IEEE Transactions on Intelligent Vehicles, Volume 9, Issue 2, February 2024, pp. 3205-3219
DOI:10.1109/TIV.2023.3342648

Learning from the Dark Side: A Parallel Time Series Modelling Framework for Forecasting and Fault Detection on Intelligent Vehicles

Yang Xing, *Member, IEEE*, Zhongxu Hu, *Member, IEEE*, Peng Hang, *Member, IEEE*, Chen Lv, *Senior Member, IEEE*

***Abstract*—Time series vehicle state modelling is crucial in various real-world applications, such as fault detection, fault tolerance, optimization, and cyber security for intelligent vehicles (IVs). In this study, we propose a novel parallel time series modeling framework (PTSM) to forecast and detect vehicle braking cylinder pressure states, thereby enhancing the safety of the braking system. Specifically, the PTSM consists of two branches: LightNet and DarkNet. The LightNet learns time-series (TS) representations of real-world signals to forecasts and identifies vehicle states. On the other hand, the DarkNet employs a novel multi-task learning and dual Relativistic Generative Adversarial Network (dual-RaGAN) framework to reconstructs healthy sequential states, detects faults, and forecasts future vehicle states using synthesized faulty sequences. To develop the PTSM framework, we introduce a novel data processing and random fault synthesizing method. We evaluate the performance of the dual-RaGAN model using real-world data and compare it with non-adversarial approaches, demonstrating the efficiency of the multi-task generative sequential representation. Extensive experimental results show that by integrating knowledge from the dark side, real-world time-series modelling (TSM) for forecasting and fault detection can be significantly improved, with a 34.7% enhancement in forecasting and an 11% improvement in fault recognition. The results indicate that signal reconstruction leads to more accurate sequence forecasting and fault recognition in both the dark and light sides. This proposed study not only introduces a novel time-series modelling framework but also establishes a new approach for vehicle testing, fault detection, and cyber security research for intelligent vehicles. Data and Codes are available at: https://github.com/YXING-CC/Dark-Light.**

***Index Terms*—Time-series modelling, generative time-series model, relativistic GAN, deep learning, fault detection, intelligent vehicles**

I. Introduction

TIME series modelling (TSM) is a valuable tool for ensuring the safe operation of intelligent vehicles, particularly in safety-critical situations. For example, TSM plays a crucial role in monitoring and forecasting vehicle states, enabling vehicle fault detection, tolerance [1,2], energy consumption optimization [3,4], motion planning [5] and cyber security [6,7]. Detecting faults in signals is essential to promptly address potential risks to passengers, other road users, and infrastructure. However, anomaly data representing faults are often scarce in real-world scenarios, making it challenging to develop data intensive TSM approaches that accurately detect faults and support system recovery [8,9]. Furthermore, while previous studies have focused primarily on fault recognition, less attention has been given to forecasting and reasoning for new future health sequences. This aspect is vital for immediate fault tolerance and ensuring system stability maintenance [10].

Therefore, in this study, we aim to: 1) Address the data limitation and robustness challenges in developing an efficient Time Series Modeling (TSM) approach for intelligent vehicles. 2) Test and quantitatively evaluate how knowledge from the virtual world can be integrated into the real-world system without change of the architecture of the real-world model. We propose a Parallel Time Series Modeling (PTSM) framework that incorporates a hierarchical multi-task learning approach for joint vehicle states prediction and health monitoring. The PTSM framework aims to consistently predict the real vehicle states within a specified prediction horizon while identifying different types of faulty signals from the sequential input data. To enhance the model's robustness and generalization ability, we employ a multi-task generative learning framework.

Additionally, we adopt the parallel concept [11] to design the two-branch PTSM model, leading to the development of two distinct "worlds" within the framework. The first world is referred to as the "light world" (real world), where only limited faulty signals exist. The second world is the "dark world" (virtual world), representing an extremely critical operation environment for intelligent vehicles, and the sequential inputs in this world are abundant with faulty signals. Meanwhile, the model architecture on the dark side is much more complex than that on the light side in case to learn comprehensive representation for both historical and future sequences from a large amount of synthesized faulty sequences. To validate our PTSM framework, we collected real-world time-series cylinder pressure signals from an electric vehicle testbed. Further, we introduced eight common artificial faults [12] by injecting them into the original data to construct the virtual world. This approach allows us to effectively simulate and analyze fault

This work was supported in part by the Agency for Science, Technology and Research (A*STAR), Singapore, under Advanced Manufacturing and Engineering (AME) Young Individual Research under Grant A2084c0156, the MTC Individual Research Grant (M22K2c0079), the ANR-NRF Joint Grant (No.NRF2021-NRF-ANR003 HM Science), and the Ministry of Education (MOE), Singapore, under the Tier 2 Grant (MOE-T2EP50222-0002). (Corresponding authors are Chen Lv and Yang Xing).

Yang Xing is with the School of Aerospace, Transport, and Manufacturing, Cranfield University, MK43 0AL, UK. (e-mail: yang.x@cranfield.ac.uk).

Zhongxu.Hu and Chen Lv are with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, 639798, Singapore. (e-mail: zhongxu.hu@ntu.edu.sg, lyuchen@ntu.edu.sg).

Peng Hang is with the Department of Traffic Engineering and Key Laboratory of Road and Traffic Engineering, Ministry of Education, Tongji University, Shanghai 201804, China. (e-mail: hangpeng@tongji.edu.cn).

scenarios in a controlled environment. Through this research, we aim to overcome the data limitation challenge in TSM for intelligent vehicles and provide a robust framework that can predict vehicle states accurately, monitor health conditions, and detect faults effectively.

In summary, in this study, we proposed a parallel time-series modelling framework for state forecasting and fault detection of intelligent vehicles. The main contribution can be summarized as follows.

- First, a novel parallel two-branch framework is developed to enable parallel learning and knowledge transfer. The proposed methods provide an efficient development, testing, and validation framework for intelligent vehicles.
- Second, a generative representation approach for multivariate sequence reconstruction in designed and the impacts of sequence reconstruction to the forecasting in both real and virtual world are quantitively studied.
- Third, a multi-task time-series modelling framework is developed based on a dual-RGAN protocol. The adversarial multi-task learning method shows significant improvement on multiple tasks compared to the conventional frameworks.
- Fourth, a synthesized data generation and preparation method is developed to inject faults into real-world data and support the training of the PTSM framework.
- Last, extensive experiments are designed to comprehensively evaluate the performance of different networks and the whole PTSM model.

The remainder of this paper is as follows. In Section II, a comprehensive literature review is proposed for state-of-the-art time-series modelling, state estimation, and parallel approaches. In Section III, the methodologies for the PTSM and dual-RGAN framework will be introduced. The extensive experiment evaluation and comparison results will be discussed in Section IV, and finally, the study is concluded in Section V.

II. RELATED WORKS

A. Time-Series Modelling

TSM has been a subject of extensive research since the 1980s. Various statistical models, such as Auto Regressive Integrated Moving Average (ARIMA) [13], Nonlinear autoregressive exogenous (NARX) [14], and exponential smoothing [15], have demonstrated their effectiveness in early univariate sequence modelling and forecasting. However, these methods have significant limitations when it comes to long-term sequence modelling and forecasting tasks [16]. Leveraging the fast development of neural networks and deep learning, recently data-driven-based machine learning models have become the mainstream for TSM. Two notable examples are Temporal Convolutional Networks (TCN) and WaveNet, which relies on 1D convolutional filters and dilated connections to enhance the receptive field of sequential data [17, 18]. Another crucial branch in this area is recurrent neural networks (RNN), which include models like Long-Short Term Memory (LSTM) and Gated Recurrent Unit (GRU) [19, 20]. These RNN-based models have proven to be highly effective in capturing long-term dependencies in time series data and have significantly improved forecasting accuracy compared to traditional statistical methods.

Since the publication of Self-attention and Transformer in 2017 [21], there have been significant advancements in Time Series Modeling (TSM) using Transformer networks. Transformer-based models have shown superior performance compared to conventional Recurrent Neural Networks (RNNs), especially in large language models like Generative Pre-trained Transformers (GPTs) [22]. In the realm of multivariate sequence modelling, numerous powerful models based on the Transformer architecture have been developed. For example, in [23], a Temporal Fusion Transformer (TFT) model was developed for multi-horizon time series forecasting with interpretable insights into the temporal dynamics. In [24], an efficient and accurate anomalous observations model based on Transformer (TransAD) model was developed for anomaly detection. A score-based self-conditioning was introduced for robust multi-modal feature extraction. In [25], a LogSparse Transformer (Logformer) network was proposed by introducing a causal convolution-based self-attention mechanism to better represent the local context. The current Informer network adopts a ProbSparse self-attention mechanism with self-attention distilling and generative decoder [26]. Then, the Autoformer further improve the long-term forecasting problem with a novel decomposition architecture and Auto-Correlation mechanism [27]. The exploration of decomposition approaches led to the development of FEDformer [28], a frequency domain TSM model for low-complexity long-term sequence forecasting. Among these advancements, TimesNet [29] currently achieves state-of-the-art results in five different TSM tasks, including long-term forecasting, short-term forecasting, imputation, anomaly detection, and classification. TimesNet relies on 2D intraperiod-interperiod variation modelling, showcasing the significant impact of Transformer-based architectures in the field of TSM.

B. State Estimation and Fault Detection of IVs

Vehicle states estimation, including state prediction [30], fault detection for CAN bus signals [31,32], motion and trajectory predictions [33], and engine, motor, and battery states (e.g., state-of-charge and state-of-health) [34,35], plays a crucial role in the operation of intelligent vehicles. Among these areas, fault and anomaly detection have been widely studied. The sensor data in connected vehicles are susceptible to vulnerabilities, and common anomalies such as faults, errors, and cyberattacks need to be identified. To address these challenges, a multi-stage attention mechanism combined with an LSTM-based Convolutional Neural Network (CNN) approach was developed for detecting four common anomalies [36]. Similarly, a combined CNN and Kalman Filter model was developed to detect the instant, constant, gradual shift, and bias anomalies [37]. In [38], an event-based one-class anomaly detection model based on a support vector machine (SVM) was developed for the hybrid control unit of the hybrid electric vehicle. In [39], an unsupervised interpretable

