Journal of Applied Automation Technologies

Research Paper

Noise Reduction in Multi-Sensor Systems Using Graph Neural Networks

Ludmiła Antczakowa[1,*] and Marianna Ciołek[1]

[1] Faculty of Automation and Electrical Engineering, Jan Kochanowski University in Kielce, Kielce, 25-020, Poland
*Corresponding author: ludmila.ant@ujk.edu.pl

Abstract. Multi-sensor networks have solved several issues in the era of smart agriculture and autonomous driving. Nevertheless, there are numerous sources of noise in the sensor data, which lowers the quality of the information and thus impairs the performance of later phases. In order to solve the issue of sensor data noise and enhance the dependability of spatially correlated sensor arrays, this paper uses Graph Neural Networks (GNNs). This creates a weighted graph of the sensor network, builds an adjacency matrix based on the statistical similarity and physical distance between sensors, and then uses a specific Graph Neural Network (GNN) architecture to guarantee consistent signal recovery. In a series of tests, varying levels of Gaussian and impulse noise, as well as different rates of missing data, were routinely added to both synthetic and real-world sensor datasets. The experiment's findings indicate that the new technique has improved the signal-to-noise ratio by an average of about 17.5 dB and is always better than previous denoising techniques like filtering and autoencoders. Additionally, the framework maintains acceptable performance across all deployment scenarios and sensor kinds, and it has good scalability for large networks of up to 2,000 sensor nodes. In summary, this article has demonstrated the great effectiveness and adaptability of Graph Neural Networks (GNNs) for noise reduction in multi-sensor fusion, offering a reliable method of reducing noise and obtaining high-quality data from a variety of real-world scenarios.

Keywords: *Sensor Networks, Data Denoising, Signal Processing, Graph Neural Networks*

Received on 29 November 2024, Accepted on 29 April 2025, Published on 05 May 2025

Copyright © 2025 Author(s), licensed to JAAT. This is an open access article distributed under the terms of the CC BY-NC-SA 4.0, which permits copying, redistributing, remixing, transformation, and building upon the material in any medium so long as the original work is properly cited.

Introduction

Many parts of modern engineering and science now require multiple-sensor systems, such as self-driving cars and robots; environmental monitoring systems and smart hospitals are other examples. Using the strengths of several kinds of sensors together, these systems can build a complete picture of a scene more reliably and have a better fault-tolerance capability [1]. However, as the scale and diversity of sensor network deployment increase, the advantages mentioned earlier are increasingly limited by the pervasive problem of noise, which stems from various sources such as sensor drift, environmental fluctuations, fabrication errors, and wireless communication interference [2]. Noise damages both local measurements and overall data collection; therefore, the general performance of traditional data fusion methods declines, and subsequent applications in anomaly detection, tracking, decision-making, etc., are also negatively impacted [3]. Therefore, stable noise reduction and guaranteed data reliability are now required foundations for practical multi-sensor system deployment [4].

Over the past few decades, many ways have been introduced to reduce noise in multiple sensors through pre-processing, filtering and intelligent data fusion. Kalman filtering and statistical inference models are relatively simple, but they only work well in specific linear and structured cases for state estimation and uncertainty quantification [5]. Wavelet denoising is also a method for decomposing a signal and suppressing localised and non-stationary noise [6]. However, the above traditional methods are unable to handle high-dimensional and non-linear correlations in large-scale sensor networks, especially those with spatially correlated and dynamically changing noise patterns [7]. Recently, deep learning-based solutions have begun to be applied to the analysis of multiple sensor data due to their hierarchical feature extraction and non-linear modelling capabilities [8].

However, most of the existing neural networks treat the sensor data independently or combine it into a single vector without considering the spatial and topological relationships among different sensor nodes. Therefore, they are not ideal for denoising in a way that requires retaining relational information. Graph Neural Networks (GNNs) have recently been applied in many ways to learn from graph-structured data by considering the interactions among nodes and propagating information across edges [9]. The message-passing mechanism of GNNs can naturally incorporate the various types of complex relationships among sensors in real-world systems to obtain local and global structural information for better noise immunity [10].

