv:2007.07518*.

Sun, B., Feng, J., & Saenko, K. (2017). Correlation alignment for unsupervised domain adaptation. *Domain adaptation in computer vision applications*, 153–171.

Sun, B., & Saenko, K. (2016). Deep coral: Correlation alignment for deep domain adaptation. , 443–450.

Tzeng, E., Hoffman, J., Saenko, K., & Darrell, T. (2017).

Adversarial discriminative domain adaptation. , 7167–7176.

Volpi, R., Morerio, P., Savarese, S., & Murino, V. (2018). Adversarial feature augmentation for unsupervised domain adaptation. In *Proceedings of the ieee conference on computer vision and pattern recognition* (pp. 5495–5504).

Wang, Q., Meng, F., & Breckon, T. P. (2023). Data augmentation with norm-ae and selective pseudo-labelling for unsupervised domain adaptation. *Neural Networks*, *161*, 614–625.

Zotov, E., & Kadirkamanathan, V. (2021). Cyclestylegan-based knowledge transfer for a machining digital twin. *Frontiers in Artificial Intelligence*, *4*. doi: 10.3389/frai.2021.767451

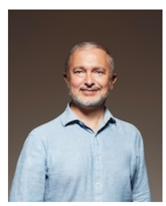
BIOGRAPHIES

Dorian Joubaud is is a doctoral researcher and data scientist engineer at the Interdisciplinary Centre for Security, Reliability and Trust (SnT) at the University of Luxembourg. His research focuses on trustworthy AI, data augmentation and predictive maintenance in Industry 4.0. He is involved in projects like Uptime4.0, which aims to optimize maintenance schedules and reduce waste in manufacturing using AI techniques.

Evgeny Zotov is a Research Associate at the Interdisciplinary Centre for Security, Reliability, and Trust (SnT) at the University of Luxembourg. His research is centered on the application of Artificial Intelligence (AI) within the manufacturing industry, with a focus on leveraging advanced statistical modelling and Deep Learning for Digital Twin technologies.

Oguz BEKTAS is an academic researcher who has graduated from the University of Warwick with a PhD. He is currently affiliated with the University of Luxembourg (Interdisciplinary Centre for Security, Reliability, and Trust (SnT)) and has made contributions to research in areas such as prognostics, condition monitoring, and artificial intelligence. His work encompasses the development of neural network filtering approaches for remaining useful life estimation, computational prognostic methods for complex systems, and dimensionality reduction for failure prognostics. Additionally, Oguz has been involved in the development of models for gas turbine engine prognostics, automated emergency landing systems for drones, and anomaly detection in unmanned aircraft. Furthermore, his research includes the analysis of visual imagery for emergency drone landing, data-driven predictive maintenance models, and the definition of degradation curves for rotary mechanisms.

Sylvain KUBLER is a Research Scientist at the Interdisciplinary Centre for Security, Reliability, and Trust (SnT) at the University of Luxembourg. Previously, he was an Associate Professor at the Research Center for Automatic Control of Nancy (UMR 7039) at Université de Lorraine (2017-2022), a Research Associate at SnT (2015-2017), and a Postdoctoral Researcher at Aalto University (2013-2015). He received the Best Thesis Award in Automatic Control from the IFAC French Workgroup GdR MACS/Club EEA. Sylvain has played a leading role in multiple projects funded by the European Commission (e.g., bIoTope H2020, ASCEND Horizon Europe, AVATAR Horizon Europe, RED WoLF Interreg NWE) and national research programs (ANR, FNR). His expertise lies in decision support systems using AI/ML, with a focus on ensuring their trustworthiness, following the developments of the AI Act and other relevant regulations.



Yves LE TRAON is full professor of Computer Science at University of Luxembourg, in the domain of software engineering, with a focus on software testing, software security, and data-intensive systems. He is the Director of the Interdisciplinary Centre for Security, Reliability and Trust. His research interests include (1) innovative testing and debugging techniques, (2) mobile Android security using static code analysis, machine learning techniques and, (3) design of machine-learning based systems in the domains of IoT, smart grid, Fintech, Industry 4.0, and data-intensive systems in general. His research is inspired from and applies to several industry partners (BGL BNP-Paribas, CREOS, Paypal, Cebi. . .). His achievements and visibility have been recently acknowledged by a JSS article, ranking him among the top-15 most impactful experienced SE researchers in the past decade, worldwide. Yves Le Traon is one of the recipients of the Facebook Testing and Verification 2019 - Research Award. He has been elevated IEEE Fellow in 2021.