Mechanical Systems and Signal Processing 226 (2025) 112296

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

journal homepage: www.elsevier.com/locate/ymssp

Semi-supervised bridge indirect structural health monitoring using Isolation Distributional Kernels

G. Tyler [a], S. Luo [a], A. Calderon Hurtado [b], M. Makki Alamdari [b,*]

[a] School of Computer Science and Engineering, University of New South Wales, Sydney, 2052, NSW, Australia
[b] Center for Infrastructure Engineering and Safety, School of Civil and Environmental Engineering, University of New South Wales, Sydney, 2052, NSW, Australia

ARTICLE INFO

Communicated by O. Avci

Keywords:
Isolation Distributional Kernel
Probability distributions
Indirect structural health monitoring
Bridge monitoring
Damage detection
Semi-supervised learning

ABSTRACT

This paper addresses the problem of structural health monitoring (SHM) involving the observation of bridges using periodically sampled acceleration responses from an instrumented vehicle as a method to detect changes in its structural integrity. It applies Isolation Distributional Kernels (IDK) with a finite and computational feature map to obtain accurate classification and severity quantization using probability distributions. This is achieved through the kernel expansion of the data's dimensionality, providing an implicit feature map framework for detecting abnormalities in higher-order space. The signals first undergo preprocessing with frequency filtering, averaging, and scaling. The unknown and benchmark signals are then compared to calculate an individual similarity score. The performance of this method is assessed through numerical simulation using a half-car model and a simply-supported beam, where various bridge states, including healthy and different percentages of damage severity, are considered. These findings are then experimentally validated from data generated from a scaled bridge within a laboratory. The proposed framework has three major advantages compared to traditional modal and non-modal frameworks. First, a new application of a novel IDK approach can successfully use drive-by monitoring for progressive damage assessment. Second, the approach works with a limited dataset due to the minimal training required. Third, the model is more efficient, obtaining linear runtime complexity as the sample size is increased with reduced pre-processing requirements compared to comparative methods. These benefits of the IDK approach to indirect SHM successfully overcome the challenges of large-scale data acquisition in drive-by monitoring, providing an accurate and computationally effective solution.

1. Introduction

The structural health of bridges is a vital global concern due to the risk of collapsed bridges to community safety. This makes the imperative need for efficient and cost-effective inspection tools apparent. However, traditional *direct* structural health monitoring (SHM) involves the installation of numerous sensors on the bridge, requiring inefficient data management, expensive installation costs, and maintenance requirements [1]. *Indirect* methods involving vehicle-mounted sensors have gained global recognition for their potential to rectify these issues, providing a more economically feasible and computationally manageable strategy [2,3]. This approach is motivated by the need for drive-by inspection frameworks on short to medium-span bridges which are likely to be

* Corresponding author.
E-mail address: m.makkialamdari@unsw.edu.au (M. Makki Alamdari).

https://doi.org/10.1016/j.ymssp.2024.112296
Received 1 November 2024; Received in revised form 19 December 2024; Accepted 29 December 2024
Available online 14 January 2025

poorly maintained due to budgetary and technological constraints [2]. Due to the dynamic coupling between the vehicle and the bridge, known as the vehicle-bridge interaction (VBI), the modal properties of the bridge can be extracted solely by the response of the vehicle passing over it, and the health state can be inferred [2]. To this aim, extensive research has been conducted using two main groups of methods: (1) traditional time series and frequency domain modal approaches; and (2) utilizing emergent non-modal dimensionality reduction and deep learning methods [4,5].

In the first category of methods, the measured vehicle acceleration response is used to determine the modal parameters of the bridge. Yang et al. [6] demonstrated that the bridge frequency is contained within and can be extracted from the vehicle's acceleration spectrum once the shifting effect is accounted for, with improved visibility in the case of higher speed and at resonance. Conversely, McGetrick et al. [7] utilized the simplified quarter-car model to extract the natural frequencies, demonstrating increased accuracy at lower speeds and upon smoother road profiles. Subsequently, Yang et al. [1] explored filtering techniques to enhance bridge frequency visibility, with a singular spectrum analysis with a band-pass filter providing the most promising results. Despite the successful identification of the modal parameters from measured signals from vehicle-mounted sensors, four major challenges remain. First, the detected modal frequencies are a measure of the stiffness of the bridge rather than a sensitive indicator of damage [2]. In other words, no indication of damage may be inferred just by investigating the extracted modal parameters. Second, these parameters are susceptible to environmental conditions and operational factors such as temperature with variations potentially more significant than those created by even severe damage [2]. Third, data acquisition requires manually driving a car across a bridge at low speeds to obtain sufficient resolution for each signal regardless of the type of post-processing method [4]. Finally, substantial quantities of data are required for methods to be robust to the complete dynamics of the system due to the additional interaction effects, increasing the costs associated with data acquisition [4].

Due to these shortcomings, the second group of methods focused on non-modal data-driven approaches which lack underlying assumptions about the structural system [2]. Some studies have used various dimensionality reduction techniques to extract differentiating features from the acceleration signals. However, these methods need more clarity regarding the physical interpretations than traditional modal approaches [8]. Mei et al. [9] used Mel-frequency cepstral coefficients (MFCCs) and principal component analysis (PCA) to successfully detect and determine the severity of damage using Euclidean distance, verified on numerical analysis and laboratory experiments. Alternatively, the novel approach of combining nonlinear dimensionality reduction with non-parametric clustering algorithms is used by Cheema et al. [10] on numerical and experimental case studies to separate bridge states successfully.

Deep machine learning approaches have also been utilized as a non-modal alternative to machine learning models including neural network and autoencoder approaches. Abdeljaber et al. [11], Sun et al. [12] and Hajializadeh [13] used the computationally inexpensive one-dimensional Convolutional Neural Networks (CNNs) in a supervised manner, demonstrating substantial accuracy in detecting and classifying damage-driven changes. Alternatively, Malekjafarian et al. [14] used an artificial neural network (ANN) on both the vehicle accelerations and the Discrete Fourier Transform (DFT) of the signal to successfully detect damage despite various road roughness profiles and measurement noise. Autoencoders subsequently emerged demonstrating superiority in detecting and classifying severity compared to PCA due to their dimensionality reduction combined with neural network architecture [4].

Despite these successful applications, these approaches have several shortcomings. First, these results are limited by often employing a supervised approach which is unattainable in practice. Second, they are limited by their limited data efficiency, requiring a large number of training examples to obtain sufficient accuracy and tune their numerous hyperparameters. Finally, they are restricted by their use of simulated datasets, limiting their practical interpretation [1,2,6,7].

An alternative approach using a higher-order probability distribution approach has yet to be explored. This approach operates under the assumption that data is not linearly separable in its original space but becomes separable in higher dimensions in the $\mathbb{R}$ domain, capturing the intricate relationships between data points. This dimensionality increase is achieved using kernels, expanding the data's dimensionality without requiring additional data [15]. While probability distributional methods often demand an explicitly defined feature map, sometimes on infinite dimensionality, they inherently provide a framework for detecting similarity in higher-order spaces. Kernel methods can operate in high-dimensional spaces without direct computation through the kernel tree. Typically, selected kernels, such as Gaussian kernels, exhibit at least quadratic time complexity. Nevertheless, efficiency can be enhanced through kernel approximation techniques like Random Fourier and Nyström approximation [15]. The essential advantage of such practices in SHM is the absence of explicit learning requirements and the built-in framework for similarity detection [16]. The proposition made by Ting et al. [5] advocates for this treatment in the context of time series anomaly detection, illustrating the empirical advantages of distributional methods in terms of accuracy and computational efficiency. This efficiency is improved with algorithms that use an exact and finite-dimension feature map, enabling it to handle large-scale datasets that Gaussian kernel methods struggle with [16]. Furthermore, these alternative kernels, including Isolation Distributional Kernels (IDK), exhibit data-dependent behavior, enhancing their classification accuracy compared to data-independent alternatives. Specifically, the IDK approach is characterized by minimal assumptions, allowing for its versatile application, making it an attractive alternative for practical applications of SHM [16].

In comparison to modal and non-modal approaches, the IDK algorithm offers equivalent accuracy whilst improving computational efficiency and ease of deployment. This advantage arises due to its kernel approach that requires minimal use of reference data and the number of parameters which helps mitigate costs of data acquisition in SHM [5]. By detecting similarities in higher-dimensional space, the IDK approach is theoretically more robust to handle real-world conditions characterized by noisy data and complex system dynamics [4,5]. It can also be applied in a semi-supervised manner, overcoming the main limitation of earlier methodologies that required labeled data, which if not impossible, is difficult to achieve in practice [1,2,6,7].

In this paper, it is predicated that these limitations associated with established approaches can be mitigated by considering each time-series signal as an independent and identically distributed (iid) point in higher-order kernel-mapped space, utilizing the

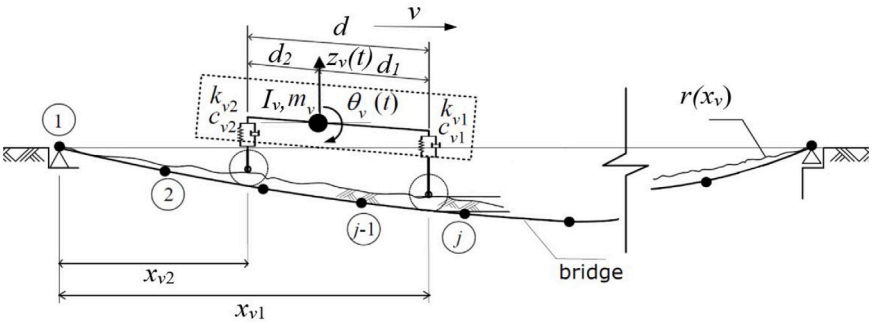


Fig. 1. Illustration of the VBI model adopted for numerical simulation [4].

preprocessing steps and datasets detailed in Calderon Hurtado et al. [4]. The initial step involves a numerical study, serving as a proof of concept. It is then proposed that a distributional-based analysis using IDK be employed to calculate a similarity score between signals to classify the severity of damage, with the detection of bridge damage states relying on proximity to other data points and the severity being quantified based on proximity to groups. Practical applications of the model are subsequently verified through the use of experimentally obtained data, establishing the approach's feasibility. Comparison is also drawn with a state-of-the-art method using Adversarial Autoencoder (AAE) in terms of runtime efficiency, computational accuracy, and sample size [4]. This paper's strengths lie its ability to achieve accurate classification with a small sample size and consequent linear runtime without necessitating extensive training as well as a decreased amount of signal averaging compared to a competing framework using AAE [5,16]. Such findings are imperative to the field of SHM due to the difficulties and costs associated with data obtainment.

The paper's structure is as follows: Section 2 outlines the VBI model, Section 3 delineates the proposed IDK-based damage detection framework, and Section 4 outlines this proposed framework for SHM. Section 5 presents the numerically acquired dataset and describes the damage detection results using this including assessment of the algorithm's equivalent accuracy, decreased dataset size, and improved runtime and computational efficiency compared to AAE. Similarly, Section 6 discusses the acquisition of and results obtained from the experimental datasets. Section 7 compares these findings to the AAE approach in terms of the effects of signal averaging and performance. Finally, Section 8 offers overall conclusions and suggests potential avenues for future research.

2. VBI model

Indirect SHM based on drive-by bridge inspection collects data from sensors mounted on a passing vehicle, such as accelerometers, and processes these signals to achieve an accurate and automated assessment of structural integrity. At each time step $m \in [1 : M]$ as the vehicle passes over the bridge, there is an observation $x_m \in \mathbb{R}$. Assume the vehicle drives over the bridge multiple times, n, at a constant speed. For each of these $j \in [1 : n]$ signals from n passes across the bridge, the signal can be represented as a time series, $\mathbf{G}_j = [x_1, x_2, \ldots, x_M]_j$, for each of the M time steps. These observations arise from the dynamic interaction between the vehicle and the bridge, offering valuable insights into the bridge's condition. This section delves into the dynamic coupling equations that illustrate the interaction of a moving vehicle over a bridge using a half-car model, serving as a proof of concept. The resulting simulated dataset is subsequently utilized in the forthcoming evaluations of the proposed algorithm as detailed in Section 5. This study employs a half-car model to establish the dynamic coupling equations between a vehicle and a simply supported Euler–Bernoulli beam. This model is chosen due to its simplicity and suitability as a proxy for most commercially available vehicles traveling on random road profiles and is widely used in the area of indirect SHM [4,17,18].

As shown in Fig. 1, the half-car model represents the vehicle as two axles - the front and the rear - treated as a single sprung mass and connected by a rigid bar with rotational inertia I_v. The vertical displacement of the front and rear suspensions can be derived from the two degrees of freedom in the model; the bouncing vibration, $z_v(t)$, and pitching vibration $\theta_v(t)$. Additionally, the model incorporates parameters such as combined tire and suspension stiffness, k_{v_i}, damping, c_{v_i}, for the ith axle (i = 1,2), and d_i, representing the distance from the ith axle to the center of gravity. The distance between the two axles is denoted as d and vehicle mass as m_v. The model assumes constant velocity, denoted as v, as the vehicle moves along the bridge. The bridge surface, on the other hand, is independently modeled with a stochastic road profile, $r(x_{v_i})$, where x_{v_i} signifies the location of the ith axle.

Eqs. (1) and (2) depict the dynamic equations governing this vehicle model [4].

$$\begin{aligned} m_v \ddot{z}_v(t) &+ c_{v_1}[\dot{z}_v(t) + d_1\dot{\theta}_v(t) - \{\boldsymbol{\iota}_b(x_{v_1})\}^T\{\dot{\mathbf{Z}}_b(t)\} + vr'(x_{v_1})] \\ &+ c_{v_2}[\dot{z}_v(t) - d_2\dot{\theta}_v(t) - \{\boldsymbol{\iota}_b(x_{v_2})\}^T\{\dot{\mathbf{Z}}_b(t)\} + vr'(x_{v_2})] \\ &+ k_{v_1}[z_v(t) + d_1\theta_v(t) - \{\boldsymbol{\iota}_b(x_{v_1})\}^T\{\mathbf{Z}_b(t)\} + r(x_{v_1})] \\ &+ k_{v_2}[z_v(t) - d_2\theta_v(t) - \{\boldsymbol{\iota}_b(x_{v_2})\}^T\{\mathbf{Z}_b(t)\} + r(x_{v_2})] = 0 \end{aligned} \tag{1}$$

$$
\begin{aligned}
&I_v\ddot{\theta}_v(t) + d_1 c_{v_1}[\dot{z}_v(t) + d_1\dot{\theta}_v(t) - \{\boldsymbol{\iota}_b(x_{v_1})\}^T\{\dot{\mathbf{Z}}_b(t)\} + vr'(x_{v_1})] \\
&- d_2 c_{v_2}[\dot{z}_v(t) - d_2\dot{\theta}_v(t) - \{\boldsymbol{\iota}_b(x_{v_2})\}^T\{\dot{\mathbf{Z}}_b(t)\} + vr'(x_{v_2})] \\
&+ d_1 k_{v_1}[z_v(t) + d_1\theta_v(t) - \{\boldsymbol{\iota}_b(x_{v_1})\}^T\{\mathbf{Z}_b(t)\} + r(x_{v_1})] \\
&- d_2 k_{v_2}[z_v(t) - d_2\theta_v(t) - \{\boldsymbol{\iota}_b(x_{v_2})\}^T\{\mathbf{Z}_b(t)\} + r(x_{v_2})] = 0
\end{aligned} \tag{2}
$$

where $\{\boldsymbol{\iota}_b(x_{v_i})\}$ and $\{\mathbf{Z}_b(t)\} = \{Z_{b,1}(t), Z_{b,2}(t), \ldots, Z_{b,i}(t), \ldots, Z_{b,n}(t)\}$ are the column vectors of the polynomial interpolation displacement functions at the contact point of the ith axle and the nodal response of the bridge system respectively, and n is the number of degrees of freedom for the bridge. Prime and overdot represent derivatives concerning space and time. The displacement interpolation functions $\{\boldsymbol{\iota}_b(x_{v_i})\}$ are cubic Hermite polynomials for an Euler Bernoulli beam, that aim to interpolate the nodal displacements $\{\mathbf{Z}_b(t)\}$ to any location between the nodes of the elements. Further information on the finite elements analysis of a beam-type structure can be found in [19]. Eq. (3) represents the governing set of equilibrium equations for the bridge system with mass, $[\mathbf{M}_b]_{n\times n}$, stiffness, $[\mathbf{K}_b]_{n\times n}$, and damping, $[\mathbf{C}_b]_{n\times n}$, formed using the finite element approach

$$
[\mathbf{M}_b]\{\ddot{\mathbf{Z}}_b(t)\} + [\mathbf{C}_b]\{\dot{\mathbf{Z}}_b(t)\} + [\mathbf{K}_b]\{\mathbf{Z}_b(t)\} + \{\boldsymbol{\iota}_b(x_{v_1})\}\mathbf{R}_1(t) + \{\boldsymbol{\iota}_b(x_{v_2})\}\mathbf{R}_2(t) = 0 \tag{3}
$$

where $\mathbf{R}_1(t)$ and $\mathbf{R}_2(t)$ are the respective contact forces at axle locations x_{v_1} and x_{v_2} as defined in Eqs. (4) and (5) and g is the gravitational acceleration.

$$
\mathbf{R}_1(t) = -c_{v_1}[\dot{z}_v(t) - \{\boldsymbol{\iota}_b(x_{v_1})\}^T\{\dot{\mathbf{Z}}_b(t)\} + vr'(x_{v_1})] - k_{v_1}[z_v(t) - \{\boldsymbol{\iota}_b(x_{v_1})\}^T\{\mathbf{Z}_b(t)\} + r(x_{v_1})] + (d_2/d)m_v g \tag{4}
$$

$$
\mathbf{R}_2(t) = -c_{v_2}[\dot{z}_v(t) - \{\boldsymbol{\iota}_b(x_{v_2})\}^T\{\dot{\mathbf{Z}}_b(t)\} + vr'(x_{v_2})] - k_{v_2}[z_v(t) - \{\boldsymbol{\iota}_b(x_{v_2})\}^T\{\mathbf{Z}_b(t)\} + r(x_{v_2})] + (d_1/d)m_v g \tag{5}
$$

Combining Eqs. (1), (2), and (3) gives the coupled VBI system equations [18]. The governing dynamic equations can be solved iteratively using time integration methods such as Newmark-Beta [2]. These Eqs. (1)–(3) are used to simulate the numerical dataset, with a 5% random white noise added to all acquired signals $\mathbf{G}_j$.