Despite the above recent developments, systematic research on the combination of GNN-based models for noise reduction in multi-sensor fusion is still in its infancy. The primary open questions are: how to optimise the construction of the graph to represent sensor correlations; what suitable model architectures can be employed in noisy environments; and how to ensure the scalability of GNN solutions for large-scale and heterogeneous sensor arrays. A full-featured framework for reducing noise in multi-sensor systems based on graph neural networks is proposed in this paper. First, use graphs to organise the sensor data and show any connections or noise changes among the sensors. Then we will introduce a GNN-based denoising architecture and design a special training method to learn well under all kinds of noise. A number of all-encompassing simulations and real-world experiments have shown that our method performs better than both the traditional approach and current state-of-the-art deep learning-based baseline methods in terms of noise suppression and information preservation. The rest of this paper is organised as follows: Section 2 introduces related work on multi-sensor fusion, noise reduction and GNN modelling; Section 3 presents the proposed method in detail, including graph construction and network design; Section 4 describes the experimental setup and visualisation; Section 5 shows and discusses the experimental results; finally, Section 6 provides a summary and future research directions.

Related Work

Multi-Sensor Data Fusion

As demonstrated in [11], numerous sensors have been incorporated to create robust perception capabilities for robots, vehicles, and environments. The initial research on multi-sensor fusion was mostly statistical and probabilistic, taking into account sensor noise and uncertainty models for recursive estimation. Algorithms like the traditional Kalman filter and its nonlinear expansions are frequently used for state estimation and tracking, while Bayesian inference and different probabilistic graphical models have been utilised to combine uncertain or partial measurements [12,13]. While these techniques work well in linear-Gaussian scenarios, they have been demonstrated to have serious shortcomings when dealing with high-dimensional, non-linear, or non-Gaussian situations. This scenario is becoming more common as sensor systems get more complicated [14].

Deep learning has recently been the foundation for the advancement of data-driven fusion. In image and remote-sensing activities, as well as other multimodal perception applications, convolutional and recurrent neural network topologies have been used to improve feature extraction and non-linear integration of various sensor streams [15]. Nevertheless, the majority of these models do not take into consideration the spatial and relational complexity of the underlying sensor topology since they flatten or concatenate sensor streams [16]. There is a chance of inter-sensor correlation and a decline in the quality of the data since it is comparatively sensitive to noise or change. Recently, researchers have shown that maintaining or explicitly introducing linkages across sensors can improve the estimation's robustness and precision in the face of unfavourable situations [17].

Noise Reduction Methods

The first is to minimise noise in the translation of physical data to digital signals; otherwise, the subsequent inference may contain mistakes and artefacts due to variations in sensor quality, external noise, and complicated operational settings [18]. The majority of outdated techniques for reducing noise are filters. Although the mean, median, and Wiener filters are comparatively easy methods to minimise impulsive and Gaussian noise, they may potentially generate artefacts or lose some information in complicated signals [19]. Other time-frequency and adaptive transform techniques that extend the aforementioned features to handle non-stationary or spatially-varying noise include wavelet denoising and empirical mode decomposition; however, their use typically necessitates a significant amount of computation and parameter optimisation [20].

By accurately characterising the statistical characteristics of the signal and corruption, model-based techniques like maximum likelihood estimation and state-space models attempt to reduce noise [21]. Though theoretically valid, these approaches are susceptible to performance deterioration when unmodeled dependencies or nonlinearities occur because they may not be able to handle the complexity of the real world under "model mismatch" [22]. Autoencoders and generative models have lately drawn attention to neural-network-based denoising; nonetheless, they often assume that the fused sensor inputs are independent or have a static relationship [23]. More research is being done to create techniques that can learn both inter-sensor dependencies and time-varying network topologies instead of using fixed, predefined correlations because recent studies have demonstrated the value of structural and spatial relationships in sensor data [24].

Graph Neural Network Developments

Recently, Graph Neural Networks (GNNs) have been developed to address issues with structured and relationship-based data. GNNs can handle both local and long-range relationships among various sensors in a systematic manner by describing the data as a graph, where the nodes are sensors and the edges are links (such as logical, statistical, or physical) [25]. In order to learn topology, heterogeneity, and other latent correlations that are unavailable to traditional neural networks, the fundamental GNN architectures of graph convolutional networks and attention-based models all use a message-passing technique to iteratively collect information in the neighbourhood.