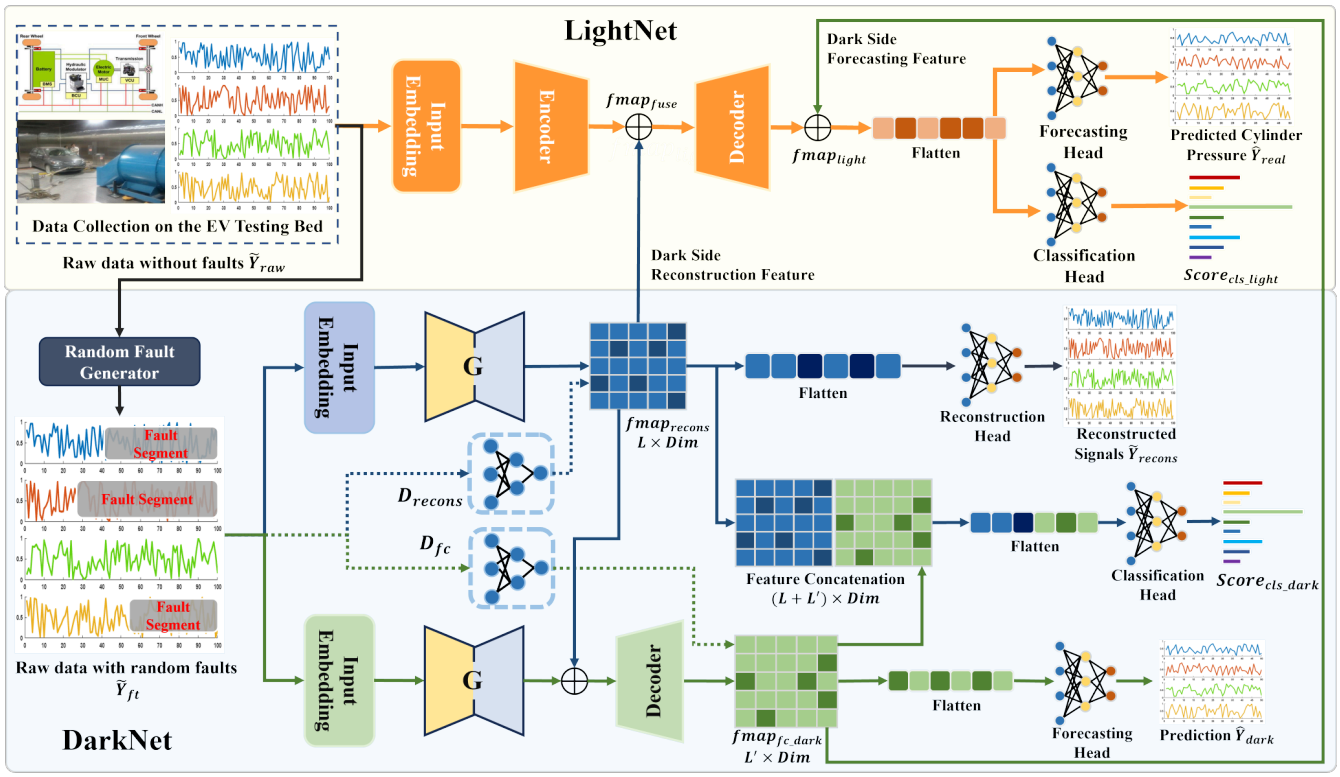


Fig. 1. Overall architecture of the proposed Parallel Time-Series Modelling (PTSM) Framework. The PTSM framework contains two parts, namely, LightNet and DarkNet. The LightNet learns to recognize the type of fault and forecast the future states based on the real-world measurement from the cylinder sensors. The DarkNet learns fault recognition, forecasting, and an extra health signal reconstruction based on the synthesised faulty signals. The learned knowledge for historical signals reconstruction and future signal forecasting from the dark side will be fed into the LightNet for real-world TSM augmentation. The four-dimensional braking cylinder pressures are used to generate the inputs and targets for both LightNet and DarkNet.

autoencoder model was used to detect the anomalies of electric vehicles while charging. In [40], a hybrid neural network that combines the 1-D convolutional neural network and active-state-tracking LSTM models are developed to capture the hierarchical features between the variables that can affect battery degradation. It was shown that the longitudinal and lateral state estimation of the proceeding vehicles can help reduce the communication burden for connected automated vehicles and an event-trigger-based cubature Kalman filter can achieve efficient and accurate estimation results. These recent studies all show the significance and necessity of sensor health status monitoring for intelligent vehicles.

Though a series of studies have been developed for vehicle state estimation, the combination of accurate sequence forecasting and fault detection for vehicle state modelling remains an open question. Meanwhile, it is unclear how the hybrid multi-task learning model can capture the complex temporal dynamics of the vehicle state based on real-world scarce data. Therefore, in this study, we will focus on building a novel parallel learning framework for TSM in intelligent vehicles. By addressing the integration of forecasting and fault detection and adapting to the scarcity of data, this study strives to contribute valuable insights into the development of robust sensor health state monitoring.

C. Virtual Testing and Parallel Learning

The collection of faulty signals in the real world can be challenging, particularly when trying to ensure diversity in faulty scenarios to support a robust data-driven model. Consequently, to develop an efficient data-driven approach for fault detection and address the lack of real-world faulty signals, virtual simulation and testing methods are often employed to synthesize artificial faults to the real-world health signals. For instance, in [41], a multi-sensor fault detection and isolation system is based on the 1D-CNN time-series model. Virtual faults including drift, hard-over, erratic, and spike faults were generated based on the Audi A2D2 datasets [42]. In [43], a fault injection module was developed for a hardware-in-the-loop testing system of automotive software systems. Eight faults (spike, offset, noise, hard-over, delay, gain, stuck, and drift) were generated and injected into the accelerator pedal and engine speed sensors in the HIL testing system. The combined 1D-CNN and LSTM approach shows a 98.88% F1 score of the fault classification. In [12], a digital twin-inspired Wasserstein GAN-based binary classifier was developed for early fault detection in wireless sensor networks. The WGAN tasks the Gramian angular filed (GAF) encoding of the time series as input and output the binary faulty/healthy condition.

In addition to direct fault classification, learning the original representation of temporal dynamics in time series and enabling reconstruction for extensive testing and simulation is also critical and beneficial. Signal reconstruction has emerged

as a key topic in recent digital twin and parallel learning theories [44-46]. For instance, parallel driving is an advanced framework that facilitates large-scale computational experiments and parallel execution for intelligent vehicles. This framework necessitates the development of a virtual environment representing the holistic ecosystem of intelligent vehicles. The study conducted in [47] introduces three key concepts: artificial systems, computational experiments, and parallel execution, to support the development of parallel intelligent systems. In [48], a cyber-physical-social systems (CPSS) framework beyond the conventional digital twin and cyber-physical- systems (CPS) were studied for parallel driving systems and the software-defined intelligent vehicles in the metaverses were clarified. Therefore, in this study, we will investigate the signal reconstruction performance in the virtual world considering the adversarial situations where synthesized faults are injected. As discussed in Section IV, the joint learning of signal reconstruction along with the forecasting and fault detection tasks in the virtual world could significantly benefit the tasks in the real world.

III. METHODOLOGY

A. Overall Architecture

The overall architecture of the proposed PTSM framework is shown in Fig. 1. Specifically, we split the framework into two branches, namely the light side, and the dark side. On the light side, we define the model as LightNet which learns to monitor the real-time faults and predict the future cylinder pressure for braking behaviour estimation. The forecasting of the continuous braking pressure can not only help tolerate the signal faults but also benefit energy management. The input to the LightNet is real-world time-series data from cylinder signals where only very limited faulty signals exist. On the dark side, however, we use much more faulty sequences to train the DarkNet to learn the same forecasting and classification tasks as the LightNet. Also, we introduce an extra reconstruction task for the faulty signals to further learn the temporal representation of the cylinder pressure signals. As the cylinder pressure time series is causal and depend on the driver's braking actions that respect to the highly dynamic and stochastic traffic context, a long-term forecasting is unreasonable. Thus, in this study, we only focus on short-term forecasting (500ms). Formally, we can define the observed variable sequence as $Y = \{Y_{t_{0-\tau}}, \ldots, Y_{t_0} | Y_i \in \mathbb{R}^d\}$ and the future target sequence as $Y_T = \{Y_{t_0+1}, \ldots, Y_{t_0+h} | Y_i \in \mathbb{R}^d\}$, where Y and Y_i is multivariate sequences with d dimensions ($d = 4$ in this study), $\tau \in \mathbb{N}$ is the historical observation horizon for the input sequence and $h \in \mathbb{N}$is the prediction horizon of the model. The target for the LightNet is to predict the future sequence $\hat{Y}_T$ and sensor states $\hat{S}$. The faulty signals to the DarkNet can be represented as $\tilde{Y}_{ft} = \{Y_{ft_{t0-\tau}}, \ldots, Y_{ft_{t0}} | Y_{ft_{ti}} \in \mathbb{R}^d\}$. The reconstruction task seeks to estimate $\tilde{Y}_{recons}$, which approximates the true observed sequence Y. In this study, the length of the input sequence τ is set to 100, and the forecasting horizon h is set to 50. These selections are made to ensure effective modelling and forecasting of the vehicle's braking behaviour.

The overall PTSM algorithm is described in Algorithm 1 below. There are four main stages for the proposed PTSM framework, which are generate and inject synthetic fault signals, train DarkNet with dual-RaGAN, train the LightNet by fusing features from DarkNet, and real-time inference. Detailed implementation for each stage will be discussion in the following sessions.

Algorithm 1 Algorithm for PTSM

INPUT: raw sequence data pair $\mathbf{X} \in \mathbb{R}^{100\times4}$, $\mathbf{Y} \in \mathbb{R}^{50\times4}$
OUTPUT: Forcasting $\hat{Y}$ and vehicle states $\tilde{S}$
1. GENERATA AND INJECT SYNTHETIC FAULT SIGNALS
  Split $(\mathbf{X}, \mathbf{Y})$ into $(\mathbf{X_{train}}, \mathbf{Y_{train}})$ and $(\mathbf{X_{test}}, \mathbf{Y_{test}})$
  While $J < max\_duplicate(\theta_d)$ **do**
    While $I < max\_samples$ in $\mathbf{X_{train}}$ **do**
      Randomly generate θ_n, θ_c
      Inject fault segments to $\mathbf{X_{train_i}}$
  Repeat fault generation for $\mathbf{X_{test}}$
  Update $\mathbf{X_{train_{dark}}}, \mathbf{X_{train_{light}}}, \mathbf{X_{test_{dark}}}, \mathbf{X_{test_{light}}}$
2. TRAIN DARKNET WITH DUAL-RAGAN
  Initialize $G_{fc}, G_{recons}, D_{fc}, and D_{recons}$
  While $K < max\_epoch$ **do**
    for each mini–batch $b \in \mathbf{X_{train_{dark}}}$ **do**
      Train the dual generators G_{fc} and G_{recons} with learning rate $\lambda_{G\_fc}, \lambda_{G\_recons}$, and loss $\mathcal{L}_{G\_FC'}, \mathcal{L}_{G\_RECONS'}$
      Train the dual discriminator D_{fc} and D_{recons} with learning rate $\lambda_{D\_fc}, \lambda_{D\_recons}$, and loss $\mathcal{L}_{D\_FC'}, \mathcal{L}_{D\_RECONS'}$
    Anneal $\lambda_{G\_fc}, \lambda_{G\_recons}, \lambda_{D\_fc}, \lambda_{D\_recons}$ with a decay factor of ψ every ξ epoch
  Update $G_{fc}, G_{recons}, D_{fc}$, and D_{recons}
3. TRAIN LIGHTNET WITH STANDARD BACKPROPAGATION
  Initialize $LightNetwithtransformerbackbone$
  While $K < max\_epoch$ **do**
    each mini–batch $b \in \mathbf{X_{train_{light}}}$ **do**
      Freeze G_{fc} and G_{recons}, extract $fmap_{recons}$, and $fmap_{fc\_dark}$
      Train the LightNet with learning rate λ_L,and loss $\mathcal{L}_{fc\_mse}, \mathcal{L}_{cls}$
  Update LightNet
4. REAL-TIME INFERENCE
  for single batch $b \in \mathbf{X_{test_{light}}}$ **do**
    query G_{fc}, G_{recons} with b for $fmap_{recons}$ and $fmap_{fc\_dark}$
    Calculate $(\hat{Y}, \tilde{S})$ with:
      $LightNet(b, fmap_{recons}, fmap_{fc_{dark}})$
    Return $(\hat{Y}, \tilde{S})$