The road surface profile is modeled using a real-valued stationary Gaussian process based on ISO8608 similarly described in [20] using a class A profile with $G_d(n_0) = 16 \times 10^6 \mathrm{m}^3$ [4]. To model the damaged beam, the simplified crack model presented by Sinha et al. [21] is used to simulate damage where additional details of this method can be found in [21].

3. IDK approach for anomaly detection

Real-world measurements can often be represented as groups of data points generated from one or more probability distributions [16]. Distributional kernels provide a powerful method for comparing these distributions and identifying underlying patterns. This section outlines how kernels and associated feature maps are used in anomaly detection problems by first explaining the application of the IDK algorithm to any pair of point measurements $x, y \in D \subseteq \mathbb{R}$ with feature space $\mathcal{X} \in \mathbb{R}$ sampled from unknown probability distribution $\mathcal{P}_D$. The paper will then move on to the time series application by considering each $\mathbf{G}_j \in D$ as a point $x \in \mathbb{R}^M$ and demonstrating this approach for a simple toy example of time series data. This approach involves parameters t and ψ which refer to the number of partitions of the dataset and the number of isolating hyperspheres respectively which are further explained in 3.2.

3.1. Kernel-based anomaly detection using feature maps

Traditional point-to-point measures between points $x, y \in \mathbb{R}$ include the extensively used Euclidean distance and Gaussian kernels methods. Kernels $\kappa(x, y)$ are a similarity measure between two points x and y. When applied to signal processing, $\kappa(x, y)$ acts as a form of pattern recognition, identifying distinguishing features of the data. These features form a set of rules that define a healthy group, with signals that contain more of these distinct features having a higher similarity score, α_j, and those with fewer having a lower α_j. This approach allows for operation in high-dimensional kernel space without explicit mapping between feature and kernel space, minimizing the resulting dimension in kernel space and, consequently, computational cost [5].

A kernel is selected by taking into account task-specifics, data characteristics, domain expertise, and computational efficiency. This choice is often verified by cross-validation methods, however, it remains that there is no general rule or approach to selecting the optimal kernel although approaches such as reformulation as a tractable optimization problem and meta-learning approaches have been considered [22–24]. These kernels can be classified into two main types: data-independent and data-driven kernels. Data-independent distributional kernels such as Gaussian or polynomial kernels have been extensively used in machine learning methods such as Kernel Mean Embedding (KME) and support vector machines (SVM) [5,22,24]. However, these methods are limited by the assumption of fixed data distribution, their inability to capture complex or unknown patterns, and their feature map has intractable dimensionality [5]. This is problematic in the field of indirect SHM where datasets are impacted by complex bridge-vehicle interactions and impacted by inhomogeneous environments. As such, data-driven kernels such as IDK emerged that do not rely on pre-specified assumptions of the data structure, allowing for greater flexibility in capturing complex relationships and greater generalizability of findings resulting in improved predictive accuracy [5].

The IDK algorithm is an example of a kernel point anomaly detection method that maps to the higher dimensional Hilbert space, $\mathcal{H}$. The $\mathcal{H}$ refers to the higher-order mathematical space where vectors and functions reside. Each point is considered a vector

in this space, allowing for mathematical operations exploiting vector properties to be performed, such as using inner products, orthogonality, and norms of vectors.

Let $\mathbb{H}_\psi(D)$ be the set of all isolating hyperspheres H from $\mathcal{D} \subset D$ such that each point $z \in \mathcal{D}$, $|D| = \psi$, is equiprobable of being selected from the rest of the points in $\mathcal{D}$. Each $\theta[z] \in H$ isolates a point $z \in \mathcal{D}$ where z represents the center of the isolating hypersphere [5]. The indicator function is denoted as $\mathbb{1}(\cdot)$

Definition 1 (*Isolation Density Kernel [5]*)**.** For any two points $x, y \in \mathbb{R}$, the Isolation Kernel of x and y is defined to be the expectation taken over all the probability distribution on all partitionings $H \in \mathbb{H}_\psi(D)$ that both x and y fall into the same isolating hyperphere $\theta[z] \in H$, where $z \in \mathcal{D} \subset D$, $\psi = |D|$:

$$\begin{aligned}\kappa_I(x, y|D) &= \mathbb{E}_{\mathbb{H}_{\psi(D)}}[\mathbb{1}(x, y \in \theta|\theta \in H)] \\ &= \frac{1}{t}\sum_{i=1}^{t} \mathbb{1}(x, y \in \theta|\theta \in H_i) \\ &= \frac{1}{t}\sum_{i=1}^{t}\sum_{\theta \in H_i} \mathbb{1}(x, y \in \theta|\theta \in H_i)\end{aligned} \tag{6}$$

where κ_I is constructed using a finite number of partitionings H_i, $i \in [1, t]$, where each H_i is created using randomly subsampled $\mathcal{D}_i \subset D$; and θ is shorthand for $\theta[z]$.

This process of identifying and leveraging distinguishing features through kernels is linked to the concept of feature maps, which transform the data into a high-dimensional space where these features can be more easily separated and analyzed. The definition of a feature map using the isolation kernel is then:

Definition 2 (*Feature Map of Isolation Kernel [5]*)**.** For point $x \in \mathbb{R}$, the feature map $\mathbf{\Phi}(x) : x \to \{0, 1\}^{t\times\psi}$ of isolation kernel is a vector that represents all partitions in all the partitionings $H_i \in \mathbb{H}_\psi(D)$, $i \in [1, t]$. In each partition H_i, the point x belongs in a maximum one of the ψ isolating hyperspheres.

For a partitioning H_i, $\mathbf{\Phi}_i(x)$ is a ψ-dimensional binary vector representing all hyperspheres $\theta_k \in H_i$, $k \in [1, \psi]$; where x falls into either exactly one or none of the ψ isolating hyperspheres. The k-component of the vector is $\mathbf{\Phi}_{i,k}(x) = \mathbb{1}(x \in \theta_k \in H_i)$ such that each element $\mathbf{\Phi}_{i,k}(x) = 1$ or $\mathbf{\Phi}_{i,l}(x) = 0$, $\forall k \neq l$ for any $k, l \in [1, \psi]$. Given t partitionings, $\mathbf{\Phi}(x)$ is the concatenation of $\mathbf{\Phi}_1(x), \ldots, \mathbf{\Phi}_t(x)$.

This $\mathbf{\Phi}$ is applied to identify these key recurring patterns and trends in the n points across D. For each point $x_j \in D$, $\mathbf{\Phi}(\cdot)$ acts as a filter highlighting each of these key features, with 1 representing the feature's presence and 0 representing its absence for each of the ψ hypersheres. The resultant $\mathbf{\Phi}(\cdot)$ is a vector with a magnitude indicative of the similarity to the greater dataset: a greater magnitude representing greater similarity and a smaller magnitude representing a lower similarity. By comparing the locations of 1s and 0s across the $[\mathbf{\Phi}(D)]_{n\times t\psi} = [\mathbf{\Phi}(x_1), \ldots, \mathbf{\Phi}(x_n)]^T$, the features in common in the points can be identified.

However, these $\mathbf{\Phi}$s are computationally expensive. As such, the kernel trick can provide an analytical solution by re-expressing Eq. (6) as:

$$\kappa(x, y|D) = \frac{1}{t}\langle \mathbf{\Phi}(x|D), \mathbf{\Phi}(y|D)\rangle \tag{7}$$

where $\langle\cdot\rangle$ represents the dot-product.

These kernel-based $\mathbf{\Phi}$s bypass the explicit computation of higher dimensional space vectors by allowing the direct calculation of an inner product in higher dimensional space. This allows for handling infinite-dimension vectors and functions, which is essential in processing complex time series data. KME is an example of this approach, converting a point-to-point kernel to a distributional kernel. However, it has two critical issues caused by using a Gaussian kernel: quadratic time complexity and weak task-specific accuracy [16]. These issues are rectified if the Gaussian kernel is replaced with an isolation kernel. This is what is considered the IDK algorithm [5].

The key benefit of IDK over other existing distributional methods, such as Gaussian distributional kernels (GDK) and One-Class Support Vector Machine (OCSVM), is that it is an exact and finite-dimensional $[\mathbf{\Phi}(\cdot)]_{n\times t\psi}$ with data-dependent distributional characterization, providing local density adaptability with reduced computational time [16,25]. It also has anomaly classification advantages and computational efficiency benefits. The IDK algorithm for time series anomaly detection has the unique benefit of only requiring minimal training and no additional processes. Consequently, it requires a reduced sample size and has a linear runtime with respect to sample size compared to the quadratic or superlinear runtime of comparative processes [5]. Collectively, these create benefits of both computational efficiency and a high capacity for dissimilarity detection [5].

3.2. Parameters selection

The IDK algorithm involves two parameters: t and ψ. The parameter t refers to the number of partitions of the dataset, and the parameter ψ refers to the number of isolating hyperspheres. Each data point within each partition t is classified into only one of the ψ isolating hyperspheres or none. Each isolating hypersphere is then mapped to a single point in $\mathcal{H}$. These values need to be optimized for the particular dataset to maximize the clarity of anomaly detection. Fig. 2 demonstrates the relationships between these parameters and the dataset, using the example of $\psi = 2$ isolating hyperspheres (circles). Each signal is mapped to a point in

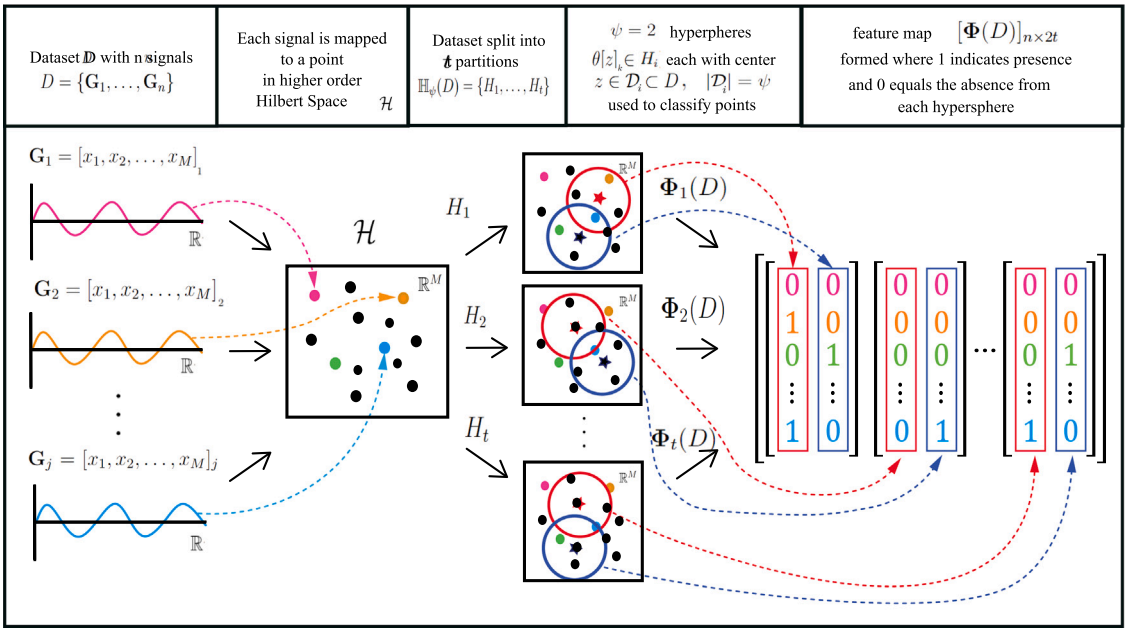


Fig. 2. Diagram illustrating the role of parameters ψ and t in algorithm. Feature map Φ of isolation kernel with t partitions of two isolating hyperspheres ψ (circles), centered about an origin (star), mapped to $\psi = 2$ points. A point, x, is either in one of these two hyperspheres (blue or red) or none. If x is in an overlapping region, it is allocated to the center of which isolating hypersphere that is closer. This classification data is kept in the Φ by concatenating matrices that represent each partition where each column represents each of the hyperspheres.

$\mathcal{H}$. These points are grouped into the $\psi = 2$ isolating hyperspheres or outside of them. If it is inside, it has a corresponding Φ value of one, but if it is outside, it has a Φ value of zero. This process is completed t times over which the average value is used to help calculate the α_j.

These parameters can be selected based on domain knowledge or tuned through empirical best fit assessment. To select the parameters ψ and t that best fit the model, their roles in creating effective isolation and ensuring stable anomaly detection must be considered. Their effect on the algorithm's runtime must also be taken into account.

For the parameter, ψ, consider that each point $z \in D \subset D$ can be isolated from the other points by constructing a hypersphere that only includes z, the radius of which is determined by the distance to z's nearest neighbor in $D\ \{z\}$. This means that the H_i partition consists of ψ hyperspheres with the $(\psi + 1)$th point that represents the region in $\mathbb{R}$ outside of the ψ hyperspheres [16]. For effective separation, there must be at least $\psi = 2$ hyperspheres to form isolated regions in the dataset such that the relative distances between points and the density of the regions in the dataset can be compared. With $\psi > 2$, the number of isolated regions increases, leading to over-isolation and making it more difficult to classify damage accurately [16]. As such, for the case of indirect SHM of bridges, $\psi = 2$ is the most appropriate choice.

To determine the parameter t, it is essential to ensure that the kernel can uniquely represent and distinguish between different probability distributions. This will occur when the kernel mean map of the isolation kernel κ_I is characteristic in $\mathcal{X}$ [16].

Definition 3 (*Characteristic Kernel [16]*)**.** A kernel $\kappa : \mathcal{X} \times \mathcal{X} \to \mathbb{R}$ is called a characteristic kernel if it satisfies the following conditions:

1. The Gram matrix G for a set of vectors $v_1, \ldots, v_n$ with entries $G_{ij} = \langle v_i, v_j \rangle$ is full rank.
2. The feature map $\boldsymbol{\Phi} : \mathcal{P} \to \mathcal{H}$ associated with the kernel is injective, such that:

$$\|\boldsymbol{\Phi}(\mathcal{P}_S) - \boldsymbol{\Phi}(\mathcal{P}_T)\| = 0 \iff \mathcal{P}_S = \mathcal{P}_T$$

for any probability distributions $\mathcal{P}_S, \mathcal{P}_T \in \mathcal{P}$, where $\mathcal{P}$ is the set of admissible probability distributions on $\mathbb{R}$ that are strictly positive on $\mathcal{X}$ and strictly zero over the complement $\bar{\mathcal{X}}$.

A matrix is said to have full rank if it satisfies:

Definition 4 (*Full Rank [26]*)**.** A matrix A of size $m \times n$ is said to be of full rank if its rank is equal to the smaller of the number of its rows or columns. Formally, this can be expressed as:

$$\text{rank}(A) = \min(m, n)$$

This implies that if $m \leq n$, the columns of A are linearly independent and if $m \geq n$, the rows of A are linearly independent.

The injectivity of the $\mathbf{\Phi}$ guarantees that distinct distributions will have distinct representations in the feature space. This characteristic occurs with asymptotic partitioning certainty, defined as follows:

Definition 5 (*Asymptotic Partitioning Certainty [16]*)**.** Let H be a partitioning of a space $\mathcal{X}$ generated by a set of points $D \subseteq \mathcal{X}$. The partitioning H exhibits asymptotic partitioning certainty if, as $t \to \infty$, the probability that H covers the entire space $\mathcal{X}$ approaches 1. Formally, this is expressed as:

$$\lim_{t\to\infty} P(H_t) = 1$$

where H_t denotes the partitioning induced by the points in D at partition t, and $P(H_t)$ represents the probability that the partitioning occurs.