GNNs are currently being used in a multi-sensor context to solve problems including data denoising, fault detection, sensor calibration, and spatial interpolation. The majority of recent research has demonstrated that Graph Neural Networks (GNNs) greatly outperform traditional and conventional deep models in environments with structured, non-uniform, or correlated noise when sensor arrays are viewed as graphs (through physical proximity, communication links, or learned similarity). For real-world heterogeneous sensor networks, a scalable and robust inference framework has been developed using dynamic weighted edges, flexible message passing, and hierarchical pooling. Building an accurate and efficient graph for dynamic or incomplete networks, solving the scalability issue with an increasing number of nodes, and creating training strategies appropriate for sparse and noisy annotations are just a few of the numerous issues that remain with the potential of graphs. In order to extend the benefits of topology learning and nonlinear function representation in the face of more complicated sensor settings, constantly new dynamic GNN architectures and hybrid graph-deep frameworks are being presented.

Methodology

Graph Construction for Sensor Data

Transforming raw sensor deployments into an appropriate graph structure is foundational for topology-aware learning and noise-resilient fusion. In this framework, the sensor network is interpreted as a weighted, undirected graph $G = (V, E, W)$, where each sensor is abstracted as a node, and meaningful physical, statistical, or communication-based connections are mapped as edges. The precise mathematical encoding of these relationships is captured in the adjacency matrix, which underpins not only spatial correlations but also potential information flow.

Each node is represented by a multi-dimensional feature vector, for example, $\mathbf{x}_i = \left[s_i^{(1)}, s_i^{(2)}, \ldots, s_i^{(d)}\right]$, where d reflects the breadth of data modalities associated with the sensor. This vector may include not only primary measurement data but also meta-information such as sensor health status, calibration factors, or time indices, improving the model's ability to infer subtle interdependencies.

The process of constructing edges involves applying a neighborhood rule that reflects the physical layout, communication reachability, or empirically determined similarity between sensors. Specifically, an edge is created between nodes i and j if the physical distance or correlation metric $d(i, j)$ is less than a predefined threshold τ :

$$E = \{(i, j) \mid d(i, j) \leq \tau\} \quad \text{Eq. (1)}$$

This abstraction allows for flexibility in heterogeneous networks, supporting various layout geometries or wireless topologies.

Edge weights in the adjacency matrix A can then be defined by physics-inspired or data-driven functions. A common approach is to let the weight decay with distance via a Gaussian kernel:

$$w_{ij} = \exp\left(-\frac{d(i,j)^2}{\sigma^2}\right) \quad \text{Eq. (2)}$$

where σ controls the kernel's sensitivity to spatial separation. Other applications might prefer direct statistical similarity or channel capacity as edge metrics to more accurately encode practical operational relationships.

As a result, the adjacency matrix A takes the form:

$$A_{ij} = \begin{cases} w_{ij}, & \text{if nodes } i \text{ and } j \text{ are connected} \\ 0, & \text{otherwise} \end{cases} \quad \text{Eq. (3)}$$

This matrix not only serves as the computational template for subsequent GNN operations, but also encapsulates all prior knowledge regarding interaction likelihood and topology.

In dynamic, real-world deployments, it is further possible to adapt this graph topology in real time:

$$A_{ij}(t) = f\big(d(i,j,t), q_{ij}(t)\big) \quad \text{Eq. (4)}$$

where $q_{ij}(t)$ may represent temporal link quality, environmental effects, or other time-varying properties influencing connectivity.

The entire process-from raw sensor array to structured, weighted graph-systematically captures the spatial and functional realities of the network, enabling a principled basis for robust graph-based learning. An intuitive summary of this workflow is depicted in Figure 1, highlighting the stages of node abstraction, edge selection, weight assignment, and the resulting adjacency matrix.