The time-series driving dataset was collected on a chassis dynamometer that was equipped with an electric vehicle operating under typical driving cycles. In this study, the standard New European Drive Cycle (NEDC), which combines the European Union Urban Driving Cycle (ECE) and the Extra Urban Driving Cycle (EUDC) is adopted to simulate real-world urban driving behaviours. The electric vehicle was a typical

electric passenger car which was equipped with a regenerative and hydraulic blended braking system. A permanent magnet synchronous motor was used to drive the front wheels of the vehicle and control the mode switch of the motor and generator. We use the real-world braking pressure of the cylinders from the four wheels for PTSM framework validation. The temporal dynamics of the brake pressures from the cylinders are more difficult to model due to the significant uncertainty and dynamic characteristics compared to the widely studied speed or SOC states. Vehicle data and powertrain states on the CAN bus are collected with a sampling frequency of 100 Hz. During testing, the heater and airconditioner are all switched off. The battery is fully charged g at 100% before the test. The driver is required to repeat NEDC driving cycles with a maximum deviation of 2 km/h in the speed profile. Finally, experimental data of 8122s containing eight NEDC driving cycles in total are recorded.

B. LightNet Development

The LightNet was built based on a vanilla Transformer Encoder with two output tasks, which are the prediction and fault classification heads. The input to the LightNet is the observed historical data $X \in \mathbb{R}^{l \times d}$, where l and d represent the length of the input signals (100) and the number of channels/dimensions (four-cylinder channels in this study). The encoder ($enc(\cdot)$) on the light side learns to forecast the future cylinder pressures as well as monitor and estimate any types of faults in the time series signal.

A fully-connected input embedding layer FC_{emb} is applied to the input X and the embedded input $X_{emb} = FC_{emb}(X)$, $X_{emb} \in \mathbb{R}^{ld^{'}}$ will be passed to the popular positional encoding layer, where the following $sine - cosine$ temporal pattern can be injected to X_{emb} [19]. The Transformer model can be viewed as a parameterized function f_θ that map input $X \in \mathbb{R}^{l \times d}$ to the output $Y \in \mathbb{R}^{l^{'} \times d^{'}}$, where the key component is the multi-head self-attention function which can be represented as:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V \tag{1}$$

where $Q(x_i) = W_q^T x_i$, $K(x_i) = W_k^T x_i$, $V(x_i) = W_v^T x_i$are the query, key, and value matrix.

Given the Transformer encoder $enc(\cdot)$, the learned temporal representation of the real-world cylinder pressure sequence on the light side is fused with the reconstruction feature map $fmap_{recons}$ from the dark side to strengthen the representation of the temporal dynamics for the observed sequential input. The fused feature map is denoted as $fmap_{fuse} \in \mathbb{R}^{100 \times 256}$. Then $fmap_{fuse}$ is flattened and passed to the decoder with a fully connected layer FC_{conv} and 1D ConvNet before it is fused with the $fmap_{fc_{dark}} \in \mathbb{R}^{50 \times 256}$ from the dark side for the final forecasting $\hat{Y}_{t+1:t+h} \in \mathbb{R}^{504}$ and fault style classification $\hat{S}_{light} \in \mathbb{R}^{9 \times 1}$.

$$fmap_{fuse_{temp}} = FC_{conv}(Conv1D(fmap_{fuse})) \tag{2}$$

$$fmap_{light} = fmap_{fuse_{temp}} + fmap_{fc_{dark}} \tag{3}$$

$$\hat{Y}_{t+1:t+\tau} = FC_{fc}(fmap_{light}) \tag{4}$$

$$\hat{S}_{light} = FC_{cls}(fmap_{light}) \tag{5}$$

where FC_{fc} and FC_{cls} represent the fully connected layers for signal forecasting and fault classification.

C. DarkNet Development

The DarkNet in this study is designed to learn the temporal representation of the time series cylinder pressure dynamic from fully synthesized faulty signals. The vehicle cylinder pressure prediction task is even more challenging because of the high uncertainty and dynamic nature of the cylinder sensors. The forecasting task on the dark side is also much more complex than that on the light side as it must be made based on faulty signals. Thus, we introduce an extra self-supervised data reconstruction task together with the forecasting and classification tasks to jointly learn the representation of both observed and predicted temporal dynamics. The multi-task learning framework can improve the robustness and generalization ability of the DarkNet and thus contribute to more accurate results on the light side (as analyzed later in Section part). Further, we designed a novel multi-task relativistic GAN framework to strengthen the forecasting, reconstruction, and classification performance for the DarkNet. As shown in [48], error accumulation in time series forecasting can hurt performance badly, and an adversarial training process can regularize the forecasting at the sequence level and contribute to the better representation of time series with more fidelity at the sequence level. In this study, we further investigate the effectiveness of the relativistic average GAN (RaGAN) approach as we believe for time series modelling, due to the commonly existing noises and uncertainties in the signals, it would be more effective to measure the relativistic nature of the real/fake data pairs $\tilde{x} = (x_r, x_f)$ and estimate the probability that the given real data is more realistic than fake data on average.

As shown in Fig.2, the DarkNet contains two parts. The RaGANFormer framework relies on a dual-RaGAN approach for multi-task learning. Separate generators and discriminators are designed for the forecasting and reconstruction tasks. In RaGAN, we can define the discriminator as $D(x)$ and generator as $G(x)$. Let Θ^G and Θ^D be the parameters for the generator G and discriminator D. Then, for the dual-RaGAN model, let the $D_{fc}(x)$ and $G_{fc}(x)$ be the discriminator and generator for the forecasting network, and $D_{fc}(x)$ and $G_{recons}(x)$ be the discriminator and generator for the reconstruction network. In this study, the $G_{fc}(x)$ and $G_{recons}(x)$ are developed based Transformer encoder, and $D_{fc}(x)$ and $D_{recons}(x)$ are built based on the WaveNet and dilated causal convolution frameworks [16]. The $G_{fc}(x)$ and $G_{recons}(x)$ share the same input $\tilde{Y}_{ft}(t-h:t)$, which is the 4-dimensional observed cylinder pressure data with random faults injected. t is the current time step, h is the historical horizon. Also, let the $FC_{fc}(x)$ and $FC_{recons}(x)$ be fully connected layers of the prediction head networks. Then, the forecasting and reconstruction networks have different tasks such as:

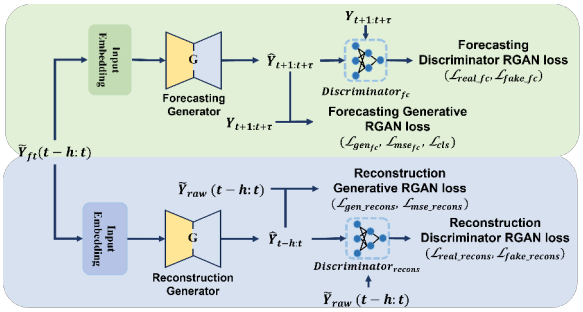


Fig. 2. Demonstration of the dual-RGAN time-series model and learning objectives on the dark side.

$$\hat{Y}_{t+1:t+\tau} = FC_{fc}(G_{fc}(\tilde{Y}_{ft}(t-h:t); \Theta^G_{fc})) \quad (6)$$

$$\hat{Y}_{t-h:t} = FC_{recons}(G_{recons}(\tilde{Y}_{ft}(t-h:t); \Theta^G_{recons})) \quad (7)$$

where $\tilde{Y}_{t+1:t+\tau}$ is the predicted values, and τ is the prediction horizon. $\hat{Y}_{t-h:t}$ are the reconstructed cylinder signals from random faults. Θ^G_{fc} and Θ^G_{recons} are the learnable parameter for G_{fc} and G_{recons}. Then, we can define the dual-RaGAN loss function in Fig.2 as follows.

$$\mathcal{L}_{G\_fc} = E_{Yf\sim Q}[D_{fc}(\tilde{Y}_{t+1:t+\tau} - E_{Yr\sim P} D_{fc}(Y_{t+1:t+\tau})] \quad (8)$$

$$\begin{aligned}\mathcal{L}_{D\_fc} &= \frac{1}{2}(\mathcal{L}_{real\_fc} + \mathcal{L}_{fake\_fc}) \\ &= \frac{1}{2}(E_{Yr\sim Q}[D_{fc}(Y_{t+1:t+\tau}) - E_{Yf\sim P} D_{fc}(\tilde{Y}_{t+1:t+\tau})] \\ &+ E_{Yf\sim Q}[D_fc((\tilde{Y})_{t+1:t+\tau}) - E_{Yr\sim P} D_{fc}(Y_{t+1:t+\tau})])\end{aligned} \quad (9)$$

$$\begin{aligned}\mathcal{L}_{G\_recons} = E_{Yf\sim Q}[D_{recons}(\tilde{Y}_{t+1:t+\tau} \\ - E_{Yr\sim P} D_{recons}(Y_{t+1:t+\tau})]\end{aligned} \quad (10)$$

$$\begin{aligned}\mathcal{L}_{D\_recons} = &= \frac{1}{2}(\mathcal{L}_{real\_recons} + \mathcal{L}_{fake\_recons}) \\ &= \frac{1}{2}(E_{Yr\sim Q}[D_{recons}(Y_{t+1:t+\tau}) \\ &- E_{Yf\sim P} D_{recons}(\tilde{Y}_{t+1:t+\tau})] \\ &+ E_{Yf\sim Q}[D_recons((\tilde{Y})_{t+1:t+\tau}) \\ &- E_{Yr\sim P} D_{recons}(Y_{t+1:t+\tau})])\end{aligned} \quad (11)$$