To prove that the Gram matrix is full rank, consider the asymptotic partitioning certainty. Due to this property, for $\psi > 2$ there exists $H_i \in \{H_1, H_2, \dots, H_t\}$ and $\theta_k \in H_i$ with probability 1 such that for any mutually exclusive points $x, y \in \mathcal{X}$ as $t \to \infty$ $x \in \theta_k$ and $y \notin \theta_k$ [16]. This property essentially guarantees that as the number of partitions increases, we can find a partition that separates any two distinct points in the space. This separation is crucial for ensuring the uniqueness of each point's representation in the feature space. In terms of $\mathbf{\Phi}$s, this result implies that as $t \to \infty$ there exists $\boldsymbol{\Phi}_{i,k}(x)$ for any $x \in D$ such that $\boldsymbol{\Phi}_{i,k}(x) = 1$ and $\boldsymbol{\Phi}_{i,k}(y) = 0$ for all $y \in D$ where $y \neq x$. This makes the Gram matrix full rank due to the mutually independent $\boldsymbol{\Phi}(x)$ for all points $x \in D$ and it makes the isolation kernel positive definite with probability 1 in the limit of $t \to \infty$ [16].

To prove the $\mathbf{\Phi}$ is injective, consider the regions $R \subset \mathcal{X}$ and $\bar{R} = \mathcal{X} \backslash R$ such that $\forall x \in R$, $\mathcal{P}_T(x) \neq \mathcal{P}_S$ and $\forall x \in \bar{R}$, $\mathcal{P}_S(x) = \mathcal{P}_T(x)$. By definition, this gives $\int_{\bar{R}} (\mathcal{P}_S(x) - \mathcal{P}_T(x))dx = 0$ and $\int_{\mathcal{X}} \mathcal{P}_S(x)dx = \int_{\mathcal{X}} \mathcal{P}_T(x)dx = 1$. From this,

$$\int_R (\mathcal{P}_S(x) - \mathcal{P}_T(x))dx = 0 \tag{8}$$

can be deduced [16]. Combined with the fact that $\forall x \in R$, $\mathcal{P}_S(x) \neq \mathcal{P}_T(x)$, $\exists R' \subset R$ such that

$$\int_{R'} (\mathcal{P}_S(x) - \mathcal{P}_T(x))dx \neq 0. \tag{9}$$

Consequently, R must contain at least two distinct points in $\mathcal{X}$ [16].

As such, there are two mutually exclusive cases for H_i of an isolation kernel for probability distributions $\mathcal{P}_S, \mathcal{P}_T$: Case 1 where the subset $\theta \in H_i$ does not overlap with $R \subset \mathcal{X}$, and Case 2 where the subset $\theta \in H_i$ partially overlaps with $R \subset \mathcal{X}$ but is not entirely contained within [16].

In Case 1, the probability of data point from either $\mathcal{P}_S$ or $\mathcal{P}_T$ falling into θ is the same as they are both outside of R, making the expected value the same regardless of distribution i.e. $\mathcal{P}_S = \mathcal{P}_T$. For Case 1, the probability is non-zero due to Property 1 with the fact that all $\theta \in H_i$ have non-zero volumes for all partitions up to the $(\psi + 1)$th region for $\psi \geq 2$. This non-zero volume is because the partitioning H_i can be interpreted as $\mathcal{X}$ being packed with randomly sized hyperspheres, known as random-close packing [16]. In these cases, it has been previously established that the higher dimensional packing is less than 100% which means that the area not covered by the hyperspheres will always have a non-zero volume. This region is referred to as the $(\psi + 1)$th region [16].

In Case 2, since there is partial overlap, some of the regions that distinguished between $\mathcal{P}_S$ and $\mathcal{P}_T$ are captured by R, resulting in different expected values i.e. $\mathcal{P}_S \neq \mathcal{P}_T$ [16]. Case 2 also has non-zero probability as R contains at least two distinct points in $\mathcal{X}$ for $\psi \geq 2$. As such, the probability p of H_i satisfying Case 1 but not Case 2 must be $0 < p < 1$. Due to the independent sampling over $i \in [1, t]$, for all t this probability becomes p^t. As $t \to \infty$, $p^t \to 0$ as $0 < p < 1$. Hence, in the limit, at least one partition will overlap, allowing us to distinguish between $\mathcal{P}_S$ and $\mathcal{P}_T$ [16]. This combined with the isolation kernel's positive definite nature makes it a characteristic kernel [16].

For anomaly detection, the kernel mean map must be characteristic to properly separate anomalies from normal points. It has been shown that $t = 100$ is sufficient for anomaly detection. However, there is increased stability with $t = 1000$ [16]. In other words, since t represents the number of repetitions that the results are averaged over, increasing values of t minimize the variance of results. Hence, larger values of t improve the stability of results but increase computational time [16].

3.3. Time-series application of IDK

In this study, our signals are measured acceleration time responses x_m of length M from n runs and the IDK approach is utilized to enhance the separation between different groups and to extract the defining features from the signals [5]. The IDK approach for anomaly detection in time series consists of two stages: first, mapping each time series to a single point in higher-order space and, second, computation of α_j between mapped points. Algorithm 1 provides an overview of this process when applied to time series data, and it is explained next. (i) The signal is first mapped to higher-order space as a set of vectors, $\boldsymbol{\Pi}$, that define the distinctive features of the original signals using kernel (line 6). (ii) Then the mean map of this kernel, $\mathcal{P}_{\boldsymbol{\Pi}}$ is calculated (line 9). (iii) Finally α_j is calculated using a dot product $(\cdot)$ of the unique vectors (line 11). Since the scalar dot product $(\cdot)$ measures the projection of one vector upon another, the greater the product, the greater the similarity between the vectors. Hence, the dot product $(\cdot)$ in higher-order space measures the similarity between the original time series data. This IDK approach allows the detection of a point anomaly within the dataset.

Algorithm 1: IDK for anomaly detection.

Function IDK(D, t, ψ)**:**
  Dataset $D = \{\mathbf{G}_j\}_{j=1}^{n}$ for $j = 1, \dots n$ containing a set of n time-series $\mathbf{G}_j$ where $D \subset \mathcal{X} \subseteq \mathbb{R}^M$
  Hyper-parameter t for the number of partitions H_i for $i \in [1, t]$;
  Hyper-parameter ψ for the number of isolating hyperspheres;
  for $j \in [1, \dots, n]$ **do**
    $\boldsymbol{\Phi}(\mathbf{G}_j) = \frac{1}{t} \sum_{k=1}^{\psi} \mathbb{1}(\mathbf{G}_j \in \theta_k | \theta_k \in H_i)$ Calculate isolation density kernel
  end
  $\Pi = \{\boldsymbol{\Phi}(\mathbf{G}_j) | j = 1, 2, \dots, n\}$
  $\boldsymbol{\Phi}(\mathcal{P}_{\Pi}) = \frac{1}{|D|} \sum_{\mathbf{G}_j \in D} \boldsymbol{\Phi}(\mathbf{G}_j)$ Calculate kernel mean map
  for $j = [1, \dots, n]$ **do**
    $\alpha_j = \frac{1}{t} \langle \boldsymbol{\Phi}(\mathbf{G}_j), \boldsymbol{\Phi}(\mathcal{P}_{\Pi}) \rangle$ Calculate similarity score
  end
  return $\{\alpha_j\}_{j=1}^{n}$

To apply the IDK method to a time series, assume each time series $\mathbf{G}_j$ is an independent and identically distributed point in $\mathbb{R}^M$, generated from an unknown probability density function (pdf) $\mathcal{P}_{\mathbf{G}_j}$. The aim is to test the similarities between time series $\mathbf{G}_j$ and $\mathbf{G}_l$, $j \neq l$, to determine if they are derived from the same distribution $\mathcal{P}_{\mathbf{G}_j} = \mathcal{P}_{\mathbf{G}_l}$ or dissimilar distributions $\mathcal{P}_{\mathbf{G}_j} \neq \mathcal{P}_{\mathbf{G}_l}$ for each level of the IDK algorithm. For a dataset, the algorithm produced a ranked list of signals $\{\mathbf{G}_j | j = 1, \dots, n\}$ in ascending order, based upon α_j [5].

For a dataset of time series $D = \{\mathbf{G}_j\}_{j=1}^{n}$, the algorithm begins by mapping each of these signals $\mathbf{G}_j \in D$ to a point in $\mathcal{H}$ using Eq. (10) for each reiteration $i \in [1, t]$ and isolating hyperspheres $k \in [1, \psi]$. This gives a $\boldsymbol{\Phi}_{j,i}(\mathbf{G}_j | D)$.

Definition 6 (*IDK [5]*)**.** The IDK of two distributions $\mathcal{P}_{\mathbf{G}_j}$ and $\mathcal{P}_{\mathbf{G}_l}$ can be defined as the dot product between the mean $\boldsymbol{\Phi}$s between the two distributions:

$$\mathcal{K}_I(\mathcal{P}_{\mathbf{G}_j}, \mathcal{P}_{\mathbf{G}_l} | D) = \frac{1}{t} \langle \boldsymbol{\Phi}(\mathcal{P}_{\mathbf{G}_j} | D), \boldsymbol{\Phi}(\mathcal{P}_{\mathbf{G}_l} | D) \rangle \tag{10}$$

where $\mathcal{K}_I$ is the IDK of the mean kernel map with the mean kernel map for each signal $\mathbf{G}_j$ calculated as:

$$\boldsymbol{\Phi}(\mathcal{P}_{\mathbf{G}_j} | D) = \frac{1}{|\mathbf{G}_j|} \Sigma_{x \in \mathbf{G}_j} \boldsymbol{\Phi}_k(x | D) \tag{11}$$

The similarity $\alpha_j \in [0, 1]$ between each point $\mathbf{G}_j$ is computed as:

$$\alpha_j = \frac{1}{t} \langle \boldsymbol{\Phi}(\mathbf{G}_j), \boldsymbol{\Phi}(\mathcal{P}_D) \rangle \tag{12}$$

Each time series is sorted in ascending order based on α_j [5]. Using this approach, two points measured by the isolation kernel in a sparse region are more similar than those derived from a dense dataset region due to its data dependent nature.

3.4. Motivating example

To motivate this analysis, consider a toy example of a healthy SHM dataset composed of simple sine waves and anomalous test waves (Fig. 3). These signals represent acceleration-time signals of a vehicle passing over a bridge. Signals (a)–(d) represent runs over a healthy bridge with varying amplitude, at a consistent frequency. As a result, they should be classified as healthy signals. Meanwhile, the signal of a noisy sine wave (e) follows the same oscillating pattern but without the smoothness of the curve. Consequently, it should be classified as having a mild anomaly. However, the signal of an anomalous sine wave (f) is abnormal compared to the reference dataset. It does not follow the same oscillating pattern. Indeed, this lower frequency of vibration represents a change in the frequency of the system. This can be related to a damaged sample of a bridge recorded using drive-by inspection, where the structural damage changes the natural frequencies of the structure. This example set will be used to illustrate the higher-order mathematics behind the algorithm.

The first step of the algorithm is to apply a isolation kernel to the entire dataset to provide an analytical solution for $[\boldsymbol{\Phi}(D)]_{n \times t\psi}$. This is used to identify the defining features of the set associated with the healthy signals (e.g. oscillation frequency) under the assumption that the healthy signals form the majority class. Suppose a isolation kernel is applied to each of the signals from Fig. 3 with signals (a)–(c) representing the benchmark dataset. In that case, the isolation kernel will identify the simple sine wave (Fig. 3(d)) as having these same characteristic patterns as the reference dataset. However, when the noisy sine wave (Fig. 3(e)) is added to the simple sine wave, the isolation kernel will identify this signal as following the same trends, such as the periodic oscillation. As such, the isolation kernel will recognize that it is related to but does not belong to the group of signals in the rest of the dataset

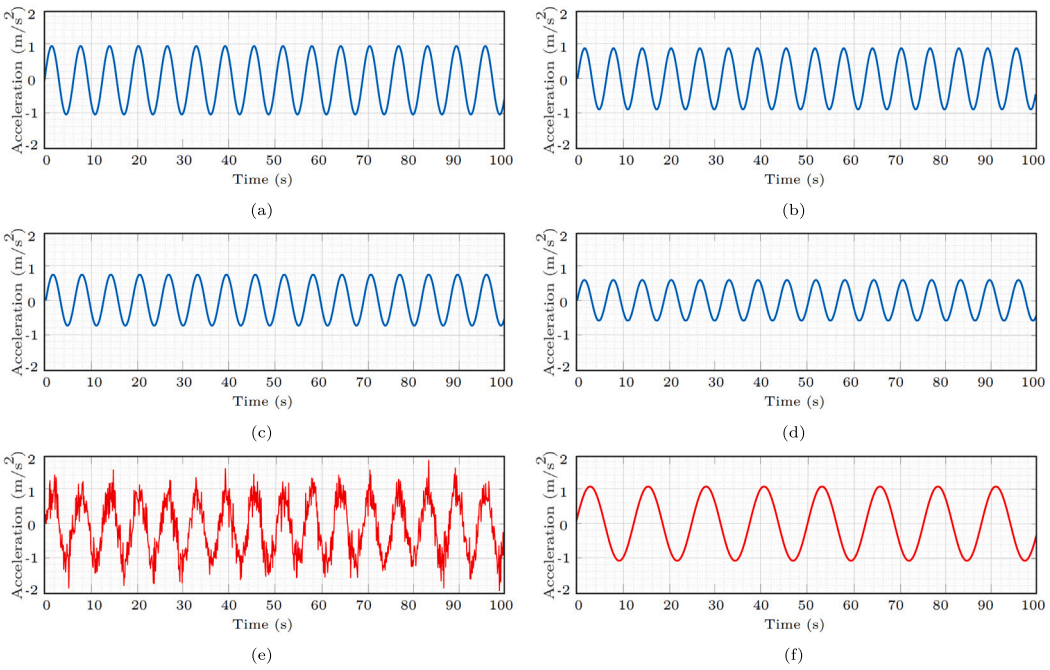


Fig. 3. Toy dataset with (a)–(d) depicting cleaned sine waves with frequency 2π of various amplitudes, (e) showing a noisy sine wave with randomly modulated amplitude, and (f) depicting an anomalous sine wave with lower frequency.

and consequently identify it as having a mild anomaly. For the case of the anomalous sine wave (Fig. 3(f)), the isolation kernel will recognize that it does not follow any of the same trends as the rest of the dataset in terms of frequency and will identify the signal as highly anomalous.

The resultant $[\Phi(D)]_{n\times t\psi}$ of this isolation kernel application numerically outlines the relationship between signals. Eq. (13) shows the $[\Phi(D)]_{6\times 2000}$ matrix of this toy example data. Each row represents signal (a-f), respectively, and the number of columns is equal to $\psi \times t$ for the case $\psi = 2$ and $t = 1000$. The justification of this choice in parameters is included later in Sections 3.2 and 5.5. The … represents columns 15–2000 that have been excluded for simplicity. For each column, a value of 1 means that the mapped point of the signal lies within the corresponding hypersphere $[1, \ldots, \psi]$ of ψ isolating hyperspheres, while a value of zero means that it lies outside of the corresponding isolating hypersphere and therefore does not have that feature. Rows 1–4 have similar values and represent signals (a)–(d), while row 5 which represents signal (e) has fewer, and row 6 which represents signal (f) has the least amount of similarity in the values as they are progressively more dissimilar from signals (a-d) and therefore have their own unique sets of features (locations of 1s). Hence, this shows that signals (a)–(d) have the most common features, while signal (e) has some in common but some additional features, and signal (f) has the most distinct features.

$$[\Phi(D)]_{6\times 2000} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & \ldots \\ 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & \ldots \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & \ldots \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 & \ldots \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 & \ldots \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & \ldots \end{bmatrix} \tag{13}$$

The final stage of the IDK algorithm is calculating the α_j for each of the signals. In the case with parameters $\psi = 2$ and $t = 1000$ (defined in Section 3.2) for the toy example, the similarity vector becomes for $j = 1 : 6$ $\alpha_j = \{0.34, 0.36, 0.35, 0.32, 0.20, 0.03\}$. This shows that the healthy signals have similar and the highest scores ($\alpha_1 - \alpha_4$). However, for noisy sine signal (3e), this reduces to $\alpha_5 = 0.20$; for the anomalous sine wave (3f), it reduces further to $\alpha_6 = 0.03$.

Overall, this toy example demonstrates the anomaly detection capacity of the point anomaly detector IDK. In the following sections, this algorithm will be employed to demonstrate its effectiveness as a potential solution to the issue of indirect SHM of bridges.