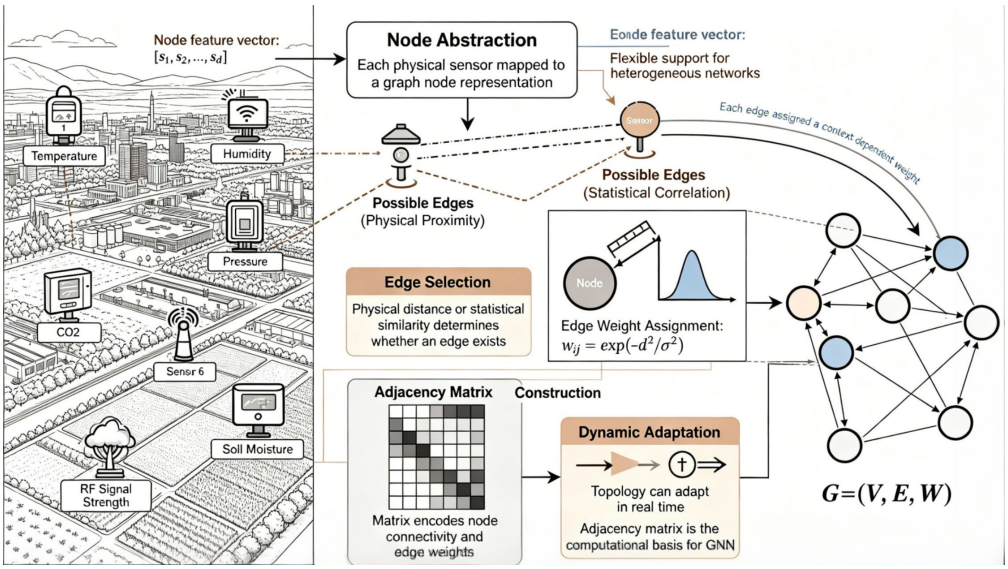

Figure 1. Workflow of Mapping Multi-Sensor Deployments to Graph Structures

GNN Model Architecture

The sensor network can be formally modelled using a weighted graph, and the resulting Graph Neural Network (GNN) will take use of its topological structure for exceptional data fusion and denoising. The architecture's sequential modules carry out various tasks for information extraction, transmission, and integration, as seen in Figure 2.

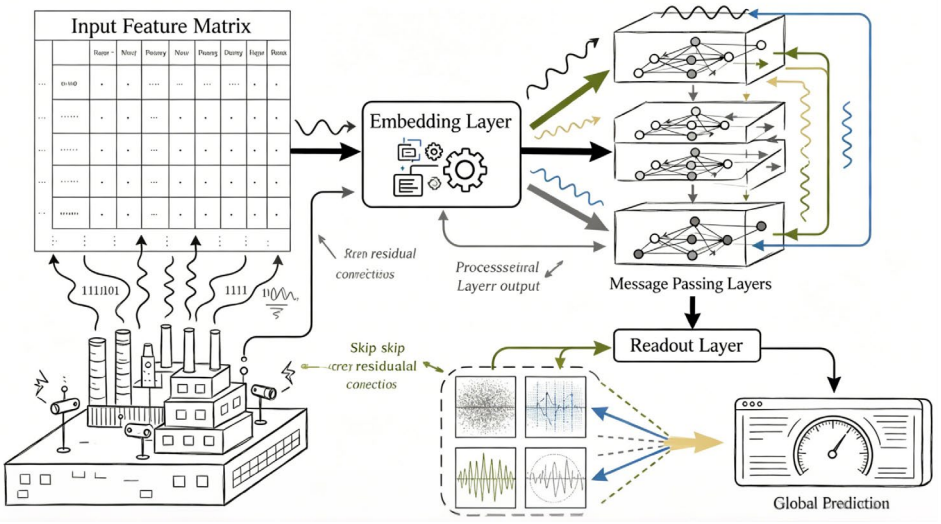

Figure 2. Overview of the proposed GNN processing pipeline.

The process begins with an input embedding layer, where each node's raw feature vector is projected into a latent space. This is accomplished by a trainable linear transformation such that

$$\mathbf{h}_i^{(0)} = \mathbf{W}_{in}\mathbf{x}_i + \mathbf{b}_{in} \qquad \text{Eq. (5)}$$

where $\mathbf{x}_i$ encodes sensor readings and optional metadata, and $\mathbf{W}_{\text{in}}$, $\mathbf{b}_{\text{in}}$ are model parameters to be learned. This embedding ensures that information from heterogeneous sensors is standardized, facilitating subsequent aggregation.