Based on the predicted and reconstructed, the point-wise mean square error losses $\mathcal{L}_{fc_mse}$ and $\mathcal{L}_{recons_mse}$ are also calculated and accumulated in case to accelerate the model training. Further the fully connected fault recognition heads FC_{cls} are designed based on the concatenation of the learned feature maps $fmap_{fc_dark}$ and $fmap_{recons}$ for forecasting and reconstruction, respectively. The estimated probabilities for the sensors' states can be represented as:

$$\hat{S}_i = softmax(FC^i_{cls}(concate(fmap_{fc_{dark}}, fmap_{recons}))) \quad (12)$$

where $\hat{S}_i$ is the $i^{th}(i \in [1,4])$ cylinder sensor states, FC^i_{cls} is the ith fault recognition head. Then, the classification losses for each sensor $\mathcal{L}_{cls_i}$ are calculated based on the standard cross-entropy function. Then the final classification loss $\mathcal{L}_{cls}$ is calculated as the summation of the four:

$$\mathcal{L}_{cls} = \sum_{i=1}^{4} \mathcal{L}_{cls_i} \quad (13)$$

D. Loss Function

In this study, to achieve the multi-task learning object, we use the popular homoscedastic uncertainty-based approach [50] to weigh multiple loss functions by considering the homoscedastic uncertainty of each task. Specifically, following the definition of the homoscedastic uncertainty weighted loss, we applied a hierarchical multi-task learning framework for the dual-RaGAN generators. First, on the dark side, two sub-loss terms $\mathcal{L}_{G\_FC'}$ and $\mathcal{L}_{G\_RECONS'}$ are calculated separately as follows:

$$\mathcal{L}_{G\_FC'} = \frac{1}{2\sigma_1^2}\mathcal{L}_{G\_fc} + \frac{1}{2\sigma_2^2}\mathcal{L}_{fc\_mse} + \frac{1}{2\sigma_3^2}\mathcal{L}_{cls} + log\sigma_1\sigma_2\sigma_3 \quad (14)$$

$$\mathcal{L}_{G\_RECONS'} = \frac{1}{2\sigma_4^2}\mathcal{L}_{G\_recons} + \frac{1}{2\sigma_5^2}\mathcal{L}_{recons\_mse} + log\sigma_4\sigma_5 \quad (15)$$

Then, the overall loss $\mathcal{L}_{G\_dark}$ for the generator in the dark side can be represented as:

$$\mathcal{L}_{G\_dark} = \frac{1}{2\sigma_6^2}\mathcal{L}_{G\_FC'} + \frac{1}{2\sigma_7^2}\mathcal{L}_{G\_RECONS'} + log\sigma_6\sigma_7 \quad (16)$$

Similarly, we can represent the overall discriminator loss on the dark side as follows:

$$\mathcal{L}_{D\_dark} = \frac{1}{2\sigma_6^2}\mathcal{L}_{D\_FC} + \frac{1}{2\sigma_7^2}\mathcal{L}_{D\_RECONS} + log\sigma_{d1}\sigma_{d2} \quad (17)$$

As there is no GAN used on the light size, the overall loss $\mathcal{L}_{light}$ on the light side is simply based on the mean square error loss $\mathcal{L}_{fc\_mse}$ and the overall classification loss $\mathcal{L}_{cls}$:

$$\mathcal{L}_{light} = \frac{1}{2\sigma_{l1}^2}\mathcal{L}_{fc\_mse} + \frac{1}{2\sigma_{l2}^2}\mathcal{L}_{cls} + log\sigma_{l1}\sigma_{l2} \quad (18)$$

where σ_i is the learnable parameter that represents the homoscedastic uncertainty or the observation noise.

E. Synthesized Fault Injection

In this study, we introduce a novel random fault generation and injection module for the PTSM network to produce eight artificial faults to the sequential cylinder pressure signals. Following the defined tasks for forecasting and reconstruction, we split the observed data into small segments and each segment has the size 50×4 and of 100×4 for the forecasting and reconstruction, respectively. The eight kinds of faults are shown in Fig. 3.

We first define an integer random seed $\theta_n \in [1,4]$ to generate a random number from the discrete uniform distribution which determines n channels in each segment (100×4) will have faults. Subsequently, we generate a random permutation of the integer set $S = [1,2,3,4]$ and select the first n numbers to represent the corresponding channels (front-left, front-right, rear-left, rear-right). Then, for each faulty channel, we randomly select one element from $\theta_c \in \{1,2,3,4,5,6,7,8\}$ to represent the specific faults. If the sequence has no fault,

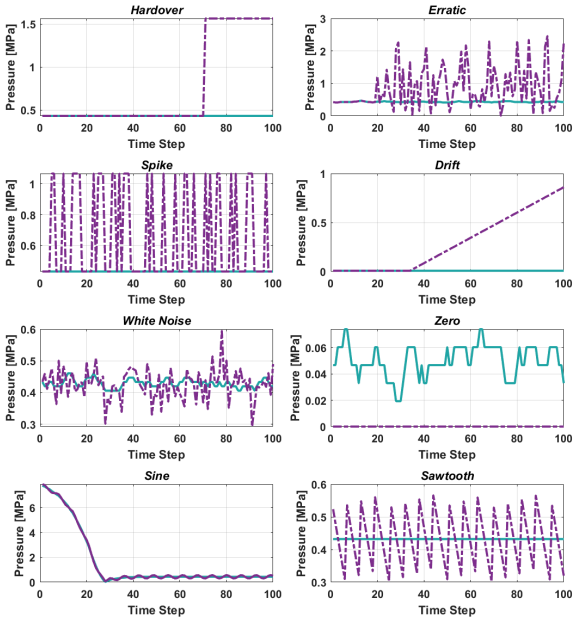


Fig. 3. The eight types of faulty signals. For each sub-graph, the x-axis is the time steps, and the y-axis is the braking pressure with the unit in MPa.

then we assign the state as 0. This setting provides us with the flexibility to control the complexity of the fault, which, in turn, affects the challenges associated with both forecasting and fault classification. The description of the eight synthesized faults is summarized below.

1) Hardover Fault: This fault is defined as the signal returning a value outside its measurable range and saturated at a point.

$$fs(t_{h0} : t_{end}) = \eta_{h0} \times M_s \tag{19}$$

where $fs(t_{h0} : t_{end})$ represents the hard-over fault signal between t_{h0} and t_{end}, where t_{h0} is determined by a uniformly distributed parameter θ_{h0} between 25 to 80 to randomly generate the start point. M_s is the maximum observed value of the signal and $\eta_{h0} \in (1, \infty)$ is a parameter to control the threshold of the fault, we use 1.1 in this study.

2) Erratic Fault: This fault adds a large noise to the signals and increases the magnitude of the signal variance over time.

$$fs(t_e) = rs(t_e) + ee \sim N(0, \delta_a^2) \tag{20}$$

where $fs(t_e)$ is the fault segment that starts from t_e which is also controlled by a uniformly distributed parameter δ_e between 25 to 80. $ee \sim N(0, \delta_a^2)$ is the added normal distribution noise with zero mean and variance $\delta_a^2 = \eta_e \times M_s$, which is determined by the M_s and a control parameter $\eta_e \in [0.2, 0.5]$, we use 0.2 in this study.

3) Spike Fault: the spike fault in this study is defined as:

$$fs(t_d)^N = rs(t_d)^N + \alpha \tag{21}$$

where α is the constant value to simulate the spike fault, and a uniformly distributed parameter θ_s between 10 and 30 (maximum 30 spike faults over the sequence) is used to generate random spike faults for the signal. N is the N^{th} fault points.

4) Drift Fault: we consider a linearly increase drift fault here, which can be represented as:

$$fs(t_d) = rs(t_d) + \eta \times t_d \tag{22}$$

where $fs(t)$ is the drift fault signal, $rs(t_d)$ is the raw signal at step t, η is the slope of the drift that controls the amount of drift, $t_d \in [t_0, t_{end}]$ is the time steps. A starting fault point $t_0 \in [25, 50]$ is randomly sampled for each signal.

5) White Noise Fault: Similar to the erratic fault, we replace the large noise ee with a small additive white Gaussian noise with a $1dB$ signal-to-noise ratio to evaluate the robustness of the PTSM on small noisy signals.

6) Stuck/Zero Fault: This fault simply simulates the zero-reading case:

$$fs(t_s) = 0 \tag{23}$$

where $fs(t_s)$ is the fault segment that starts from t_e which is also controlled by a uniformly distributed parameter θ_s between 25 to 80.

7) Sin Fault: A Sine wave with magnitude 0.1 and period 2π was added to the original signal.

8) Sawtooth Fault: A sawtooth wave with magnitude 0.1 and period 2π was added to the whole observed signal.

F. Implementation Details

We use a vanilla Transformer encoder for the LightNet and DarkNet, with six encoder layers and each layer contains eight heads of multi-head attention module. The decoder layer contains one 1D-Convolutional neural network and one fully connected layer to form the feature map and fully connected layers are used for the multi-task heads in the end. A similar architecture is also applied to the DarkNet, while we double the MHA layers in the DarkNet, and it contains two separate Transformer encoder branches for forecasting and reconstruction tasks, respectively.

We employed a two-stage training framework for model development, as depicted in Fig. 4. Initially, we trained the DarkNet using fully synthesized faulty signals. Consequently, all input sequences to the DarkNet contained faults, while the target sequences for forecasting and reconstruction were normal sequences without faults. Afterwards, we froze the DarkNet parameters and proceeded to train the LightNet using the knowledge acquired from the DarkNet. For this study, we conducted eight NEDC experiments, utilizing the first six cycles for model training and the remaining two cycles for testing. The whole multivariate sequence is split every 150 steps (1.5s) without overlap, which contributes to 3932 and 1481 initial training and testing sequences, respectively. The data are smoothed and normalized to the range of [0-1]. The faults are generated for and injected into the training and testing data separately.

For the LightNet, we concatenated the original training data with a very limited number of faulty sequences (1/10 of the

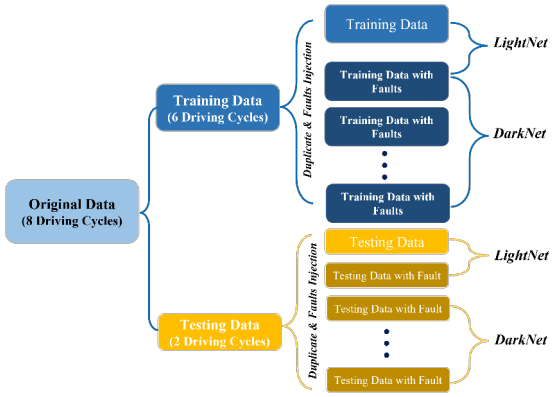


Fig. 4. Data split and generation process for LightNet and DarkNet.

original sequences). For testing, an equal number of faulty sequences was combined with the original testing sequences to simulate critical situations. For the DarkNet, We define $\theta_d \in \{3, 6, 9, 12, 15\}$ to control the amount of synthesized faulty signals for training and testing, which means the number of the faulty sequences on the dark side are θ_d times of the original training ($\theta_d \times 3932$) and testing sequences ($\theta_d \times 1481$). The impact of the amount of training data will be evaluated in Section IV. E. We use the Transformer encoder as the generator, with 12 layers in total and eight self-attention heads in each layer. We use Adam optimizer for model training, with a batch size equals to 128 and maximum epoch equals to 300. The learning rate is 8e-5, and $\beta 1$ and $\beta 2$ equal to 0.5 and 0.999, respectively. We implement our network with PyTorch. The model is trained on local machine with one Nvidia A5000 GPU and Intel i9 CPU.

IV. EXPERIMENT RESULTS

A. Evaluation and Metrics

In the following parts, we evaluate the proposed PTSM framework from various aspects. First, the DarkNet will be evaluated based on the synthesized data. Then, the LightNet and the combined Light-Dark network will be evaluated based on real-world data with limited synthesized faulty signals. Finally, the impact of healthy signal reconstruction and the data volume on the development of the parallel DarkNet and the overall performance will be analyzed to inform the future design of digital twin and parallel learning systems for IVs.