4. Framework overview

In this paper, a numerical and experimental case study is undertaken. The numerical dataset is obtained through the VBI framework previously outlined in Section 2, and the experimental dataset is obtained through a scaled VBI model, which is described

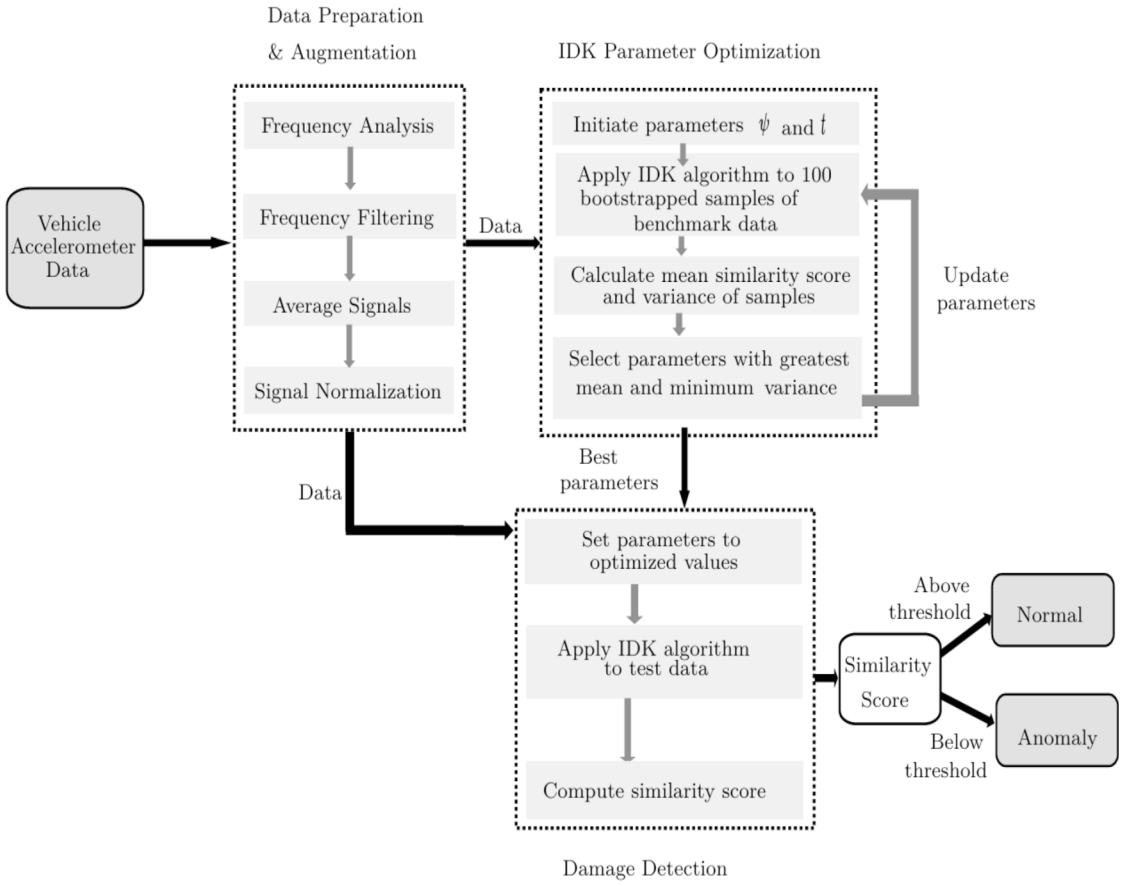


Fig. 4. Summary of the proposed framework.

later in detail in Sections 6.1 and 6.2. Fig. 4 summarizes the methodology proposed in this paper as (i) The measured acceleration signals from vehicles passing over a bridge first undergo a preprocessing step involving frequency analysis, filtering, averaging, and normalization (will be described in Section 5.4 and in greater detail in [4]). This involves obtaining the Power Spectral Density (PSD) and filtering the frequency to the range of 2.5–5 Hz. Thirty of these signals are sampled and averaged before being normalized to the range [0,1]. (ii) The dataset is then used to optimize parameters to find the values that give the greatest median α_j scores with the lowest variance and runtime (Section 5.5) under the assumption that the healthy signals form a majority class. This involves initializing the parameters at their lowest values ($\psi = 2$ and $t = 1$) and applying them to 100 bootstrapped samples to obtain the mean α_j and variance of the samples. The parameters are then subsequently increased until the mean is maximized and the variance is minimized whilst considering their effect on runtime (will be described in Section 5.5). (iii) These optimized parameters are then input into the algorithm, and α_j scores are computed in higher-dimensional space (Section 5.6 for the numerical case study and Section 6.3 for the experimental case study). Under this approach, higher α_j scores correspond to lower levels of damage, while lower α_j scores correspond to higher levels of damage.

5. Numerical case study

In this section, we obtain a numerical data set based on the VBI framework outlined in Section 2 to model the dynamic response of a moving vehicle across multiple states of a simply supported bridge. This dataset is utilized to assess the feasibility of the proposed semi-supervised distributional framework and compare its performance to a competing state-of-the-art framework published in [4].

5.1. Bridge model

Using a simply-supported beam with parameters based on studies by McGetrick et al. [7] (Table 1), we model multiple bridge states. We use the Euler–Bernoulli beam theory to obtain the finite element model of the bridge, with the first three bending modes.

The simulated dataset encompasses four distinct bridge conditions: a healthy bridge and three varying levels of damage severity (10%, 20%, and 30% damage) at three different positions along the bridge (quarter, half, and three-quarter span). These damage

Table 1
Summary of the bridge's and vehicle's physical properties.

	Property	Value
Bridge	Span length (L)	25 m
	Young's Modulus (E)	3.5×10^{10} N/m^2
	Mass per unit length (μ)	18,358 kg/m
	Cross section area (A)	16.68 m^2
	Second moment of inertia (I_0)	1.39 m^4
	Damping ratio (ξ)	3%
Vehicle	Vehicle mass (m_v)	16,200 kg
	Vehicle moment of inertia (I_v)	93,234 kg m^2
	Axle stiffness (k_v)	4×10^5 N/m
	Axle damping (c_v)	1×10^4 N s/m
	Axle distance (d)	4.75 m
	Bouncing frequency	1.10 Hz
	Pitching frequency	1.11 Hz

Table 2
Summary of the bending modes of each bridge state[a].

	Bridge state	Mode 1	Mode 2	Mode 3
Quarter-span	Healthy	4.09(−)	16.36(−)	36.8(−)
	10% Damage	4.05(0.98%)	16.06(1.83%)	36.48(0.87%)
	20% Damage	4.00(2.20%)	15.71(3.97%)	36.11(1.87%)
	30% Damage	3.94(3.67%)	15.33(6.3%)	35.74(2.88%)
Mid-span	10% Damage	4.01(1.9%)	16.35(0.06%)	36.16(1.74%)
	20% Damage	3.92(4.16%)	16.34(0.12%)	36.44(3.70%)
	30% Damage	3.81(6.85%)	16.34(0.12%)	34.66(5.82%)

[a] The frequency results are in Hz (% change relative to healthy state).

states correspond to the percentage of crack depth (h_c) relative to the beam height h as elaborated in [21]. The front and rear axles are taken to be identical in each of these cases.

To quantize the characteristics of each bridge state, eigenvalue analysis is employed to extract the first three bending modes where the first three bending modes of the healthy bridge are 4.09 Hz, 16.36 Hz, and 36.8 Hz. A summary of these results is presented in Table 2. Notably, at the mid-span location, the bridge with the lowest level of damage at 10% exhibits a roughly 2% change in the fundamental frequency compared to the healthy state. In contrast, the moderately 20% damaged bridge shows an approximately 4% change, and the severely 30% damaged bridge exhibits a shift of around 7% [4]. However, the percentage change at the quarter-span location is roughly half that at mid-span, making it more challenging to detect damage at this location.

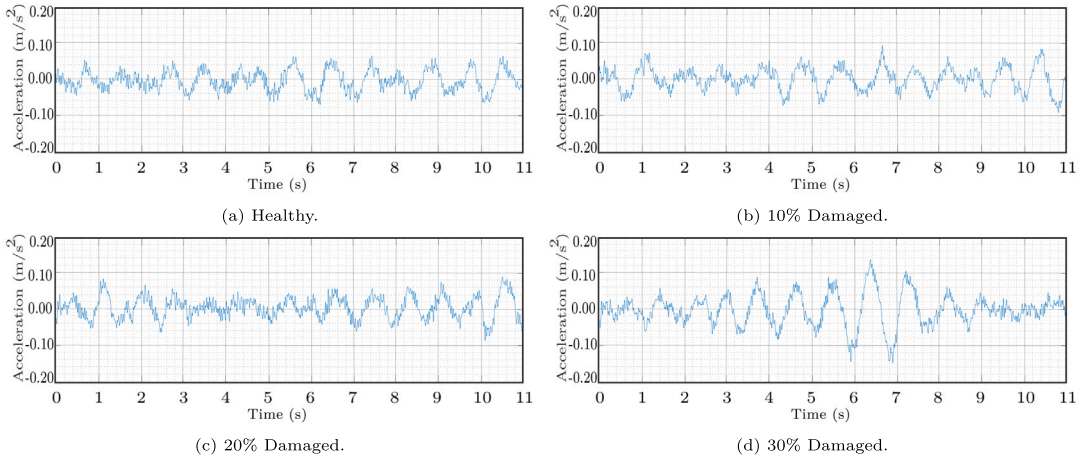
5.2. Vehicle model

For this investigation, the vehicle model utilized is a half-car model with properties detailed in Table 1. These properties are based on relevant reference studies [4] and result in a bouncing mode with a frequency of $f_z = 1.11$ Hz. Additionally, the speed on each vehicle pass is defined by a normal distribution with a mean of 2 m/s and a standard deviation of 0.2 m/s to better replicate real-world conditions as consistent speeds are not practically feasible. Due to these variations in speed and the fixed span of the bridge, the length of each signal generated varied.

5.3. Numerical dataset

The numerical dataset comprises ten distinct bridge conditions, with three different levels of damage at three locations (quarter-span, mid-span, and three-quarter-span). The conventional Newmark-Beta time integration approach is adopted to solve the dynamic equations with a time step of 0.001 s [4]. Acceleration responses from both the front and rear axles of the vehicle are recorded for the duration they are on the bridge. Examples of the raw acceleration signals of the vehicle's front and rear axle for each bridge condition are displayed in Figs. 5 and 6. The underlying assumptions are as previously defined in [2].

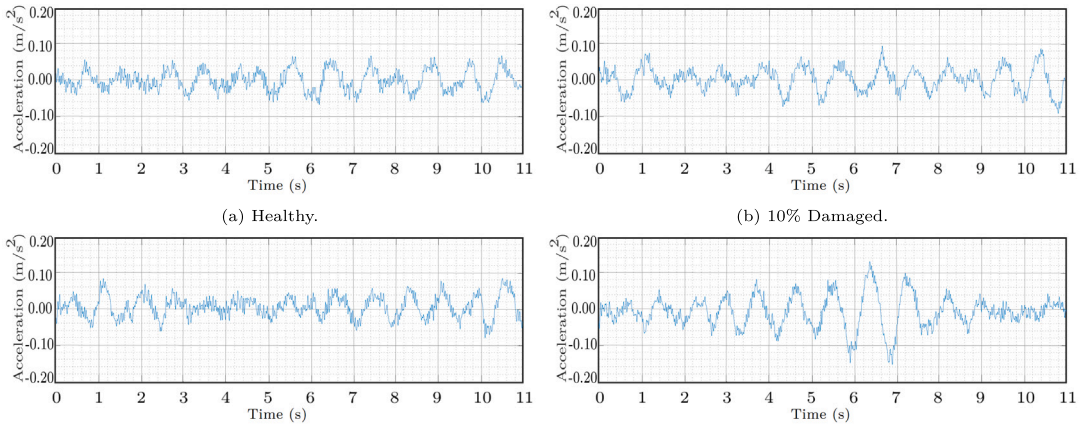
For the healthy state, 900 response signals are generated. Conversely, for each of the damaged states, 100 time series are simulated. These quantities were selected such that the dataset matched that of [4] for the later comparative section. However, we later show that smaller quantities are sufficient for accurate damage severity classification (Section 5.7). To account for the inherent variability in road surface conditions over time, each of these signals has 5% random white noise, and a different stochastic type A road profile is used at every pass of the vehicle as per Section 2, thus simulating the natural variations in road conditions as per [4]. To this aim, it is assumed that the dataset is collected over an extended duration of time. This adds additional variability, increasing the challenge of addressing the problem using this dataset.



(a) Healthy. (b) 10% Damaged.

(c) 20% Damaged. (d) 30% Damaged.

Fig. 5. An example of raw acceleration signals for the front axle for each severity of damage for the case of mid-span damage.



(a) Healthy. (b) 10% Damaged.

(c) 20% Damaged. (d) 30% Damaged.

Fig. 6. An example of raw acceleration signals for the rear axle for each severity of damage for the case of mid-span damage.

5.4. Dataset preprocessing

Fig. 7 shows an overview of the four preprocessing steps for both the numerical and experimental datasets as proposed in [4]. We begin the preprocessing by (i) first transforming the signal of the front and rear acceleration to the frequency domain using the PSD. In addition to the vehicle dynamics, the transformed signal contains characteristics of the bridge dynamics and the stochastic road profile, (ii) this is then followed by subtracting the recorded signals from the two axles in the frequency domain to find the residual spectrum to reduce the effects of the road roughness [27] (iii) then, the residual spectrum is filtered in the desired frequency range of 2.5–5 Hz which encompasses bridge-related information around the first bending mode which ensures the focus remains on the structural properties of the bridge whilst excluding unnecessary vehicle-induced effects [4]. This definition of the cutoff frequencies is made based on preliminary analysis of the frequency spectrum of the recorded signals, from which an estimate of the first natural frequency of the structure can be made.. Later, 30-sample groups are randomly selected and averaged to minimize external disturbances from the spectrum and (iv) finally, the averaged frequency spectrum is normalized to a range of [0–1] and used as the dataset for the IDK anomaly detection model.

The Welch method of PSD estimation is used to obtain the frequency domain representation of the signals and the result of this process on the numerical signals is depicted in Fig. 8. For this PSD, a Hamming window with a length half of the signal length is used, with an overlap of half the window length. The number of points used in the Fast Fourier Transform (FFT) is set to the next power of two greater than or equal to the window length to optimize computational efficiency. Peaks in these figures represent the modal frequencies of the individual signals, i.e., the first bending mode of the bridge is close to 4 Hz. The plot demonstrates a leftward shift of modal frequency with increasing damage [28]. As shown in Figure 8, a frequency range of 2.5-5 Hz is applied to filter the data using a bandpass filter.

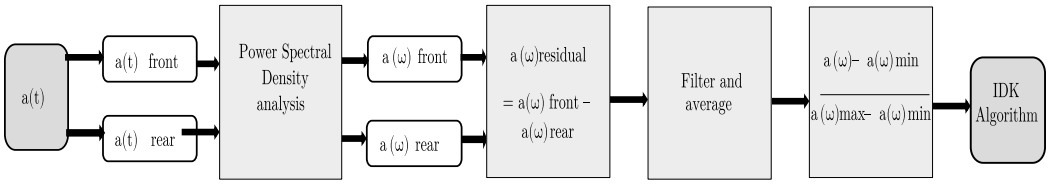
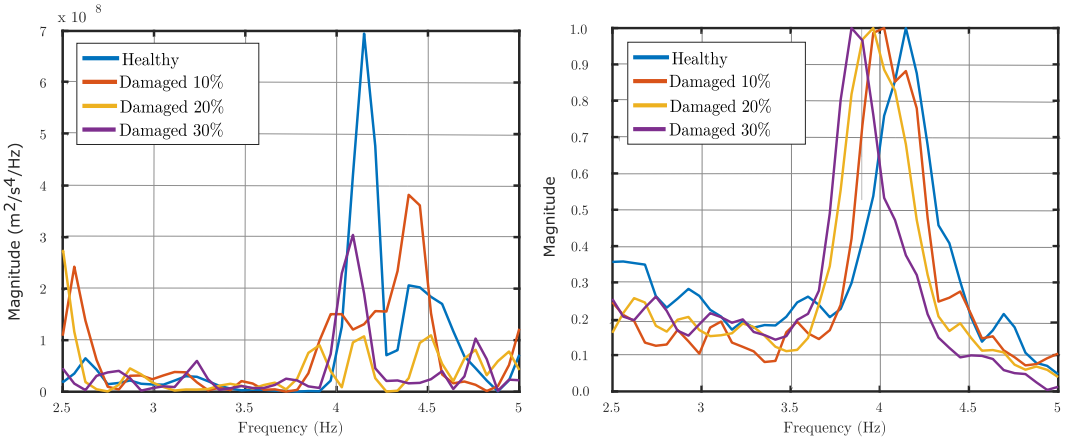


Fig. 7. Overview of the preprocessing steps for initial acceleration signal in the time domain $a(t)$ that is transformed into the frequency domain $a(\omega)$ using PSD. The averaged filtered residual spectrum is then found and used in the IDK algorithm.



(a) Residual frequency spectrum. (b) Pre-processed spectrum.

Fig. 8. Comparison of residual frequency spectrums (PSD) before and after preprocessing numerical dataset for the case of mid-span damage.