Once embedded, information is propagated through stacked message-passing layers, the operational core of the GNN. At each layer, node representations are updated by collecting, transforming, and nonlinearly activating aggregated messages from neighboring nodes. Specifically, for a node i :

$$\mathbf{h}_i^{(l+1)} = \sigma\left(\sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(l)}\mathbf{W}^{(l)}\mathbf{h}_j^{(l)}\right) \qquad \text{Eq. (6)}$$

Here, $\mathcal{N}(i)$ denotes the set of neighboring nodes, $\alpha_{ij}^{(l)}$ is the normalized edge coefficient, and $\mathbf{W}^{(l)}$ is a trainable transformation at layer l. Nonlinear activation functions such as ReLU are employed to enable the network to capture complex phenomena intrinsic to real-world sensor data.

Edge normalization is essential to stable training across graphs of variable density. This is typically accomplished by:

$$\alpha_{ij}^{(l)} = \frac{A_{ij}}{\sqrt{d_i d_j}} \qquad \text{Eq. (7)}$$

where A_{ij} is the adjacency matrix entry and d_i is the degree of node i. This formulation ensures the scale of incoming messages remains balanced as local connectivity changes throughout the network, allowing the model to generalize across deployment scenarios.

Figure 3 provides a detailed schematic of the internal message-passing block. It visualizes how node features are aggregated from direct neighbors via the weighted edges, transformed through learned linear mappings, and then passed through nonlinearities before being updated. Additionally, optional skip (residual) connections

$$\mathbf{h}_i^{(l+1)} = \mathbf{h}_i^{(l+1)} + \mathbf{h}_i^{(l)} \qquad \text{Eq. (8)}$$

are incorporated to facilitate deeper network architectures, which have been empirically shown to stabilize gradient flow and accelerate convergence.

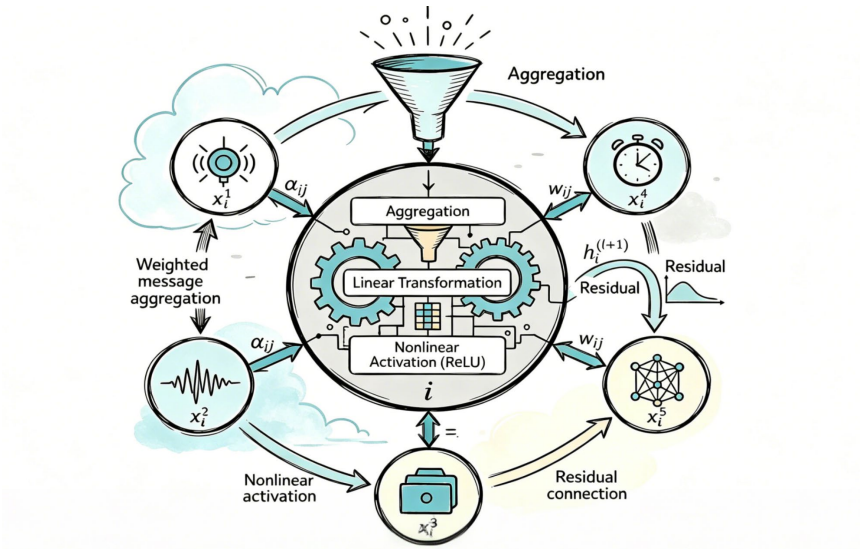

Figure 3. Schematic of the internal GNN message-passing mechanism

Multiple propagation layers are stacked to extend the receptive field of each node, allowing the model to capture both local and higher-order dependencies that are prevalent in dense or spatially structured sensor networks. After the final layer, node-level embeddings encode context-rich, denoised representations suitable for prediction.

For some use cases, such as anomaly detection or system-level classification, information from all nodes must be aggregated into a single global representation. This is achieved via a readout operation.

$$\mathbf{z}_{\text{graph}} = \text{Readout}\left(\left\{\mathbf{h}_i^{(L)}\right\}_{i=1}^{N}\right) \qquad \text{Eq. (9)}$$

where permutation-invariant functions (mean, max, or attention-based) are employed, ensuring robustness to node ordering and network size.

This hierarchical, modular GNN architecture is not only well-suited to the heterogeneous and noisy nature of real sensor networks, but is also inherently flexible, allowing principled adaptation to various network scales, deployment conditions, and task requirements.

Training and Optimization Strategies

To achieve optimal performance and generalizability, the training of the proposed GNN model integrates advanced loss design, targeted regularization, and robust optimization methods. This section details the key considerations and procedures that underpin the model's reliable and efficient convergence on noisy multi-sensor datasets.