To evaluate the point-wise forecasting and reconstruction loss, a standard mean squared error (MSE) function as shown in (24) is used. The overall classification accuracy (25) and the confusion matrix are reported for the fault classification.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 \tag{24}$$

$$Acc = \frac{Num_c}{Num} \tag{25}$$

where Y_i and $\hat{Y}_i \in \mathbb{R}^(d \times l)$ are the ground truth generated sequences, d is the dimension of the sequence, which equals four (four-cylinder channels) in this study. l equals 50 and 100 for the forecasting and reconstruction cases, respectively.

TABLE I
COMPARISON OF DARKNET WITH BASELINES

Models	Pred[MSE]	Acc	Recons[MSE]
TCN	0.1837	0.8301	0.0764
RNN	0.2511	0.8921	0.0232
LSTM	0.2143	0.8885	0.0255
BiLSTM	0.2027	0.8756	0.0177
GRU	0.2225	0.9100	0.0246
BiGRU	0.2064	0.8854	0.0196
Transformer	0.1720	0.9176	0.0151
Informer	0.1610	0.9190	0.0232
RaGAN_TCN	0.1813	0.8201	0.0343
RaGAN_TCN	0.2171	0.8867	0.0185
RaGAN_LSTM	0.2142	0.8978	0.0160
RaGAN_GRU	0.1956	0.8646	0.0136
RaGAN_Informer	0.1645	0.9195	0.0113
RaGANFormer	0.1695	0.9176	0.0076
RaGANFormer_f	0.1432	0.9229	0.0078

Num_c denotes the number of correctly classified sequences and Num is the overall number of sequences.

B. Evaluation on the Dark Side

In this section, we conduct a comprehensive comparison between the Transformer-based RaGAN model and several common baseline models, including LSTM, GRU, the vanilla Transformer, and Informer. The objective is to evaluate the generalization, adaptability, and advantages of the dual-RaGAN framework over these traditional sequential models. In theory, the dual-RaGAN framework can be adapted to any sequential backbone network, including advanced models like Autoformer and Fedformer. However, such an adaptation may come at the cost of increased computational resources, as these backbones tend to have much larger sizes. For the sake of simplicity, we have chosen to test the framework with light backbone models, such as the vanilla Transformer encoder ($RaGANFormer$, and $RaGANFormer_f$) and Informer ($RaGAN_{Informer}$), among others. The main difference between $RaGANFormer$ and $RaGANFormer_f$ lies in the early fusion between the reconstruction branch and the forecasting branch. $RaGANFormer_f$ ensures that the reconstruction feature map is fused to the forecasting branch at an early stage through a direct connection (as shown in Fig. 1). On the other hand, for $RaGANFormer$, the reconstruction task is still preserved, but there is no knowledge transfer before the feature concatenation. As shown in Table 1, this early fusion in $RaGANFormer_f$ leads to more accurate forecasting and higher classification accuracy (0.1432 and 92.29%) compared to $RaGANFormer$ (0.1695 and 91.76%), although the reconstruction MSE slightly increases. By using these light backbone models, we can still demonstrate the efficiency and effectiveness of the dual-RaGAN framework without incurring excessive computational overhead.

In Table I, the upper part shows results that are developed based on the conventional non-adversarial training method, while the lower part shows the results derived from the dual-

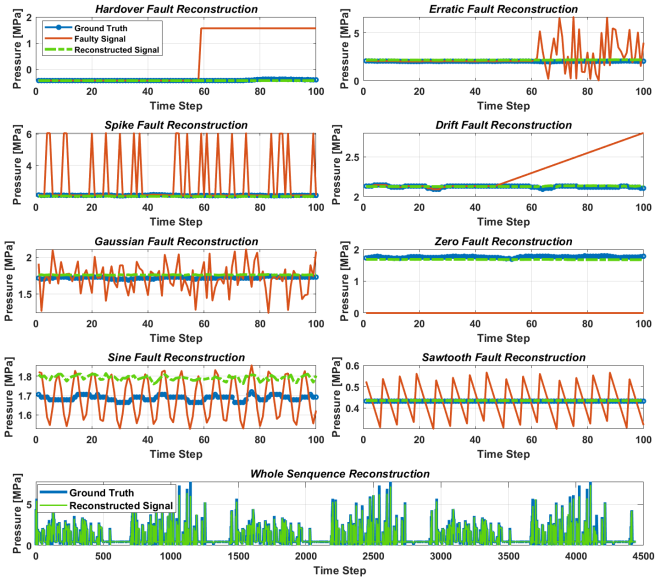


Fig. 5. Exemplar signal reconstruction results for the front-left braking cylinder. The first eight graphs represent the reconstruction for a specific fault. The bottom graph shows the overall reconstruction for the whole braking pressure sequence from the two testing driving cycles.

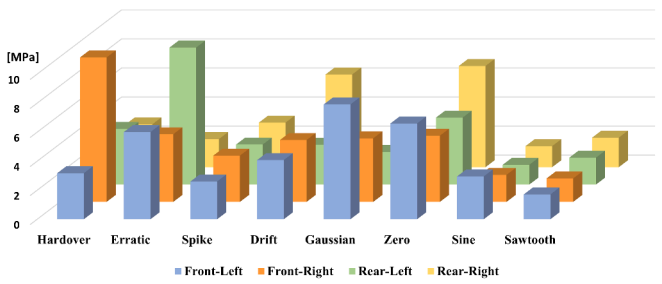


Fig. 6. Average MSE of signal reconstruction for the four braking cylinders.

RaGAN framework. Based on the upper part of Table I it can be found that the convolution-based approach-TCN shows competitive performance in forecasting and fault classification tasks, while there is a significant increase in the MES for healthy sequence reconstruction. The $Informer$ model (with most of the default setup as in [24]) provides the most accurate forecasting and classification performance compared to other baselines, while the reconstruction loss is higher than the vanilla Transformer encoder. When adopting the $Informer$ model into the dual-RaGAN framework, though the forecasting loss slightly increased, the reconstruction error significantly dropped from 0.0232 to 0.0113. Further, with the $Transformer$ network, the $RaGANFormer$ achieved 1.48% and 47.95% increase in the forecasting and reconstruction tasks, respectively, and a 0.58% increase in the fault classification task compared to the $Transformer$ model which is trained without the dual-adversarial approach. Compared with the normal LSTM model, the $RaGAN_{LSTM}$ framework increases the forecasting, classification, and reconstruction accuracy by 0.059%, 0.93%, and 37.28%. Similarly, the $RaGAN_{GRU}$ increases the performance on the forecasting and reconstruction by 12.06% and 44.42%, with a decrease in the classification task by 4.99%, which is caused by the biased multi-task learning process. Based on the comparison results on the forecasting and classification tasks, the healthy signal reconstruction accuracy can be significantly improved with the dual-RaGAN framework.

Examples of forecasting and reconstruction results with $RaGANFormer_f$ is shown in Fig.5. As shown in Fig. 5, in general, the reconstruction network can generate near-real signals which show the $RaGANFormer_f$ can successfully learn the representation of the original health signal even from the faulty training input. The average MSE of the signal reconstruction regarding the eight kinds of faults for the four braking cylinders are shown in Fig. 6. It is shown that, in general, the Zero, Erratic, and Hardcover faults lead to the most significant reconstruction errors among the eight different faults, while the Sine and Sawtooth faults lead to the significant lower MSE compared to the other faults. Here we only try to evaluate the general reconstruction performance on the four-cylinder pressures and analysis the common difficult faults without considering the specific MSE. This is because the MSE of the signal reconstruction can largely depend on the setting of faults in the fault injector part (i.e., using a larger magnitude of the fault will increase the MSE).

In sum, based on the performance comparison in Table I it can be found that 1) the proposed multi-task dual-RGAN network achieved significant improvement in the sequence forecasting, reconstruction, and fault classification tasks compared to the common sequence modelling approach. 2) the dual-RGAN is a general model-agnostic framework that can be adapted to various backbone networks. 3) the dual-RaGAN framework can significantly contribute to the learning of temporal representation and reconstruction of healthy signals from faulty sequences.

C. Evaluation on the Dark Side

In this part, we evaluate the impact of the DarkNet and the corresponding knowledge transfer (with $fmap_{recons}$ and $fmap_{fc_{dark}}$ fusion) on the light side. The testing data in the light side represents a real-world situation where most of the sequences are normal but faulty signals occasionally exist. The comparison between the LightNet and Light-DarkNet is shown in Table II. It is evident that injecting knowledge from the dark side significantly improves the model's performance on both forecasting and fault recognition tasks, despite using the same Transformer backbone. Specifically, the forecasting Mean Squared Error (MSE) is reduced from 0.2390 to 0.1560, resulting in a substantial 34.7% improvement. Moreover, the classification accuracy experiences an 11% increase, climbing from 75.11% to 86.66%. It is shown that even though we freeze the parameters of the DarkNet during the training process, the extracted feature map can still contribute to the network learning for LightNet. The forecasting of the front-left cylinder pressure is shown in Fig. 7. Based on the 50-prediction horizon (0.5s), the Light-DarkNet can make a more accurate prediction of the highly dynamic braking pressure compared to the LightNet only.

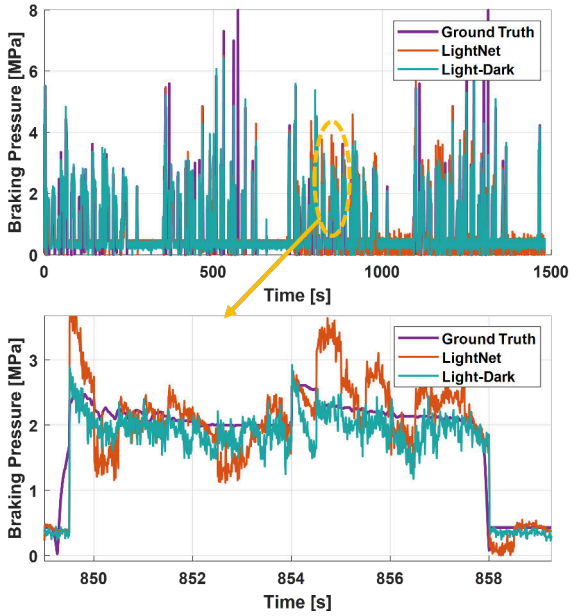


Fig. 7. The real-time signal forecasting for the front-left braking cylinder. The upper graph shows the overall forecasting results on the two-testing driving cycle, and the bottom graph shows the detailed view of a specific region in the upper graph.

TABLE II
COMPARISON OF THE LIGHT-DARK AND LIGHTNET ON THE TESTING DATASET

Models	Pred[MSE]	Acc
Light_only	0.2390	0.7511
Light_Dark	0.1249	0.9531

In Fig. 8, the confusion matrix illustrates the fault classification results using RaGANFormer. Fig. 8(a) shows the classification results using the LightNet, while Fig. 8(b) presents the results obtained with the combined Light-DarkNet.In Fig. 8(a), it can be observed that the normal state and the Hardover state achieve the same recognition accuracy of 95.4% on average across all four cylinders. However, the LightNet performs poorly on the sine wave fault, achieving only 2.1% recognition accuracy, with most samples misclassified into the normal state. A similar situation is observed for the Gaussian noise and sawtooth faults, with 334 and 363 samples misclassified into the normal group, respectively. The LightNet only achieves 7.7% and 7.1% accuracy on these two faults. The results indicate that the Light-DarkNet model significantly improves the fault classification accuracy for most fault categories compared to the LightNet. However, challenges remain in accurately classifying the sine wave, Gaussian noise, and sawtooth faults. Further refinements and enhancements are needed to address these issues and make the fault classification model more robust and reliable.