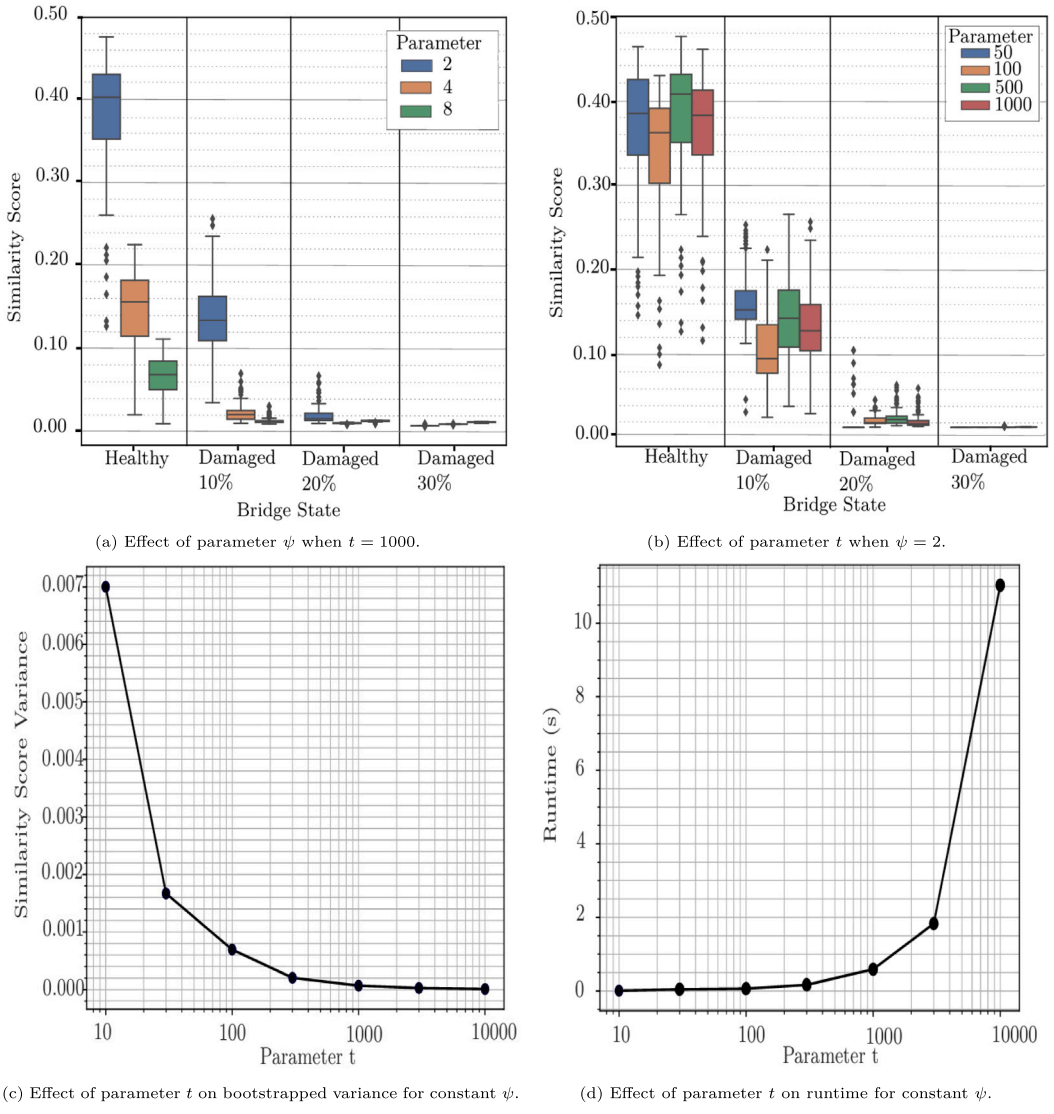
This preprocessing procedure is used to generate 5000 different averaged spectra for the healthy bridge state and 100 for each damage state. This 5000 comes from 30 randomly samples signals across those 900 ones and purely to train a deep learning model used in [4] for the subsequent comparison section but later analysis reveals that much smaller sample sizes are sufficient (Section 5.7).

5.5. IDK model parameter selection

To determine the optimal parameters ψ and t for the model, we need to evaluate their contributions to achieving effective isolation and reliable anomaly detection. Additionally, how these parameters affect the algorithm's runtime must be considered. All results are presented as boxplots in which the central line represents the median, the edges of the box represent the first and third quartiles, and the whiskers represent the 2.5 and 97.5 percentiles. As shown in Fig. 9(a) over the set $\psi \in \{2^q | q = 1, 2, 3\}$, $\psi = 2$ has the greatest median value of α_j for the healthy bridge state, with a substantial decline with increasing values which prevents more severely damaged bridge states clustering about zero due to overfitting. Alternatively, Fig. 9(b) shows the effect of parameter t over the set $t \in \{50, 100, 500, 1000\}$, depicting a decreasing variance over the entire set of α_js as t increases but the ability to distinguish between bridge states regardless of the value of t. Fig. 9(c) then demonstrates the relationship between t and result stability using the variance of 100 bootstrapped means of α_j over the complete dataset. It depicts how the variances decrease as t increases, with minimal change past $t = 1000$. Fig. 9(d) then shows the effect of increasing parameter t on runtime. This plot demonstrates that runtime exponentially increases as t increases, increasing substantially past $t = 1000$. Thus, the parameters $\psi = 2$ and $t = 1000$ is selected for this study as they balance computational efficiency with result stability whilst effectively isolating the bridge states.

5.6. Damage severity assessment

To demonstrate the effectiveness of this preprocessing methodology, the IDK algorithm is run on the raw acceleration signals for both the front and rear axles, the residual acceleration signal obtained via the subtraction of the rear from the front axle, and the preprocessed signals. The α_j of each signal to its most similar neighbors is computed. Fig. 10 shows the effect of preprocessing on the identification of the bridge state using the IDK approach. Fig. 10(a) depicts the inability to distinguish classes using the raw front or rear axle acceleration signals. Fig. 10(b) also depicts the inability to distinguish between states using the residual spectrum, however, this has fewer outliers than the raw spectrum results. Finally, Fig. 10(c) demonstrates a clear distinction between classes using the preprocessed data to obtain the frequency domain representation of the signals, demonstrating a decrease in the α_j corresponding to more severely damaged signals. There is no overlap of the middle two quartiles between any damage severity levels, demonstrating



(a) Effect of parameter ψ when $t = 1000$.

(b) Effect of parameter t when $\psi = 2$.

(c) Effect of parameter t on bootstrapped variance for constant ψ.

(d) Effect of parameter t on runtime for constant ψ.

Fig. 9. Effect of parameter tuning on result stability, runtime, and class separation for damage assessment procedure for the case of mid-span damage.

effective separation of damage levels by the algorithm.

To further analyze the results additional statistical testing is performed. The statistical testing confirms these differences between severity levels, including the Kruskall-Wallis h test and Dunn's pairwise testing using p-values. These p-values provide a quantitative measure to look at the error ranges of measurements to verify differences between groups. In most academic communities, a p-value < 0.05 is considered to be statistically significant. Kruskall-Wallis h test is a non-parametric rank-based test equivalent to a one-way Analysis of Variance (ANOVA) to determine the presence of significant differences in a continuous variable between two or more independent groups of independent observations. It is the appropriate choice given the non-normality of the dataset [29]. The test demonstrates a significant difference between damage severity groups (p-values < 0.001). Given these statistically significant results, pairwise comparisons using Dunn's test are undertaken. Dunn's test is a method of non-parametric pairwise comparison of the mean rank of each independent group [30]. To decrease type I error, Bonferroni correction is applied to the analysis to adjust the significance level to the number of comparisons made [30]. Upon pairwise comparison using Dunn's test, all pairwise comparisons show significant differences (p-values< 0.001). A logistic regression model is applied to the α_j scores to assess accuracy, and the one-vs-rest approach is used to calculate the area under the receiver operating characteristic (ROC-AUC). These values demonstrate substantial accuracy for all classes, with 99% accuracy for the healthy state, 94% accuracy for the 10% damaged state, 98% for the 20% damaged state, and 100% accuracy for the 30% damaged state when applied to the combined healthy and damaged dataset.

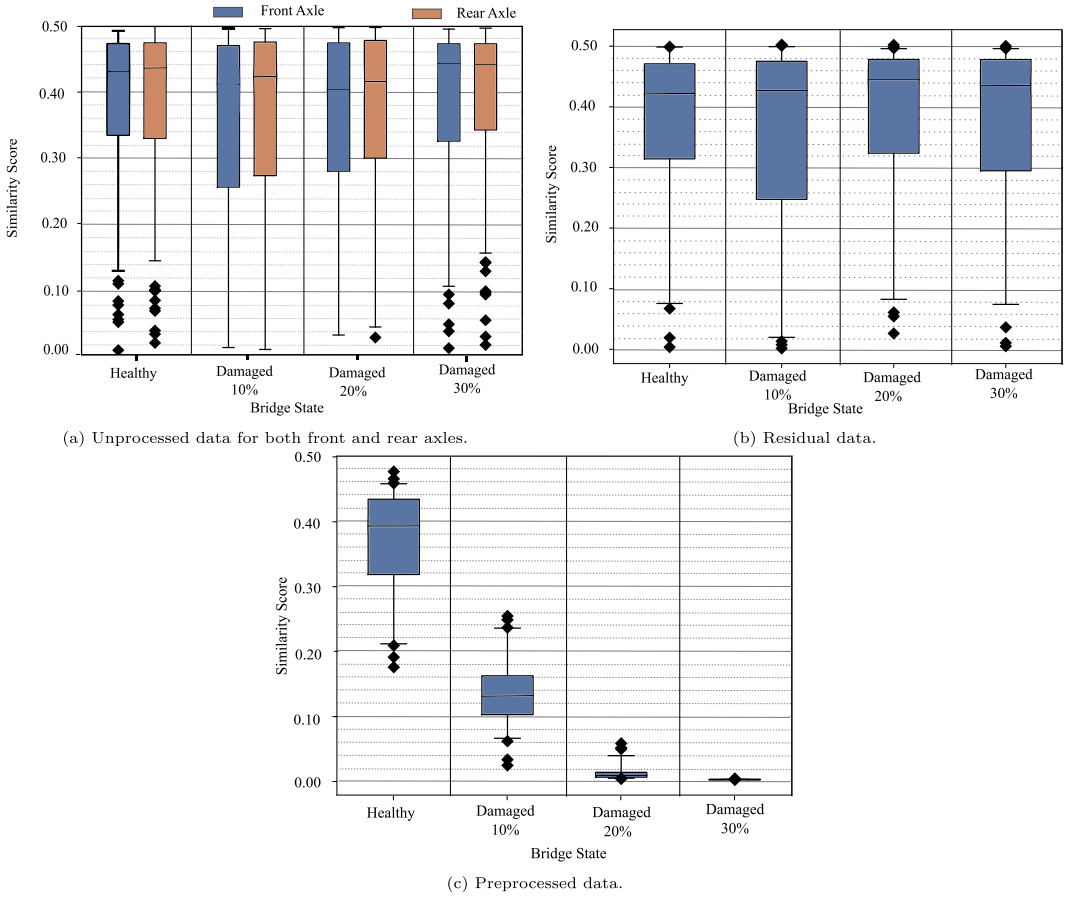


Fig. 10. Comparison of the damage classification assessment performance for the effect of preprocessing on data classification accuracy using IDK algorithm for the case of mid-span damage.

Consequently, the proposed IDK algorithm with preprocessing discussed earlier effectively detects bridge damage and distinguishes different severity of damage from one another. These findings are essential in indirect SHM as this novel algorithm can accurately classify damage severity by simple drive-by inspection.

5.7. Effect of sample size

To assess the effect of the sample size, the number of healthy responses is varied. For computational efficiency, the value of t is set to 1000, and ψ is set to 2 (see Section 5.5). This was implemented under the assumption that the data can be considered healthy if obtained as an initial reference measurement of a bridge and any data obtained at a later date is considered unknown. Fig. 11 depicts the findings of these experiments for healthy sample sizes 5, 10, 50, 100, 500, 1000, and 5000, demonstrating significant separation between the healthy and damaged bridges even at very small sample sizes (i.e., 5 signals) reflecting the approach's high sensitivity and specificity but greater delineation between groups increasing as sample size increases. This illustrates the algorithm's efficiency, with the IDK approach requiring minimal reference data of 5 signals for accurate binary damage severity classification but it requires a greater number of signals for damage severity assessment. This ability to distinguish an anomaly from minimal benchmark data reiterates the novelty of the IDK approach and its practicality in real-world situations where obtaining large datasets, such as in drive-by SHM, is difficult.

To further quantize these findings for classification accuracy, a threshold to define an anomaly was set using the first and third quantiles (Q1 and Q3 respectively) as $Q1 - 1.5 \times (Q3 - Q1)$. Signals with similarity scores above this threshold were classified as healthy, while those below it were classified as damaged. This threshold was chosen because it does not require labeled data, relying only on the assumption that healthy signals are the majority class which makes it practical in real-life applications of the algorithm. Table 3 summarizes these results, showing high binary classification accuracy across all sample sizes and damage states. However, as the healthy sample size increases, the accuracy marginally declines due to the algorithm's tendency to identify anomalies within the healthy dataset, leading to overfitting. Furthermore, accuracy marginally decreases with increased damage severity, as the inclusion of more severe damage signals introduces greater variability in healthy similarity scores. This issue arises from the lack of distinct

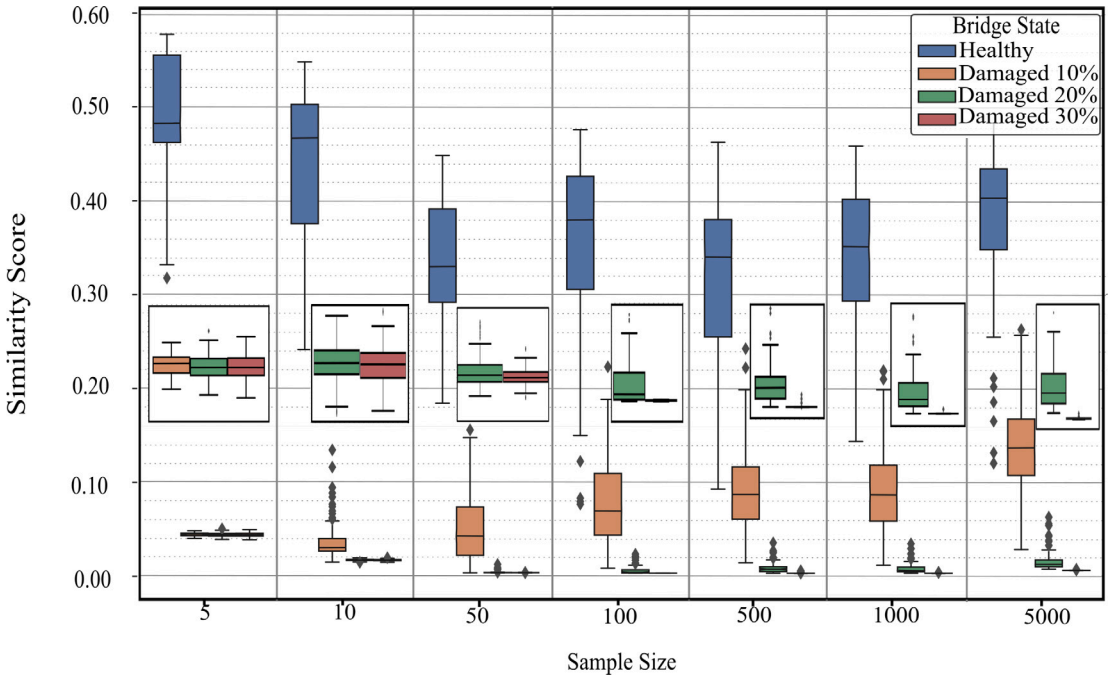


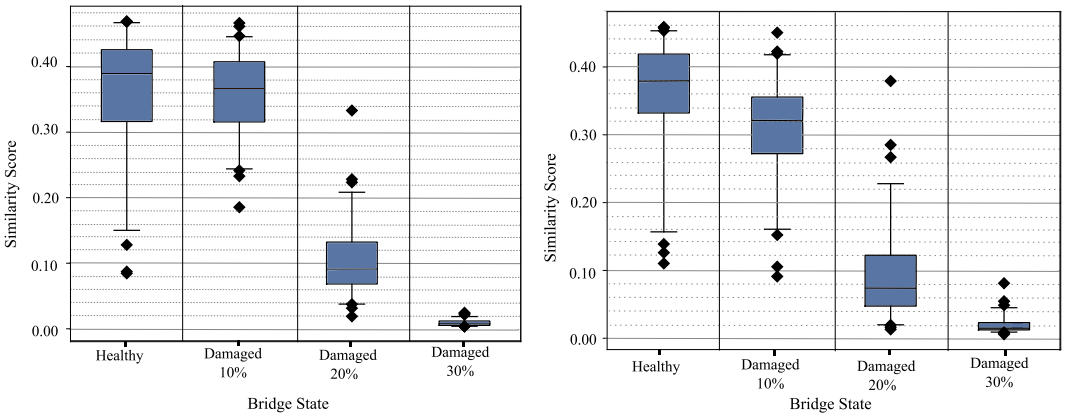
Fig. 11. Effect of healthy sample size on the damage identification and severity assessment tasks with a box showing zoomed-in results. Additional zoomed-in figures are included in A.1 for additional clarity.

Table 3
Effect of sample size of classification accuracy metrics for the numerical mid-span damage case study[a].