Compared to the LightNet-only case, the Light-DarkNet model demonstrates significant improvements in the classification results. The Sine and Sawtooth fault classification rates show significant enhancement, achieving 98.1% and

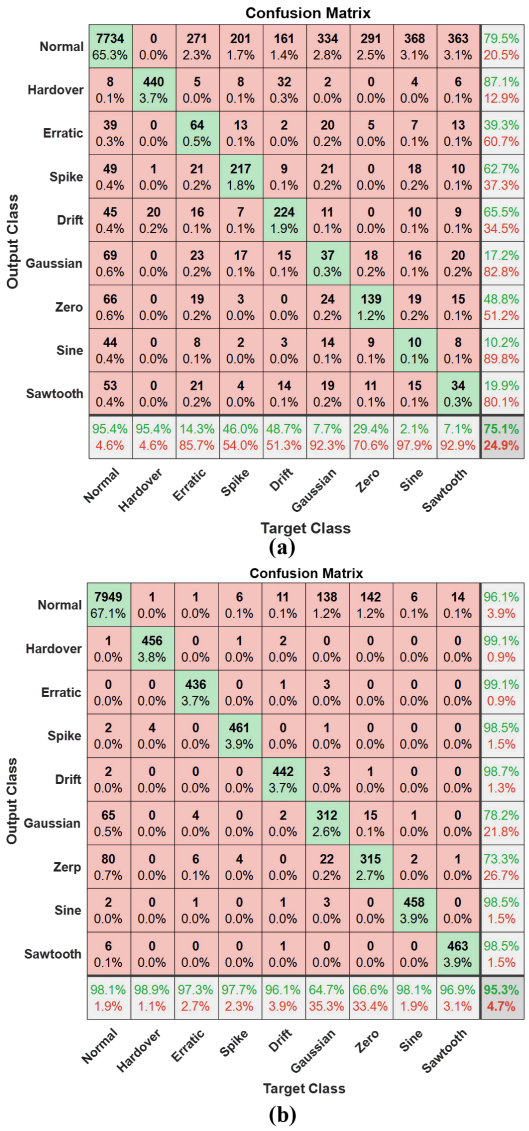


Fig. 8. Confusion matrix for the fault classification results. (a) shows the performance on the testing data with the LightNet only, and (b) shows the performance with the combined Light-DarkNet. The overall samples used to generate these two graphs are the summation of the four cylinders and hence represent the average performance of the four braking cylinders. The precision and the recall rate are shown in the far-right column and bottom row, respectively. The overall accuracy is shown in the bottom right corner.

96.9% accuracy, respectively. On the other hand, the most challenging fault for the Light-DarkNet is Gaussian and Zero noise, achieving only 64.7% and 66.6% accuracy, with a large number of samples misclassified into the normal group.

A common challenge faced by both the LightNet and Light-DarkNet is the misclassification of faulty sequences into the normal group, which can be attributed to data imbalance. Addressing this data imbalance issue is an interesting area for future research, and efficient solutions such as incorporating multi-task learning frameworks could be explored. Additionally, just as the reconstruction case is influenced by the Mean Squared Error (MSE) and specific fault configurations, the classification results can also be impacted by these factors.

TABLE III
PROFILE OF MODEL MEMORY USAGE WITH BATCH-SIZE 128

Model	Model Memory [MB]	Optimizer Memory [MB]	GPU Usage [MB] Training	Inference
DarkNet	1380.8	2741.1	23552.0	3584.0
LightNet	38.9	64.1	5427.2	2150.4
Light-Dark	38.9	64.1	9625.6	7327.8

TABLE IV
COMPARISON OF THE MODEL TRAINING AND INFERENCE SPEED

Modes		Time Cost/Sequence [ms] DarkNet	LightNet	Light-Dark
Training	CPU	88.4	38.3	75.1
	GPU	2.8	2.4	4.6
Inference	CPU	26.4	4.6	59.6
	GPU	2.1	1.9	3.1

To make the classification model more robust, it might be beneficial to design more challenging faulty signals based on adversarial examples [51] and explore complex combined and correlated fault scenarios [52].

The model memory usage is detailed in Table III. To ensure real-time processing, we employed a lighter 1D-Conv decoder for both the LightNet and the Light-DarkNet, effectively reducing the model size and enhancing forecasting performance with real-world small-scale data. While the model and optimizer memory requirements for LightNet and Light-DarkNet are comparable, the Light-DarkNet, due to feature fusion, demands more GPU memory for both training (9.4G) and inference (7.2G). This increased memory utilization is a consequence of information fusion, surpassing even the DarkNet's GPU memory usage for real-time inference (3.5G), despite the latter's larger model size. Similar trends are evident in model training and inference speeds, as presented in Table IV below. Notably, the use of fully connected layers in the DarkNet decoder results in an exceptionally large model size. However we found that a larger model, with a more complex decoder, can be more efficient in handling large-scale synthetic data and lead to higher accuracy in forecasting, reconstruction, and fault classification.

We conducted real-time inference speed tests for DarkNet, LightNet, and Light-DarkNet in the context of whole sequence forecasting. Our evaluation included assessing the training and inference speed on both CPU and GPU platforms. As illustrated in Table III, the LightNet achieved a sequence-level prediction time of 4.6 ms on CPU and 1.9 ms on GPU. Notably, the GPU demonstrated a remarkable performance boost, increasing the inference speed by over 200%. The model training and inference speed decrease when the LightNet queries the DarkNet and incorporates DarkNet's knowledge. As detailed in the table, sequence-level forecasting of the Light-DarkNet using GPU and CPU modes takes 3.1 ms and 59.6 ms, respectively. This indicates that in the CPU mode, a substantial portion of CPU resources is allocated to DarkNet, in contrast to the LightNet-only case with a CPU time of 4.6 ms. It is interesting to see that, even through the Light-DarkNet is much lighter than the DarkNet in terms of the model size, the inference speed of the Light-DarkNet with either CPU or GPU are even lowerer than the DarkNet, which shows the feature fusion between the DarkNet and LightNet require significant processing. In general, considering the CAN bus sampling frequency is 100Hz, the 3.1 ms sequence (50 steps) forecasting speed will be fast enough to make real-time inference.

TABLE V
EVALUATION OF THE RECONSTRUCTION NETWORK

Models	Pred[MSE]	Acc
Dark_Rec	0.1432	0.9229
Dark_No_Rec	0.1783	0.9135
Light_Dark_No_Rec	0.2428	0.8282
Light_Dark_Rec	0.1249	0.9531

In sum, it can be found that injecting the knowledge learnt from the dark side (virtual world) can: (1)dramatically improve sequence forecasting (34.7% improvement) and fault classification accuracy (11.5% improvement) in the light side (real world). (2)Slightly decrease the real-time inference speed from an average of 1.89 ms to 3.12 ms with RTX A5000 GPU.

D. Impact of Data Reconstruction

The impact of data reconstruction and multi-task learning of the dual-RaGAN on both the dark side and the light side is presented in Table III. The results demonstrate that incorporating the reconstruction of healthy signals and employing hierarchical multi-task learning within the dual-RGAN framework significantly improves forecasting and classification accuracy on the dark side. Specifically, when compared to the scenario without reconstruction learning ($Dark\_No\_Rec$), the multi-task learning for the dual-RaGAN reduces forecasting loss from 0.1783 to 0.1432 and increases classification accuracy to 92.29%. Notably the $Dark\_No\_Rec$ case is different from the previous $RaGANFormer$ case as the reconstruction loss is no longer retained in the computational graph.

On the other hand, on the light side scenarios, the reconstruction network has a more substantial impact on task performance. In this case, the LightNet is injected only with the learned forecasting map ($fmap_{fc_{dark}}$) from the DarkNet, and the efficiency of the reconstruction map ($fmap_{fc_{recons}}$) is tested. The results reveal that without the reconstructed knowledge of the normal signals, the forecasting loss increases to 0.2428 from 0.1249, indicating a 50% decrease in performance, while the fault recognition accuracy drops by 13% to 0.8282.

In sum, it can be found that the reconstruction network plays an important role on the dark side and light side through joint learning and direct knowledge injection to improve signal forecasting and state recognition.

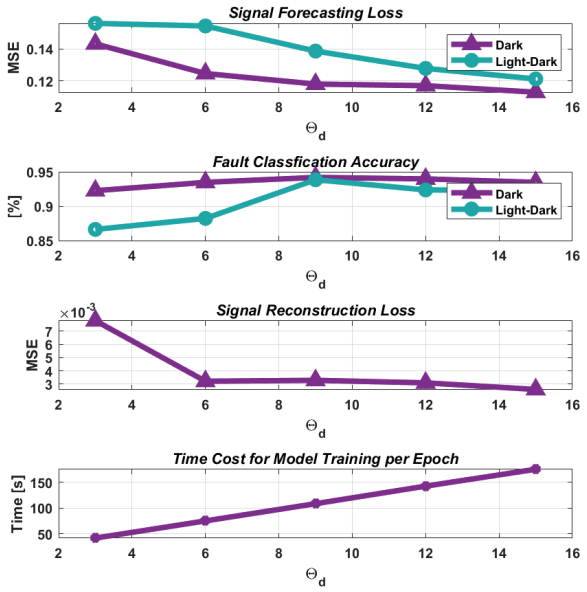


Fig. 9. Evaluation of data volume on dark side and light side (denoted as Light-Dark) in the figure. As the reconstruction is only implemented for the DarkNet, only the dark side evaluation is demonstrated. $\theta_d \in \{3, 6, 9, 12, 15\}$.

E. Impact of Data Volume

We aim to explore the influence of data volume on the dark side and its impact on the light side considering knowledge transfer. As mentioned in Section III.D, we utilize θ_d to regulate the data volume on the dark side. Here, we assess whether increasing the data volume (θ_d) on the dark side leads to improved forecasting and classification results on the light side (real world)

The evaluation results are depicted in Fig. 9. It is shown that with an increase in data volume on the dark side, the forecasting error (MSE loss) consistently decreases. Notably, when θ_d is set to 15, the MSE for the DarkNet and Light-DarkNet reduces to 0.1129 and 0.1212, respectively. However, the variation in classification and reconstruction results is less significant. For fault classification, the highest accuracy is achieved when θ_d equals nine, where the fault classification accuracy for the DarkNet and Light-DarkNet reaches 93.88% and 94.00%, respectively. Subsequently, the classification accuracy remains stable in the range of 92% to 93%. Similarly, the reconstruction loss experiences a significant decrease when θ_d is increased to six. However, after that, the reconstruction MSE loss saturates at around $3e^{-3}$ and eventually reaches $2.6e^{-3}$ with θ_d set to 15. The time cost for model training (every epoch) in the bottom graph of Fig. 9 exhibits a clear linear increase trend can be found in the DarkNet with growing data size (θ_d). The per epoch training takes 42.1s, 109.7s, and 175.3s when θ_d equals to 3, 9, and 15, respectively.

We evaluate the model complexity of the DarkNet and Light-DarkNet by varying the number of encoder layers (1x, 5x, and 10x of the original six layers). The per-epoch GPU usage and inference speed were evaluated with a batch size of 128. As shown in Fig. 10, the model size exhibits linear growth, even with a substantial increase in the number of encoder layers, highlighting the FC decoder as the most heavy component in the DarkNet. GPU usage during model training demonstrates an increase, while it remains relatively stable during inference (7.5G for Light-DarkNet and 4G for DarkNet). The 10x model incurs exceptionally high GPU usage (83G), resulting in an extended training time due to the need of extra resources from the integrated graphic. This value, however, could potentially be significantly reduced with multiple GPUs or a decreased batch size (e.g., the training time for the 10x model decreases to 248s per epoch with a batch size of 16). In summary, the utilization of the Dual-RaGAN structure in the DarkNet leads to a more pronounced increase in GPU resource and time costs during both training and inference compared to the Light-DarkNet.

In sum, the evaluation of data volume reveals several key insights. Increasing the data volume on the virtual side significantly improves the model's performance, and transferring the learned knowledge to the real-world contributes to enhanced forecasting and classification results. However, it is observed that as data volume continues to increase, the network performance in both the dark side and light side eventually reaches a saturation point. It is important to note that in this evaluation, we only varied the data volume while keeping the model size fixed, to understand how data volume influences the virtual and real-world performance. For future research on DT and parallel systems, both data volume and model size can be increased in the virtual world, potentially yielding further performance gains but leading to even higher computational demands.

V. Discussion and Future Works

A. Model Improvement

To demonstrate the effectiveness of the proposed PTSM framework, we use vanilla Transformer backbone in this study. Additionally, we conducted tests using various backbone networks to highlight the robustness of both the PTSM and dual-RaGAN framework. Considering the computational complexity and limitations in long-term sequential representation inherent in the original Transformer, there is potential for further enhancement of the PTSM framework by integrating advanced sequential networks such as the sparse Transformer [49], Autoformer [27], or contemporary networks like TimesNet [29]. Our future objectives include refining the PTSM framework by exploring these advanced sequential networks, considering their computational efficiency and suitability for long-term sequential representations. Furthermore, we plan to further investigate the dual-RaGAN training using common baselines and datasets such as the ETT datasets. This will provide deeper insights into the performance improvements achievable with the proposed time-series models and comparison to existing models.

B. Complex Faults Generation

In this study, we developed a configurable fault generator capable of controlling the complexity of each fault. Our findings underscored the diverse impacts of different faults and

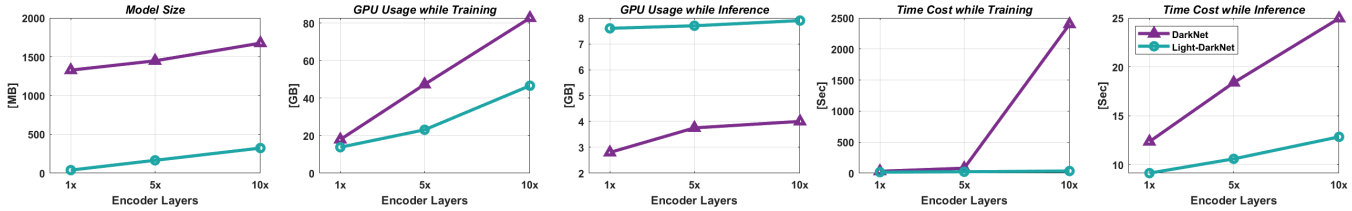


Fig. 10. Evaluation of model complexity for DarkNet and Light-DarkNet with per-epoch performance with various encoder layers and $\theta_d = 3$.

highlighted the challenge posed by insufficient data in real-time vehicle fault detection. The current fault generator operates on a hand-crafted policy to generate faults, primarily in a single mode. However, future work could involve designing a more intricate fault generator by combining various faults and signal frequencies. Additionally, the introduction of adversarial faults using techniques such as the Fast Gradient Signed Method (FGSM) [53] or other adversarial training methods holds promise for enhancing fault detection algorithms for IVs. Acknowledging the constraints posed by data limitations and the sparsity of real-world faults, future research might explore the integration of few-shot learning approaches to address these challenges effectively.Currently we only considered the fault from CAN bus signal, however, fault can also occur in the measurement, communication, and visualization layer with either system failure, malfunction, or cyber-attack, it is interesting to study how a more comprehensive virtual sequential modeling system can be developed with the PTSM framework.

C. Real-time Operation of PTSM

In this study, we demonstrated how the parallel learning framework can enhance real-world time-series modeling when knowledge acquired from the virtual world is seamlessly integrated into real-world scenarios. As an illustrative example, we applied this approach to the challenging task of forecasting braking pressure signals, showcasing its efficacy in handling dynamic and highly uncertain sequences in real-time. While the feature fusion method employed in this project utilizes a simple additive approach, the resulting performance improvements in forecasting and classification are significant. Future research will explore more efficient feature fusion methods, incorporating attention mechanisms to extract valuable information from nuanced and complex data, further enhancing the capabilities of the parallel learning framework.

The study reveals that the Light-Dark network places a significant computational burden, resulting in extremely low inference speeds in CPU mode. Several strategies can be employed to enhance the real-time operation of the PTSM framework. First, drawing inspiration from current large-scale foundation models, a more complex and heavier DarkNet can be developed and implemented on the cloud, thereby further improving vehicle safety and fault detection. The local ego-vehicle would then exclusively carry the LightNet, querying the cloud only when necessary through the Vehicle-to-Cloud communication. To augment data diversity and enrich fault signal information, connected vehicular technologies utilizing a federated learning scheme could be introduced. This approach would involve gathering health and fault signals from vehicles worldwide. Additionally, to expedite feature extraction, encoding the heavy DarkNet could be achieved using methods such as a simple look-up table or a much lighter autoencoder network.

VI. Conclusion

In this study, we present a groundbreaking parallel time series modelling framework designed to facilitate knowledge transfer from the virtual world to the real world. We further introduce a novel multi-task adversarial learning framework, specifically tailored for the proposed dual-RGAN architecture in the PTSM approach. This framework efficiently learns shared representations for signal forecasting, fault classification, and health signal reconstruction. The results of extensive experimental studies reveal several key findings:

1) The proposed dual-RaGAN network serves as an efficient and versatile model-agnostic framework for enhancing time-series forecasting and classification tasks.
2) Incorporating knowledge from the virtual world, derived from various fault scenarios, substantially improves real-world time-series modelling performance.
3) The reconstruction of healthy signals from faulty signals enhances our understanding of the temporary dynamics of time-series signals. This, in turn, contributes to more accurate forecasting and fault classification in both the virtual world and the real world.
4) Augmenting data in the virtual world positively impacts real-world performance, highlighting the importance of expanding the virtual dataset.

In summary, our work introduces a novel Parallel Time Series Modelling framework, specifically designed to address the challenges of complex sequence modelling tasks in scenarios where data scarcity is an issue. To enhance sequence modelling across various tasks, we also present a novel multi-task dual-RaGAN learning framework for complex time series. Our contributions to the field include showcasing the potential of efficient data generation and parallel semi-supervised learning methods in improving complex time-series modelling. The proposed PTSM framework and multi-task RaGAN approach hold promises in advancing parallel systems for intelligent mobilities.

References

[1] W. Lang, *et al.* “Artificial intelligence-based technique for fault detection and diagnosis of EV motors: A review.” *IEEE Trans. Transportation Electrification*, 8.1: 384-406, 2021, doi:10.1109/TTE.2021.3110318.

[2] X. Wang, *et al.* "Active fault tolerant control for unmanned underwater vehicle with actuator fault and guaranteed transient performance." *IEEE Trans. Intelligent Vehicles*, 6.3: 470-479, 2021, doi: 10.1109/TIV.2020.3038785.

[3] J. Chen, *et al.* "ACP-Based Energy-Efficient Schemes for Sustainable Intelligent Transportation Systems." *IEEE Trans. Intelligent Vehicles*, 2023, 10.1109/TIV.2023.3269527.

[4] N. Mohammad, *et al.* "An AI-Assisted Systematic Literature Review of the Impact of Vehicle Automation on Energy Consumption." *IEEE Trans. Intelligent Vehicles*, 2023, doi: 10.1109/TIV.2023.3268300.

[5] Siyu Teng, *et al.* "Motion Planning for Autonomous Driving: The State of the Art and Future Perspectives." *IEEE Trans. Intelligent Vehicles*, 2023, doi: 10.1109/TIV.2023.3274536.

[6] Y. Lin, *et al.* "Mobility 5.0: Smart Logistics and Transportation Services in Cyber-Physical-Social Systems." *IEEE Trans. Intelligent Vehicles*, 2023, doi: 10.1109/TIV.2023.3286995.

[7] L. Chen, *et al.* "Milestones in autonomous driving and intelligent vehicles: Survey of surveys." *IEEE Trans. Intelligent Vehicles* ,8.2: 1046-1056, 2022, doi: 10.1109/TIV.2022.3223131.

[8] J. Zhang, *et al.* "Deep learning based attack detection for cyber-physical system cybersecurity: A survey." *IEEE/CAA Journal of Automatica Sinica,* 9.3: 377-391, 2021, doi: 10.1109/JAS.2021.1004261.

[9] Y. Jin, H. Lei, and Y. Chen. "A time series transformer based method for the rotating machinery fault diagnosis." *Neurocomputing*, 494: 379-395, 2022, doi: 10.1016/j.neucom.2022.04.111 .

[10] C. Kukjin, *et al.* "Deep learning for anomaly detection in time-series data: review, analysis, and guidelines." *IEEE Access*, 9: 120043-120065, 2021, doi: 10.1109/ACCESS.2021.3107975.

[11] J. Lu, *et al.* "Event-Triggered Deep Reinforcement Learning Using Parallel Control: A Case Study in Autonomous Driving." *IEEE Trans. Intelligent Vehicles*,2023, doi: 10.1109/TIV.2023.3262132.

[12] Md. N. Hasan, S. U. Jan, and I. Koo. "Wasserstein GAN-based Digital Twin Inspired Model for Early Drift Fault Detection in Wireless Sensor Networks." *IEEE Sensors Journal*, 2023, doi: 10.1109/JSEN.2023.3272908.

[13] G. EP. Box, and D. A. Pierce. "Distribution of residual autocorrelations in autoregressive-integrated moving average time series models." *Journal of the American Statistical Association*, 65.332: 1509-1526, 1970, doi: 10.2307/2284333.

[14] S. A. Billings. "Nonlinear system identification: NARMAX methods in the time, frequency, and spatio-temporal domains." *John Wiley & Sons,*2013, doi: 10.1002/9781118535561.

[15] Jr. Gardner and S. Everette."Exponential smoothing: The state of the art." *Journal of forecasting*, 4.1: 1-28, 1985, doi: 10.1002/for.3980040103

[16] P. Esling, and C. Agon. "Time-series data mining." *ACM Computing Surveys*, 45.1: 1-34, 2012, doi:10.1145/2379776.2379788.