Bridge state	Sample size	Metric			
		True positive	True negative	False positive	False negative
10% Damage	5	100	100	0	0
	10	99	99	1	1
	50	97	100	0	3
	100	86	98	2	14
	500	83	96	4	17
	1000	81	97	3	19
20% Damage	5	100	100	0	0
	10	100	100	0	0
	50	100	100	0	0
	100	100	92	8	0
	500	100	97	3	0
	1000	100	98	2	0
30% Damage	5	100	100	0	0
	10	100	100	0	0
	50	100	100	0	0
	100	100	100	0	0
	500	100	93	7	0
	1000	100	98	2	0

[a] The true positive rate refers to the rate of correctly identified damage signals whilst the true negative rate refers to the correctly identified healthy signals.

training and testing phases in the algorithm where similarity scores are computed relative to each other rather than definitive values. Despite this, the results retain a high level of accuracy. While this method demonstrates promising results, it could be further refined by adopting a more sophisticated definition of the anomaly threshold tailored to each damage state to improve the true negative rate. However, this would require prior knowledge of the different damage states, which is unlikely to be available for SHM of bridges.



(a) Quarter span. (b) Three-quarter span.

Fig. 12. Damage detection for the numerical dataset.

Table 4
ROC-AUC score for severity classification in the numerical study.

Damage location	Bridge condition			
	Healthy	10% Damage	20% Damage	30% Damage
Mid-span	0.99	0.94	0.98	1.00
Quarter-span	0.83	0.49	0.96	1.00
Three-quarter-span	0.90	0.69	0.96	1.00

5.8. Effect of damage location

The ability of the proposed methodology to distinguish damage at varying locations is also examined. To this aim, damage signals at quarter and three-quarter spans are considered in addition to the previously used mid-span data. The preprocessing is applied to all signals. Fig. 12 shows the damage detection capacity of the method in Fig. 12(a) for quarter-span and Fig. 12(b) for three-quarter span damage, demonstrating that the algorithm's ability to distinguish damage severity is dependent on damage location with greater identification capacity when the damage is located at mid-span. The damage detection algorithm retains the ability to detect 20% and 30% damage states from the healthy data, with minimal overlap of middle quartiles for quarter-span and three-quarter-span data sets.

Statistical testing is undertaken as per the mid-span dataset to confirm these differences between severity levels. The Kruskall-Wallis test demonstrated significant differences between groups ($p < 0.001$), with all Dunn's pairwise comparisons showing significant differences ($p < 0.001$) except the healthy and 10% damage comparison for both quarter-span ($p = 1.000$) and three-quarter-span datasets ($p = 0.100$). Table 4 provides the ROC-AUC for each damage severity classification, demonstrating high-level accuracy for detecting the 20% and 30% damage classes, remaining above 95%. The healthy class accuracy reduced to a moderate 83% for the quarter span and 90% for the three-quarter span α_j scores, and the accuracy of the 10% damaged class reduced substantially to 49% and 69%, respectively for quarter- and three-quarter-span. These findings encapsulate the reduced effectiveness of the IDK approach for non-centrally located minor damage yet retained the ability to identify more severely damaged states.

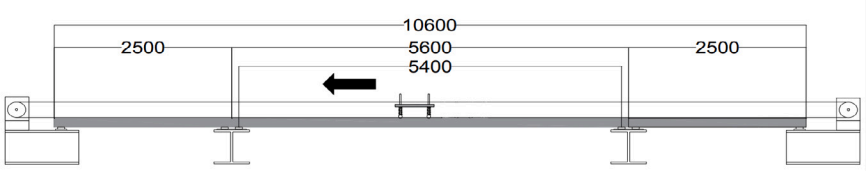
6. Experimental case study

For the experimental case study, a scaled VBI model composed of a two-axle vehicle and a simply supported beam is established to evaluate the viability of the proposed indirect SHM framework in more realistic scenarios. The addition of concentrated masses at various locations is employed to simulate damaged conditions on the structure as per the approach used by [31–34]. Four different states, including the intact state or healthy state, are considered. This section provides an overview of the experimental setup, identical to the one utilized in [4] and the utilization of this dataset to validate the methodology presented in Section 5.

6.1. Experimental set-up

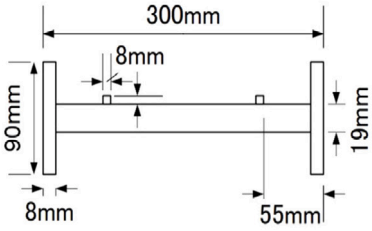
The laboratory-scale bridge comprises a steel beam with an I-section featuring three supported spans, oriented such that the web is in the horizontal (weak) direction (see Fig. 13(d)). The central 5.4 m of the structure constitutes the target span, with the initial and final 2.5 m allocated for vehicle acceleration and deceleration. The boundaries of this target region are designed as simple supports (see Figs. 13(b) and 13(c)). A pair of 8 mm rails, laser-engraved to achieve a road profile class A, are positioned across the

(a) Bridge model, motor, and pulley system.



(b) Schematic of the beam model (dimensions are in millimeters).

(c) Simply-support boundary condition.



(d) Cross section of the beam.

Fig. 13. Illustration of the test setup with details of the scaled bridge model [4].

Table 5
Summary of four bridge conditions considered in the laboratory.

Label	Description
B0	Intact bridge
BM1	Bridge with 8.7 kg mass added at mid-span
BM2	Bridge with 16.1 kg mass added at mid-span
BQ1	Bridge with 9.4 kg mass added at quarter-span

Table 6
Summary of the first bending mode of each experimental bridge state[a].

Bridge state	Healthy	BM1	BM2	BQ1
Mode 1	3.60(–)	3.54(1.64%)	3.51(2.50%)	3.42(5.00%)

[a] The frequency results are in Hz (% change relative to healthy state).

beam, guiding the vehicle as it traverses the bridge. An electric motor and pulley system are employed to move the car, with an electronic controller ensuring a constant speed [4] (see Fig. 13(a)).

To simulate the healthy bridge state, B0, no additional mass was added to the bridge. To simulate damaged bridge states, masses of 8.7 kg and 16.1 kg are added at the mid-span, along with a 9.4 kg mass at the quarter-span position. These masses are chosen to replicate the impact of actual damage on the dynamic properties of the bridge [4]. Table 5 lists these three bridge conditions with corresponding labels: BM1, BM2, BQ1. Table 6 summarizes the first bending modes for each of these damage states [4]. For more detailed information on the bridge space and the determination of these modes, refer to [4].

Regarding the vehicle, a scaled two-axle mode is utilized. The spring stiffness is determined to be 35,693 N/m, and the axle distance measured 400 mm. Acceleration transducers (TML ARS-A) from Tokyo Measuring Instruments Laboratory Co. are mounted on the beam at quarter-span, mid-span, and three-quarter-span positions, as well as on the vehicle itself, centered on the front and rear axles and the center of gravity. These transducers are used to measure the bouncing and pitching modes of the car. A sampling frequency of 200 Hz is employed throughout the experiments [4].

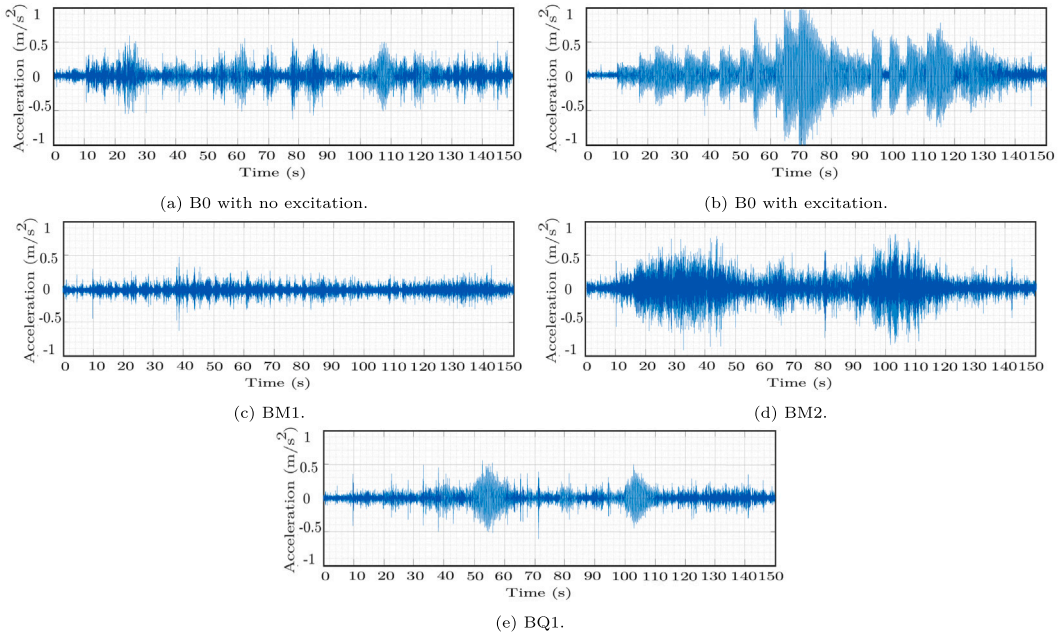


Fig. 14. Raw experimental acceleration signals for the front axle for each severity of damage.

6.2. Experimental dataset

The bridge and vehicle responses are simultaneously recorded for 50 forward, and 50 backward passes over the bridge for each bridge state. Using an electronic controller, the vehicle's speed is maintained at a constant value of 0.04 m/s. To introduce natural variability caused by environmental disturbances, artificial excitation is randomly applied, ranging from no excitation to gentle and sharp excitation patterns created by randomly striking the bridge with hands. Figs. 14 and 15 provide examples of the raw signals obtained from this experiment with randomly applied excitation. Notably, the acceleration response amplitude is substantially lower when no excitation is applied.

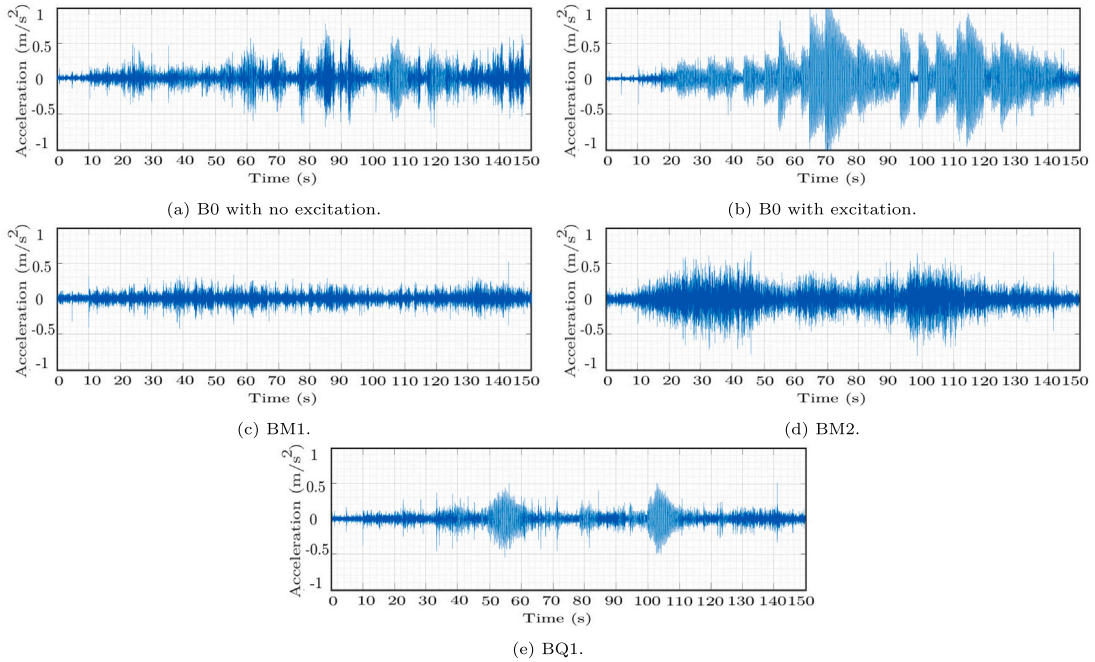
The resultant dataset contains a healthy bridge state and three distinct damaged states (Table 5). The same preprocessing method is employed using a [3.1–3.9] Hz filter, generating 5000 averaged signals for the healthy dataset and 100 averaged samples for each damage scenario. The result of this process on the experimental signals is depicted in Fig. 16 which shows the improved differentiation of different damage states. An additional logarithm transformation is then applied to prevent the clustering of anomalous data at zero by reducing the differences between large values whilst preserving the differences in small values and effectively reducing the impact of noise. The α_j is then calculated using the same method involving IDK. This dataset serves to validate the proposed framework. All boxplots were created using the same format as the previous section.

6.3. Damage severity assessment and the effect of damage location

Fig. 17 shows the boxplot of the α_j for each of these damage states applied to the entire experimental dataset following the same approach as Sections 5.6 and 5.8. This figure highlights the ability of the algorithm to clearly distinguish healthy from damaged states and distinguish between damaged states when the algorithm is applied to the experimental dataset. The figures show that the algorithm clearly delineates between the healthy state B0 and each of the other three states. It also shows the ability to quantify the damage, clearly distinguishing between milder (BM1) and more severe (BM2) damage states, with lower α_js corresponding to a greater level of damage. Additionally, the figure illustrates the algorithm's proficiency in detecting damage within a quarter-span (BQ1). This indicates the algorithm's ability to pinpoint damage in specific locations along the bridge, further showcasing its robustness in assessing and isolating varying damage levels across different areas. This ability to quantify damage severity is advantageous in practice by allowing for prioritizing incurred bridge damage. It reiterates the high potential of the proposed IDK damage detection and assessment framework for drive-by bridge inspection.

6.4. Effect of sample size

To test the effectiveness of the IDK approach on the experimental dataset, the ability to detect and quantify damage based on reference sample size is assessed using the same method as Section 5.7. To this aim, various sample sizes of B0 signals (2,10,100)



(a) B0 with no excitation.

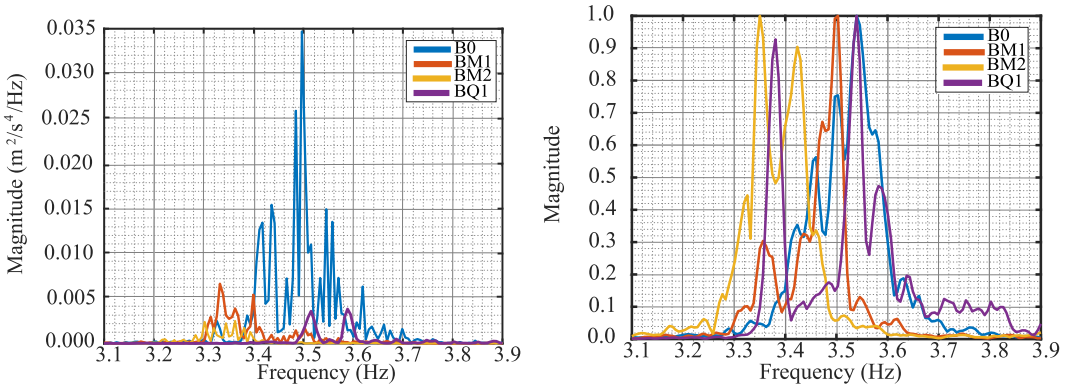
(b) B0 with excitation.

(c) BM1.

(d) BM2.

(e) BQ1.

Fig. 15. Raw experimental acceleration signals for the rear axle for each severity of damage.



(a) Residual frequency spectrum.

(b) Pre-processed spectrum.

Fig. 16. Comparison of residual frequency spectrums (PSD) before and after preprocessing experimental dataset.

were combined with each BM1, BM2, and BQ1 signal to determine the algorithm's capacity to isolate a damage signal from healthy cases. These sample sizes were chosen to depict the trend.

Fig. 18 depicts this effect of B0 sample size on α_j, displaying the algorithm's strong ability to perform binary damage classification despite minimal healthy reference data, clearly distinguishing between healthy and damage states with only two B0 signals with 100% accuracy (Fig. 18(a)). These findings reiterate the usefulness of the IDK approach in practice. However, regardless of the B0 sample size the approach cannot quantify damage and distinguish between the damage states BM1, BM2, and BQ1 without reference signals from other damage states (Figs. 18(a)–18(c)).

To further quantize these findings for classification accuracy, a threshold to define an anomaly was set as per Section 5.7. Table 7 summarizes these results, showing high binary classification accuracy across all sample sizes and damage states. However, like the numerical case study this decreases with higher healthy sample sizes and damage severity due to the relative nature of similarity scores. Regardless, the results retain a high level of accuracy.

7. Comparative study with AAE

To evaluate the effectiveness of the IDK algorithm, a comparison is made to the methodology used in [4] in terms of the extent of signal averaging required in the pre-processing step, damage assessment performance, damage scenarios classification accuracy, and

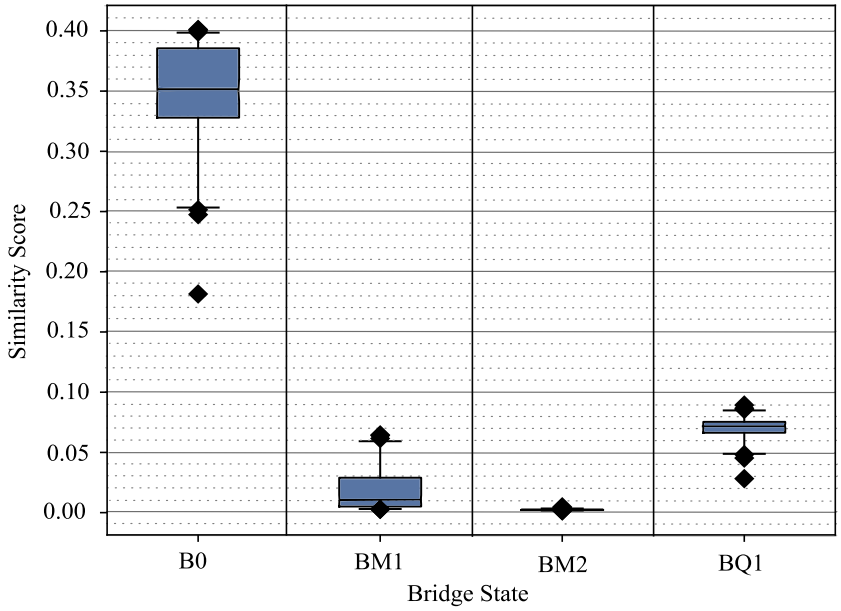


Fig. 17. Damage assessment results for the experimental case study using the complete dataset.