[17] Bai, Shaojie, J. Zico Kolter, and Vladlen Koltun. "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling." *arXiv preprint arXiv:*1803.01271 2018, doi: 10.48550/arXiv.1803.01271

[18] A. Oord, *et al.* "Wavenet: A generative model for raw audio." *arXiv preprint arXiv:*1609.03499, 2016, doi:10.48550/arXiv.1609.03499.

[19] S. Hochreiter, and J. Schmidhuber, *et al.* "Long short-term memory." *Neural computation*, 9.8: 1735-1780, 1997, doi: 10.1162/neco.1997.9.8.1735.

[20] J. Chung, *et al.* "Empirical evaluation of gated recurrent neural networks on sequence modeling." *arXiv preprint arXiv:*1412.3555, 2014, doi: 10.48550/arXiv.1412.3555.

[21] A. Vaswani, *et al.* "Attention is all you need." *Advances in neural information processing systems*,30, 2017, doi: 10.48550/arXiv.1706.03762.

[22] J. Wei, *et al.* "Emergent abilities of large language models." *arXiv preprint arXiv:*2206.07682, 2022, doi: 10.48550/arXiv.2206.07682.

[23] B. Lim, *et al.* "Temporal fusion transformers for interpretable multi-horizon time series forecasting." *International Journal of Forecasting*37.4: 1748-1764, 2021, doi: 10.1016/j.ijforecast.2021.03.012.

[24] S. Tuli, G. Casale, and N. R. Jennings. "Tranad: Deep transformer networks for anomaly detection in multivariate time series data." *arXiv preprint arXiv:*2201.07284, 2022, doi: 10.48550/arXiv.2201.07284.

[25] S. Li, *et al.* "Enhancing the locality and breaking the memory bottleneck of transformer on time series forecasting." *advances in neural information processing systems*, 32, 2019, doi: 10.48550/arXiv.1907.00235.

[26] H. Zhou, *et al.* "Informer: Beyond efficient transformer for long sequence time-series forecasting." *Proceedings of the AAAI conference on artificial intelligence.*, Vol. 35. No. 12. 2021, doi: 10.48550/arXiv.2012.07436.

[27] H. Wu, *et al.* "Autoformer: Decomposition transformers with auto-correlation for long-term series forecasting." *Advances in Neural Information Processing Systems*, 34: 22419-22430, 2021, doi: 10.48550/arXiv.2106.13008.

[28] T. Zhou, *et al.* "Fedformer: Frequency enhanced decomposed transformer for long-term series forecasting." *International Conference on Machine Learning*,PMLR, 2022, doi: 10.48550/arXiv.2201.12740.

[29] H. Wu, *et al.* "Timesnet: Temporal 2d-variation modeling for general time series analysis." *arXiv preprint arXiv*:2210.02186, 2022, doi: 10.48550/arXiv.2210.02186.

[30] Y.Xing, and C. Lv. "Dynamic state estimation for the advanced brake system of electric vehicles by using deep recurrent neural networks." *IEEE Trans. Industrial Electronics*, 67.11: 9536-9547, 2019, doi: 10.1109/TIE.2019.2952807.

[31] D. Capriglione, *et al.* "Online fault detection of rear stroke suspension sensor in motorcycle." *IEEE Trans. Instrumentation and Measurement*, 68.5: 1362-1372, 2019, doi: 10.1109/TIM.2019.2905945.

[32] L. Zhu, *et al.* "Distributed cooperative fault-tolerant control of high-speed trains with input saturation and actuator faults." *IEEE Trans. Intelligent Vehicles*, 8.2: 1241-1251, 2022, doi: 10.1109/TIV.2022.3168550.

[33] X. Mo, *et al.* "Multi-agent trajectory prediction with heterogeneous edge-enhanced graph attention network." *IEEE Trans. Intelligent Transportation Systems*, 23.7: 9554-9567, 2022, doi: 10.1109/TITS.2022.3146300.

[34] MS. H. Lipu, *et al.* "Real-time state of charge estimation of lithium-ion batteries using optimized random forest regression algorithm." *IEEE Trans. Intelligent Vehicles*, 8.1: 639-648, 2022, doi: 10.1109/TIV.2022.3161301.

[35] M. R. Hajidavalloo, *et al.* "Artificial intelligence-based technique for fault detection and diagnosis of EV motors: A review." *IEEE Trans. Intelligent Vehicles*,2023, doi: 10.1109/TIV.2023.3275952.

[36] A. R. Javed, *et al.* "NMPC-Based Integrated Thermal Management of Battery and Cabin for Electric Vehicles in Cold Weather Conditions." *IEEE Trans. Intelligent Transportation Systems*,22.7: 4291-4300, 2020, doi: 10.1109/TITS.2020.3025875.

[37] F. Van Wyk, *et al.* "Real-time sensor anomaly detection and identification in automated vehicles, *IEEE Trans. Intelligent Transportation Systems*.21.3: 1264-1276, 2019, doi: 10.1109/TITS.2019.2906038.

[38] Y. Ji, and H. Lee. "Event-based anomaly detection using a one-class svm for a hybrid electric vehicle." *IEEE Trans. Vehicular Technology*, 71.6: 6032-6043, 2022, doi: 10.1109/TVT.2022.3165526.

[39] X. Zhang, *et al.* "A novel battery abnormality detection method using interpretable Autoencoder." *Applied Energy*, 330: 120312, 2023, doi: 10.1016/j.apenergy.2022.120312.

[40] Y. Wang, *et al.* 'An event-triggered scheme for state estimation of preceding vehicles under connected vehicle environment." *IEEE Trans. Intelligent Vehicles*, 8.1: 583-593, 2022, doi: 10.1109/TIV.2022.3181330.

[41] S. Safavi, *et al.* "Multi-sensor fault detection, identification, isolation and health forecasting for autonomous vehicles." *Sensors*, 21.7: 2547, 2021, doi: 10.3390/s21072547.

[42] J. Geyer, *et al.* "A2d2: Audi autonomous driving dataset." *arXiv preprint arXiv:*2004.06320, 2020, doi: 10.48550/arXiv.2004.06320.

[43] M. Abboush, *et al.* "Intelligent fault detection and classification based on hybrid deep learning methods for hardware-in-the-loop test of automotive software systems." *Sensors*, 22.11: 4066, 2022, doi: 10.3390/s22114066.

[44] Z. Hu, *et al.* "Review and perspectives on driver digital twin and its enabling technologies for intelligent vehicles." *IEEE Trans. on Intelligent Vehicles*, 2022, doi: 10.1109/TIV.2022.3195635.

[45] Z. Wang, C. Lv, and FY. Wang, "A New Era of Intelligent Vehicles and Intelligent Transportation Systems: Digital Twins and Parallel Intelligence." *IEEE Trans. Intelligent Vehicles*, 2023, doi: 10.1109/TIV.2023.3264812.

[46] Liu, Shimin, *et al.* "AAdaptive reconstruction of digital twins for machining systems: A transfer learning approach." *Robotics and Computer-Integrated Manufacturing*,78: 102390, 2022, doi: 10.1016/j.rcim.2022.102390.

[47] F. Y. Wang. "Parallel control and management for intelligent transportation systems: Concepts, architectures, and applications." *IEEE Trans. Intelligent Transportation Systems*, 11.3: 630-638, 2010, doi: 10.1109/TITS.2010.2060218.

[48] F. Y. Wang, *et al.* "Parallel driving in CPSS: A unified approach for transport automation and vehicle intelligence." *IEEE/CAA Journal of Automatica Sinica* ,4.4: 577-587, 2017, doi: 10.1109/JAS.2017.7510598.

[49] S. Wu, *et al.* "Adversarial sparse transformer for time series forecasting." *Advances in neural information processing systems*,33: 17105-17115, 2020.

[50] A. Kendall, Y. Gal, and R. Cipolla. “Multi-task learning using uncertainty to weigh losses for scene geometry and semantics.” *Proceedings of the IEEE conference on computer vision and pattern recognition.*2018, doi: 10.48550/arXiv.1609.03499

[51] J. Zhang, and L. Chen. “Adversarial examples: Opportunities and challenges.” *IEEE Trans. Neural Networks and Learning Systems* 31.7: 2578-2593, 2019, doi: 10.1109/TNNLS.2019.2933524.

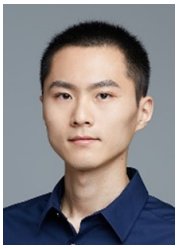
[52] Özgür Gültekin, *et al.* “Multisensory data fusion-based deep learning approach for fault diagnosis of an industrial autonomous transfer vehicle.” *Expert Systems with Applications*200: 117055, 2022, doi: 10.1016/j.eswa.2022.117055.

[53] I. Goodfellow, J. Shlens, and C. Szegedy. “Explaining and harnessing adversarial examples.” *arXiv preprint arXiv:*1412.6572 (2014). doi:10.48550/arXiv.1412.6572.

Yang Xing Yang Xing received his Ph. D. degree from Cranfield University, UK. He is currently a Lecturer in Applied Artificial Intelligence at Cranfield University. He was a Research Associate at the University of Oxford and a Research Fellow at Nanyang Technological University. His research interests include human-centred autonomous systems, intelligent multi-agent collaboration, and machine learning.

Zhongxu Hu Zhongxu Hu received a mechatronic Ph.D. degree from the Huazhong University of Science and Technology of China, in 2018. He was a senior engineer at Huawei. He is currently a research fellow at the School of Mechanical and Aerospace Engineering of Nanyang Technological University in Singapore. His current research interests include human-machine collaboration, human cyber-physical systems, human digital twins, and deep learning applied in intelligent vehicle and industrial scenarios.

Peng Hang Peng Hang is a Research Professor at the Department of Traffic Engineering, Tongji University, Shanghai, China. He received his Ph.D. degree with the School of Automotive Studies, Tongji University, Shanghai, China, in 2019. From 2020 to 2022, he served as a Research Fellow with the School of Mechanical and Aerospace Engineering, Nanyang Technological University, Singapore. His research interests include vehicle dynamics and control, decision-making, motion planning and motion control for autonomous vehicles.



Chen Lv Chen Lv is currently an assistant professor at Nanyang Technology University, Singapore. He received a Ph.D. degree at the Department of Automotive Engineering, Tsinghua University, China in 2016. From 2014 to 2015, he was a joint Ph.D. researcher at EECS Dept., University of California, Berkeley. His research focuses on cyber-physical systems, hybrid systems, advanced vehicle control, and intelligence, where he has contributed over 40 papers and obtained 11 granted China patents.

2023-12-13

Learning from the Dark Side: a parallel time series modelling framework for forecasting and fault detection on intelligent vehicles

Xing, Yang

IEEE

Xing Y, Hu Z, Hang P, Lv C. (2024) Learning from the Dark Side: a parallel time series modelling framework for forecasting and fault detection on intelligent vehicles. IEEE Transactions on Intelligent Vehicles, Volume 9, Issue 2, February 2024, pp. 3205-3219

https://doi.org/10.1109/TIV.2023.3342648