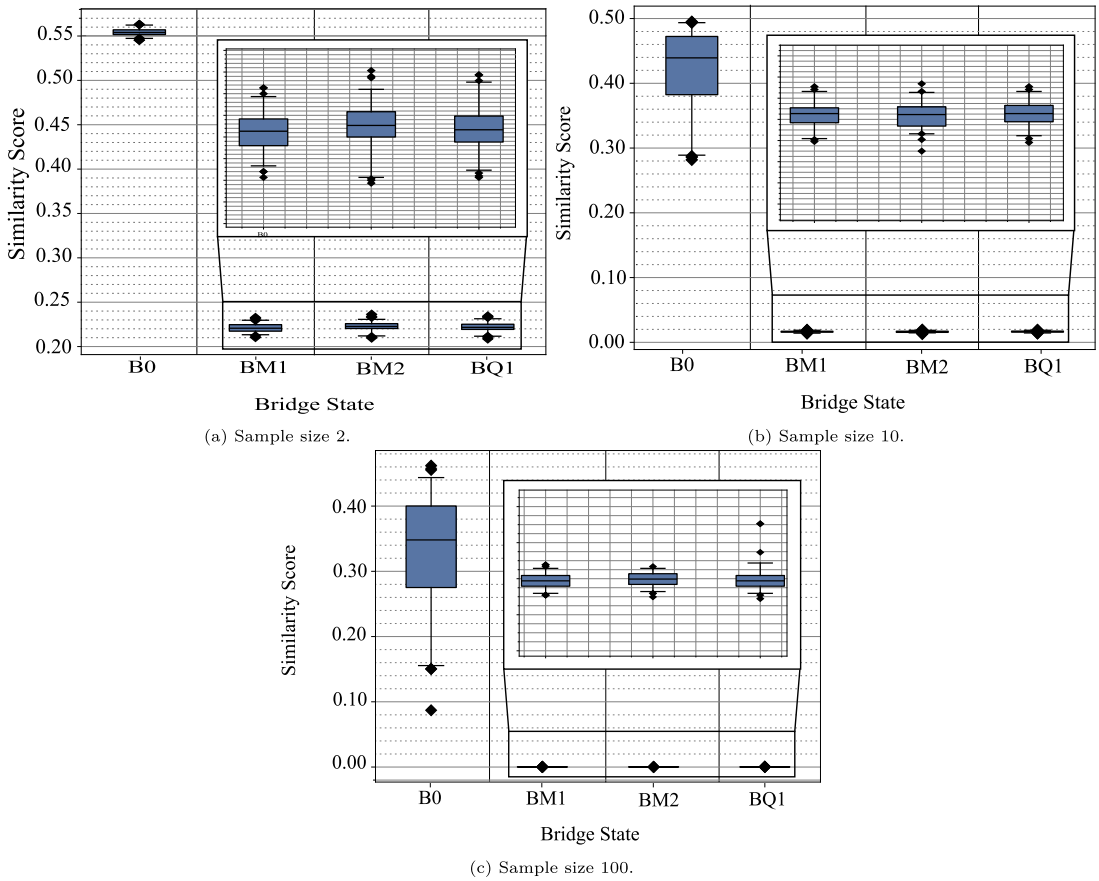


Fig. 18. Damage assessment results based on B0 signal sample size for experimental damage signals.

Table 7
Effect of sample size of classification accuracy metrics for the experimental case study[a].

Bridge state	Sample size	Metric			
		True positive	True negative	False positive	False negative
10% Damage	5	100	100	0	0
	10	100	100	0	0
	50	100	100	0	0
	100	100	99	1	0
	500	100	97	3	0
	1000	100	98	2	0
20% Damage	5	100	100	0	0
	10	100	100	0	0
	50	100	100	0	0
	100	100	99	1	0
	500	100	99	1	0
	1000	100	96	4	0
30% Damage	5	100	100	0	0
	10	100	100	0	0
	50	100	100	0	0
	100	100	99	1	0
	500	100	99	1	0
	1000	100	99	1	0

[a] The true positive rate refers to the rate of correctly identified damage signals whilst the true negative rate refers to the correctly identified healthy signals.

computational cost. To this end, damage at the mid-span scenario is considered. It is essential to emphasize that both methodologies are subjected to evaluation on the same computational system with specified components, including a 12th Generation Intel Core i7-12700H processor with 24 GB of RAM and a single NVIDIA GeForce RTX3070 graphics card with 20 GB of memory, all from the year 2022.

The methodology proposed by [4] is divided into two parts: (1) the data preprocessing step and (2) a damage assessment methodology using AAE. The data processing presented by [4] is the same as the one presented in Section 4. The second part uses an AAE model trained with healthy preprocessed frequency spectrums to assess damage similar to the present study.

During the testing phase, 100 preprocessed samples from each of the various bridge conditions are utilized for comparisons. The α_j for damage assessment is computed by taking the cosine similarity derived from the dot product between the original samples and their respective reconstructions

$$\alpha_j = \frac{\mathbf{a} \cdot \mathbf{b}}{\|\mathbf{a}\|\|\mathbf{b}\|} \tag{14}$$

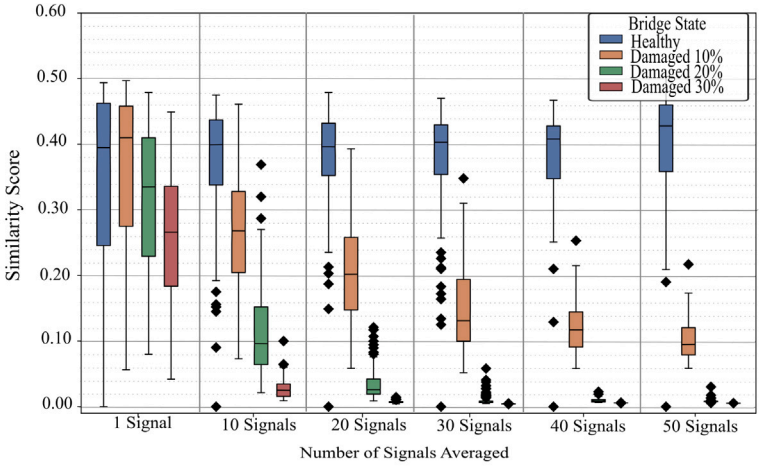
where $\mathbf{a}$ represents the original acceleration signal of magnitude $\|\mathbf{a}\|$, and $\mathbf{b}$ represents the reconstructed acceleration signal after the AAE model has been applied with magnitude $\|\mathbf{b}\|$. This ensures consistency in the assessment process across both methods. As the algorithm is exclusively trained on healthy samples, the α_j is anticipated to decrease when applied to damaged samples. Further details on the definition of the AAE can be found in [4]. Finally, the α_j scores obtained from both methodologies are normalized within the range of (0,1).

7.1. Signal averaging

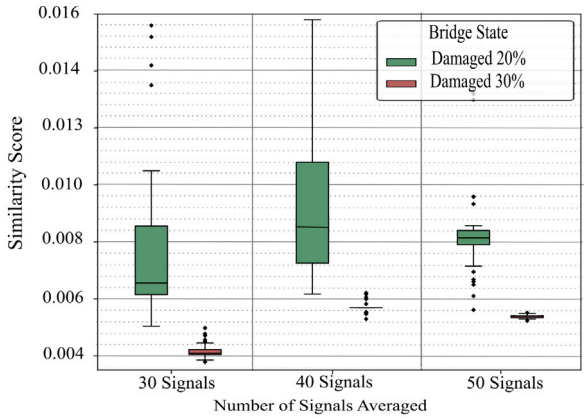
To compare the extent of signal averaging required in the pre-processing of the signals, we compare the number of signals averaged in the pre-processing are the following: $1, 10, 20, 30, 40$ and 50. The ability to distinguish between each of the bridge states is then assessed and compared between IDK and AAE. Fig. 19(a) displays the effect of signal averaging on the IDK algorithm, demonstrating the ability to distinguish between bridge states with an average of 10 signals. Meanwhile, Fig. 19(c) displays the effect of signal averaging on the AAE algorithm, depicting the ability to distinguish between groups with an average of 20 signals. This distinction demonstrates the key strength of the IDK algorithm in terms of data acquisition as the algorithm requires less data to produce separation of damage classes. This is extremely important in practice as obtaining large data sets is expensive and time-consuming in indirect SHM. The IDK also demonstrates high precision on the severity assessment task, with separation between damage classes achieved with less data variability whilst AAE demonstrates high precision for the healthy cases. This is another key benefit of the IDK algorithm as identification of damage is the primary goal of SHM.

7.2. Computational performance and accuracy

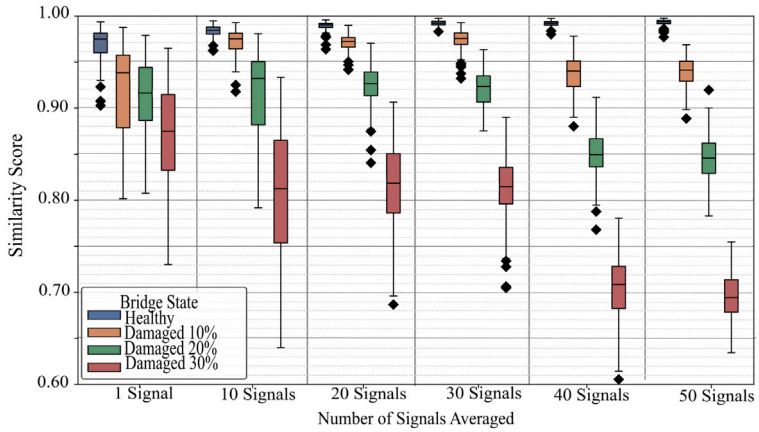
To compare the performance of the AAE-based approach and the IDK-based methodology introduced in this study, a dataset consisting of preprocessed samples obtained from the bridge's healthy state is employed for training the AAE model. This is performed with a pre-processing averaging of 30 signals.



(a) Effect of signal averaging on IDK damage classification.

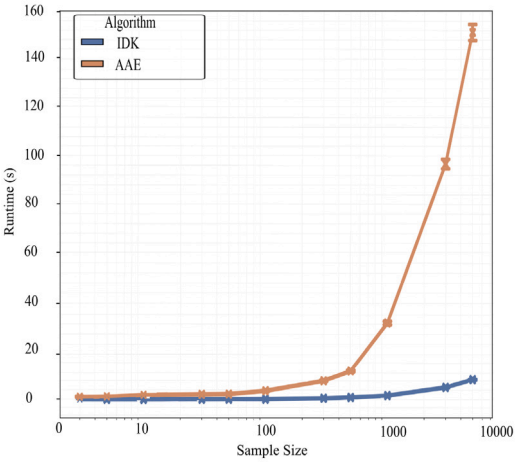


(b) Zoomed in version of the effect of signal averaging on IDK damage classification.

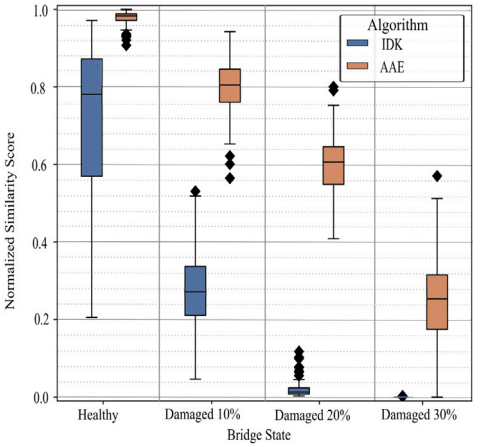


(c) Effect of signal averaging on AAE damage classification.

Fig. 19. Comparison of damage assessment between AAE and IDK for different levels of signal averaging, for damage located at mid-span.



(a) Comparison of runtime performance for various sample sizes demonstrating improved performance of IDK compared to AAE.



(b) Comparison of damage assessment performance with a healthy sample size of 5,000, demonstrating similar performance.

Fig. 20. Comparison of damage assessment and runtime performance between AAE and IDK, for damage located at mid-span and varying severities.

Table 8
Comparison of ROC-AUC scores between IDK and AAE for the damage severity assessment task.

Damage location	Bridge condition			
	Healthy	10% Damage	20% Damage	30% Damage
IDK	0.99	0.94	0.98	1.00
AAE	1.00	0.96	0.98	1.00

Fig. 20(a) presents a comparative analysis of the runtime between the IDK algorithm and that of the AAE across varying sample sizes. Notably, it becomes evident that the procedure for AAE requires significantly more computational time than IDK across all sample size scenarios. In a holistic assessment, it can be inferred that IDK exhibits an approximate 95% reduction in runtime compared to AAE. This observation holds considerable practical significance, particularly in the context of real-world applications of drive-by bridge monitoring, where datasets often exhibit intricate and extensive characteristics. Fig. 20(b) presents a comparison of the α_j scores achieved by the two methodologies, IDK and AAE, for assessing different bridge conditions using a sample size of 5000 (i.e., where AAE is trained with 5000 preprocessed healthy samples). Both IDK and AAE showcase their capability to discern the three distinct damage conditions of the bridge by exhibiting lower similarity indices in contrast to the healthy state. Moreover, it is conspicuous in both approaches that as the severity of damage escalates, the α_j diminishes, thereby enabling a reliable assessment of damage severity.

Furthermore, a means of comparing the performance of the IDK and AAE methodologies is implemented, similar to the procedure outlined in Section 5.6. The logistic regression model is applied to the α_j scores generated by both models, utilizing a sample size of 5000 signals. The objective is to assess the accuracy of classifying the bridge's condition using the ROC-AUC methodology described in Section 5.6. The outcomes of the classification accuracy for each methodology are documented in Table 8, which illustrates that IDK presents a marginally diminished classification accuracy in discerning the bridge conditions. This discrepancy in accuracy may be attributed to the relatively higher degree of overlap observed between the α_j distributions corresponding to the healthy and 10% damage conditions. However, it is crucial to underscore that this observation, in isolation, does not provide conclusive evidence regarding the superiority of one method over the other.

7.3. Overall comparison

Thus, it can be affirmed that both methodologies exhibit a capacity for precise damage detection and the assessment of damage severity within the context of drive-by bridge monitoring. Nevertheless, it is imperative to underscore the notable efficiency of the IDK algorithm, in terms of computational time, which stands out as a crucial determinant in the domain of damage assessment where datasets are characterized by their complexity and dimension. It is also important to recognize that the IDK algorithm requires substantially less signal averaging to distinguish between bridge states, reducing the associated cost of signal obtainment in practice. Table 9 summarizes these findings in terms of accuracy, F1 score, precision, and recall, for varying numbers of signal averages,

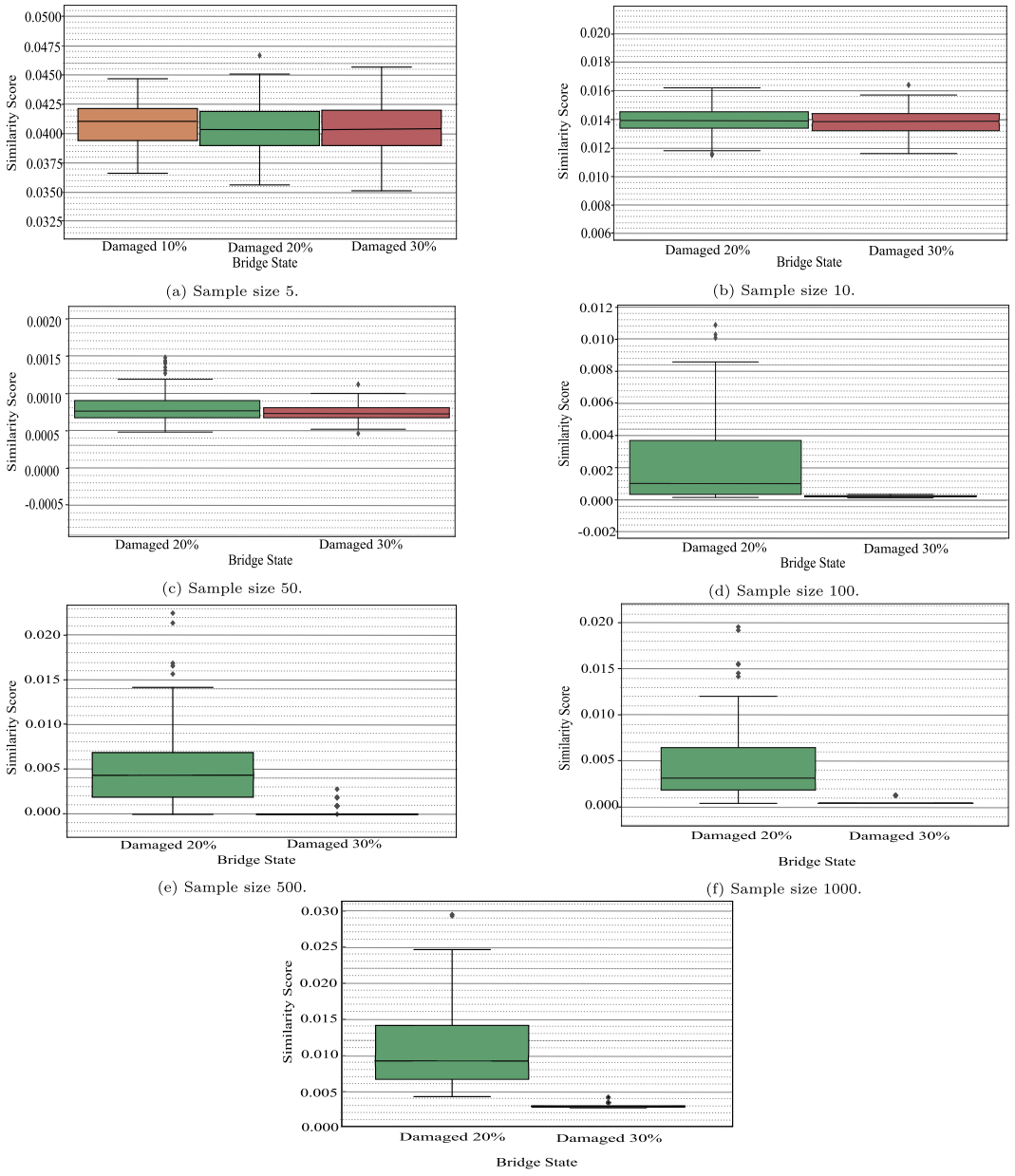
Table 9
Comparisons of model assessment averaged metrics for IDK and AAE by varying the amount of signal averaging in pre-processing.

Algorithm	Average size	Class metric			
		Accuracy	F1 Score	Precision	Recall
IDK	1	38%	35%	40%	39%
	3	48%	47%	46%	48%
	5	59%	58%	58%	59%
	8	67%	66%	66%	67%
	10	79%	78%	78%	79%
	15	77%	77%	77%	77%
	20	86%	86%	87%	86%
	25	87%	87%	87%	87%
	30	91%	91%	91%	91%
	40	93%	92%	93%	92%
	50	92%	93%	92%	92%
AAE	1	50%	48%	48%	50%
	3	46%	46%	47%	46%
	5	50%	49%	50%	50%
	8	64%	64%	64%	64%
	10	67%	67%	67%	67%
	15	71%	71%	71%	71%
	20	90%	90%	90%	90%
	25	87%	87%	87%	87%
	30	91%	91%	92%	91%
	40	97%	97%	98%	97%
	50	99%	99%	99%	99%

demonstrating improved metrics for the IDK methods compared to the AAE approach for lower samples averaged. In this table, we can see that, for sample sizes 3–15, the IDK approach has a higher average class matrics. For higher sample sizes of 20–30 signals, the methods achieve similar metrics. However, for larger sample sizes, AAE exceeds the performances of IDK due to the significant overfitting of the AAE approach. Collectively, the findings demonstrate the benefits of IDK in terms of cost in SHM, requiring less data to be obtained and improving run-time compared to deep learning methods such as AAE, improving the cost-to-benefit ratio of drive-by SHM.

8. Conclusion

In this study, we proposed a semi-supervised framework based on the IDK approach for anomaly detection to address the outstanding issues with indirect SHM. To this aim, preprocessed numerical and experimental datasets are mapped to higher-order space, and the resulting distribution is used to calculate a α_j. Our investigations successfully identified the severity of bridge states at all locations in the simulated and experimental datasets with substantial accuracy above 94%. In addition, the algorithm benefitted by requiring minimal prior training, allowing for greater computational efficiency compared to traditional machine learning methods. Compared to the competing drive-by techniques, this proposed method successfully operated on a limited sample size with less pre-processed signal averaging than predecessor approaches. Overall, the findings of this study demonstrate how the IDK algorithm is a useful and reliable approach to real-world SHM by requiring significantly less data than equivalent deep learning methods, improving the cost-to-benefit ratio of drive-by SHM. However, the authors would like to acknowledge the limitations and assumptions of the technique. First, the methodology assumes that the same test vehicle is employed for all monitoring of the bridge. Second, it also assumes that the vehicle passes at a constant and slow test speed to enhance spectral resolution which is impractical to achieve. To extend this methodology to its full potential, four research paths should be undertaken. First, a second layer of the IDK algorithm (IDK^2) should be applied to more complicated datasets that better simulate real-life conditions to perform group anomaly detection such as with varying vehicle speeds and environmental conditions. Additionally, the impact of different parameters in these scenarios should be further investigated. Second, the variation of frequency over time can be studied or the application of IDK to segments of each time series should be considered to identify the exact location of the anomaly which can be correlated with the location of bridge damage. Third, its applicability to be used in real-time data processing should be explored due to its minimal training and computation efficiency offering significant benefits in such applications. Finally, the algorithm should be validated in a practical setting on data collected from real-world bridges. In addition, the IDK approach could be applied in other areas of civil engineering outside of SHM for anomaly detection such as other concrete structures or railway systems [35,36].



(a) Sample size 5.

(b) Sample size 10.

(c) Sample size 50.

(d) Sample size 100.

(e) Sample size 500.

(f) Sample size 1000.

(g) Sample size 5000.

Fig. A.21. Additional zoomed-in plots of Fig. 12 for each sample size. These demonstrate the improved ability to distinguish between damage classes as sample size is increased.

CRediT authorship contribution statement

G. Tyler: Writing – review & editing, Writing – original draft, Validation, Conceptualization. **S. Luo:** Supervision, Investigation, Funding acquisition, Conceptualization. **A. Calderon Hurtado:** Writing – original draft, Investigation, Formal analysis, Conceptualization. **M. Makki Alamdari:** Supervision, Methodology, Funding acquisition, Conceptualization.

Declaration of competing interest

The corresponding author declares that there is no conflict of interest for this manuscript.

Acknowledgments

The authors would like to express their gratitude to the Australian Research Council (ARC) for their support under the Discovery Early Career Research Award (DECRA) scheme, grant number DE210101625. Thanks are also extended to Professor C.W. Kim and Professor K.C. Chang and their research group for their help in conducting the experiments. The authors also wish to thank the Japan Society for the Promotion of Science (JSPS) for their support in conducting this research. Additionally, this research was supported under the Australian Research Council's Industrial Transformation Research Programme (IH210100048).

Appendix. Additional experimental results

A.1. Effect of sample size on damage identification

Fig. A.21 contains zoomed figures for each sample size (5, 10, 50, 100, 500, 1000, 5000) of Fig. 11 damage identification and severity assessment boxplots. These figures demonstrate the improved ability to distinguish between bridge states as sample size increases for the IDK approach to SHM. Specifically, Fig. A.21(a) shows that for a sample size of 5, it is not possible to distinguish between any of the damage states. However, as sample sizes increase this improves, as seen in Figs. A.21(d)–A.21(g) which shows the ability to clearly distinguish between states.

Data availability

Data will be made available on request.

References

[1] Y.B. Yang, K.C. Chang, Y.C. Li, Filtering techniques for extracting bridge frequencies from a test vehicle moving over the bridge, Eng. Struct. 48 (2013) 353–362, http://dx.doi.org/10.1016/j.engstruct.2012.09.025.
[2] M.M. Alamdari, An evolutionary vehicle scanning method for bridges based on time series segmentation and change point detection, Mech. Syst. Signal Process. 210 (2024) 111173, http://dx.doi.org/10.1016/j.ymssp.2024.111173, URL: https://www.sciencedirect.com/science/article/pii/S0888327024000712.
[3] A. Malekjafarian, P.J. McGetrick, E.J. Obrien, A review of indirect bridge monitoring using passing vehicles, Shock Vib. 2015 (2015) 1–16, http://dx.doi.org/10.1155/2015/286139.
[4] A. Calderon Hurtado, K. Kaur, M. Makki Alamdari, E. Atroshchenko, K.C. Chang, C.W. Kim, Unsupervised learning-based framework for indirect structural health monitoring using adversarial autoencoder, J. Sound Vib. 550 (2023) http://dx.doi.org/10.1016/j.jsv.2023.117598.
[5] K. Ting, Z. Liu, H. Zhang, Y. Zhu, A new distributional treatment for time series and an anomaly detection investigation, Proc. VLDB Endow. 15 (11) (2022) 2321–2333, http://dx.doi.org/10.14778/3551793.3551796.
[6] Y.B. Yang, C.W. Lin, J.D. Yau, Extracting bridge frequencies from the dynamic response of a passing vehicle, J. Sound Vib. 272 (3–5) (2004) 471–493, http://dx.doi.org/10.1016/s0022-460x(03)00378-x.
[7] P. McGetrick, A. Gonzalez, E. OBrien, Theoretical investigation of the use of a moving vehicle to identify bridge dynamic parameters, Insight, Non-Destr. Test. Cond. Monit. 51 (2009) 433–438, http://dx.doi.org/10.1784/insi.2009.51.8.433.
[8] M. Silva, A. Santos, R. Santos, E. Figueiredo, J.C. Costa, Damage-sensitive feature extraction with stacked autoencoders for unsupervised damage detection, Struct. Control Health Monit. 28 (5) (2021) e2714, http://dx.doi.org/10.1002/stc.2714, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/stc.2714, URL: https://onlinelibrary.wiley.com/doi/abs/10.1002/stc.2714, e2714 stc.2714.
[9] Q. Mei, M. Gül, M. Boay, Indirect health monitoring of bridges using Mel-frequency cepstral coefficients and principal component analysis, Mech. Syst. Signal Process. 119 (2019) 523–546, http://dx.doi.org/10.1016/j.ymssp.2018.10.006, URL: https://www.sciencedirect.com/science/article/pii/S0888327018306678.
[10] P. Cheema, M.M. Alamdari, K.C. Chang, C.W. Kim, M. Sugiyama, A drive-by bridge inspection framework using non-parametric clusters over projected data manifolds, Mech. Syst. Signal Process. 180 (2022) http://dx.doi.org/10.1016/j.ymssp.2022.109401.
[11] O. Abdeljaber, O. Avci, S. Kiranyaz, M. Gabbouj, D. Inman, Real-time vibration-based structural damage detection using one-dimensional convolutional neural networks, J. Sound Vib. 388 (2017) 154–170, http://dx.doi.org/10.1016/j.jsv.2016.10.043.
[12] L. Sun, Z. Shang, Y. Xia, S. Bhowmick, S. Nagarajaiah, Review of bridge structural health monitoring aided by big data and artificial intelligence: From condition assessment to damage detection, J. Struct. Eng. 146 (2020) 04020073.
[13] D. Hajializadeh, Deep learning-based indirect bridge damage identification system, Struct. Health Monit. 22 (2) (2022) 897–912, http://dx.doi.org/10.1177/14759217221087147.
[14] A. Malekjafarian, F. Golpayegani, C. Moloney, S. Clarke, A machine learning approach to bridge-damage detection using responses measured on a passing vehicle, Sensors (Basel) 19 (18) (2019) http://dx.doi.org/10.3390/s19184035, URL: https://www.ncbi.nlm.nih.gov/pubmed/31546759.
[15] X. Wang, E.P. Xing, D.J. Schaid, Kernel methods for large-scale genomic data analysis, Brief. Bioinform. 16 (2) (2015) 183–192.
[16] K. Ting, B. Xu, T. Washio, Z. Zhou, Isolation distributional kernel: A new tool for kernel based anomaly detection, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, KDD '20, Association for Computing Machinery, New York, NY, USA, 2020, pp. 198–206, http://dx.doi.org/10.1145/3394486.3403062.
[17] R. Desai, A. Guha, P. Seshu, A comparison of quarter, half and full car models for predicting vibration attenuation of an occupant in a vehicle, J. Vib. Eng. Technol. 9 (5) (2021) 983–1001, http://dx.doi.org/10.1007/s42417-020-00278-3.
[18] M. Makki Alamdari, K. Chang, C. Kim, K. Kildashti, H. Kalhori, Transmissibility performance assessment for drive-by bridge inspection, Eng. Struct. 242 (2021) 112485, http://dx.doi.org/10.1016/j.engstruct.2021.112485, URL: https://www.sciencedirect.com/science/article/pii/S0141029621006350.
[19] K.-J. Bathe, Finite Element Procedures, Klaus-Jurgen Bathe, 2006.
[20] ISO 8606:2016(E), Mechanical Vibration - Road Surface Profiles - Reporting of Measured Data, Standard, International Organization for Standardization, Geneva, CH, 2016.

[21] J. Sinha, M. Friswell, S. Edwards, Simplified models for the location of cracks in beam structures using measured vibration data, J. Sound Vib. 251 (1) (2002) 13–38, http://dx.doi.org/10.1006/jsvi.2001.3978, URL: https://www.sciencedirect.com/science/article/pii/S0022460X01939781.
[22] R. Debnath, H. Takahashi, Kernel selection for the support vector machine, IEICE Trans. Inf. Syst. 87 (12) (2004) 2903–2904.
[23] S.-J. Kim, A. Magnani, S. Boyd, Optimal kernel selection in kernel fisher discriminant analysis, in: Proceedings of the 23rd International Conference on Machine Learning, 2006, pp. 465–472.
[24] S. Ali, K.A. Smith-Miles, A meta-learning approach to automatic kernel selection for support vector machines, Neurocomputing 70 (1–3) (2006) 173–186.
[25] K. Yang, S. Kpotufe, N. Feamster, An efficient one-class SVM for anomaly detection in the internet of things, 2021, arXiv preprint arXiv:2104.11146.
[26] R. Piziak, P.L. Odell, Full rank factorization of matrices, Math. Mag. 72 (3) (1999) 193–201.
[27] Y. Yang, Y. Li, K. Chang, Using two connected vehicles to measure the frequencies of bridges with rough surface: A theoretical study, Acta Mech. 223 (2012) 1851–1861, http://dx.doi.org/10.1007/s00707-012-0671-7.
[28] I. Mekjavic, Identification of structural damage in bridges using high-frequency vibrational responses, Shock Vib. 2015 (2015) 1–8, http://dx.doi.org/10.1155/2015/906062.
[29] K.M. Ramachandran, C.P. Tsokos, Chapter 12 - Nonparametric statistics, in: M.R. Kandethody, P.T. Chris (Eds.), Mathematical Statistics with Applications in R (Third Edition), third ed., Academic Press, 2021, pp. 491–530, http://dx.doi.org/10.1016/B978-0-12-817815-7.00012-9, URL: https://www.sciencedirect.com/science/article/pii/B9780128178157000129.
[30] A. Dinno, Nonparametric pairwise multiple comparisons in independent groups using dunn's test, Stata J. 15 (1) (2015) 292–300, http://dx.doi.org/10.1177/1536867X1501500117, arXiv:https://doi.org/10.1177/1536867X1501500117.
[31] C.-W. Kim, K.-C. Chang, P.J. McGetrick, S. Inoue, S. Hasegawa, Utilizing moving vehicles as sensors for bridge condition screening-A laboratory verification, Sens. Mater. 29 (2) (2017) 153–163.
[32] F. Cerda, J. Garrett, J. Bielak, P. Rizzo, J. Barrera, Z. Zhuang, S. Chen, M.T. McCann, J. Kovacevic, Indirect structural health monitoring in bridges: scale experiments, 2012.
[33] G. Lederman, Z. Wang, J. Bielak, H. Noh, J.H. Garrett, S. Chen, J. Kovacevic, F. Cerda, P. Rizzo, Damage quantification and localization algorithms for indirect SHM of bridges, in: Proc. Int. Conf. Bridge Maint., Safety Manag., Shanghai, China, 2014, pp. 640–647.
[34] J. Liu, S. Chen, M. Bergés, J. Bielak, J.H. Garrett, J. Kovačević, H.Y. Noh, Diagnosis algorithms for indirect structural health monitoring of a bridge model via dimensionality reduction, Mech. Syst. Signal Process. 136 (2020) 106454.
[35] J. Chow, Z. Su, J. Wu, P. Tan, X. Mao, Y. Wang, Anomaly detection of defects on concrete structures with the convolutional autoencoder, Adv. Eng. Inform. 45 (2020) 101105, http://dx.doi.org/10.1016/j.aei.2020.101105, URL: https://www.sciencedirect.com/science/article/pii/S1474034620300744.
[36] M. Koohmishi, S. Kaewunruen, L. Chang, Y. Guo, Advancing railway track health monitoring: Integrating GPR, InSAR and machine learning for enhanced asset management, Autom. Constr. 162 (2024) 105378, http://dx.doi.org/10.1016/j.autcon.2024.105378, URL: https://www.sciencedirect.com/science/article/pii/S0926580524001146.