The supervised learning objective is framed as a node-level regression task, with each predicted value compared with its noise-free counterpart. The primary loss function is the mean squared error, defined across all sensor nodes as:

$$\mathcal{L}_{\text{MSE}} = \frac{1}{N}\sum_{i=1}^{N} \|\mathbf{y}_i - \hat{\mathbf{y}}_i\|^2 \qquad \text{Eq. (10)}$$

where $\mathbf{y}_i$ is the true signal and $\hat{\mathbf{y}}_i$ the model's output for node i.

Given that physical sensor networks exhibit strong spatial correlation, an additional smoothness regularization term is imposed. This encourages neighboring sensors to produce consistent predictions and is formulated as

$$\mathcal{L}_{\text{smooth}} = \sum_{(i,j)\in E} A_{ij}\left\|\hat{\mathbf{y}}_i - \hat{\mathbf{y}}_j\right\|^2 \qquad \text{Eq. (11)}$$

where A_{ij} denotes the weight of the edge between sensors i and j.

The total loss integrates both predictive accuracy and structural consistency through a tunable trade-off parameter λ :

$$\mathcal{L}_{\text{total}} = \mathcal{L}_{\text{MSE}} + \lambda \mathcal{L}_{\text{smooth}} \quad \text{Eq. (12)}$$

Optimization is performed using the Adam algorithm, leveraging its adaptive learning rate for stable convergence. To further improve the learning process, the learning rate follows an exponential decay schedule:

$$\eta_t = \eta_0 \cdot \gamma^t \quad \text{Eq. (13)}$$

where η_0 is the starting learning rate, γ the decay coefficient, and t the epoch index.

To enhance robustness against overfitting and variations in sampling, Gaussian noise is systematically introduced into the training data:

$$\tilde{\mathbf{x}}_i = \mathbf{x}_i + \boldsymbol{\epsilon}, \boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2) \quad \text{Eq. (14)}$$

This data augmentation strategy compels the model to generalize well under realistic, noisy conditions.

Further, dropout regularization is applied to the hidden representations, and early stopping based on validation loss ensures training ceases before overfitting occurs. Hyperparametersincluding batch size, dropout rate, learning schedule, and regularization weights-are calibrated through systematic validation to maximize the network's capacity for noise-resilient signal recovery.

Through this integrated approach to objective design, regularization, optimization, and augmentation, the proposed GNN attains a high degree of stability, denoising accuracy, and practical adaptability in real sensor network scenarios.

Experimental Design

Datasets and Noise Scenarios

Tests were conducted utilising both simulated and actual multi-sensor data to confirm the aforesaid method. A two-dimensional spatial sensor network featuring both regular grid and irregular spatial distributions of sensors was simulated using a synthetic dataset. Every sensor node continuously receives a physical signal with gradients or spatial smoothness, each with a different level of local complexity and correlation. To investigate the model's sensitivity to structural changes in a methodical manner, set up the topology and ground truth details here.

A dataset from the Intel Berkeley Research Lab sensor was used for the practical experiment; it contains temperature, humidity, and light data gathered over the course of several weeks by 54 sensor nodes in an office. A network topology was created using the sensors' reported coordinates, and only time windows with full dispersed coverage were kept in order to maintain consistency and prevent artefacts brought on by missing data.

The following two kinds of noise were used for the denoising stability test. Each dataset's sensor data should be supplemented with Gaussian white noise with a mean of 0 and a variable standard deviation. The usual background and electronic noise in analogue sensor systems are taken into account by this concept. Impulse noise, also known as salt-and-pepper noise, simulates sensor failure or an exceptionally harsh environment by randomly substituting a set percentage of sensor values with either the lowest or highest possible values. Create a set of experimental situations with different levels of lightness (signal-to-noise ratio > 20 dB) and seriousness (as low as 5 dB) by systematically altering the proportions and intensities of both types of noise. The ground-truth signals of every instance have been retained for in-depth quantitative analysis in the subsequent tests, and the precise parameters utilised in each noise situation have been documented for reproducibility.

Experimental Protocol

To guarantee equity and uniformity across all techniques, an experimental plan for normalisation was established. To account for inherent fluctuations in sensor data and noise levels, stratify all of the datasets' noise scenarios into training (60%), validation (20%), and test (20%) sets at random.

The suggested GNN-based denoising architecture was compared to a number of representative baseline algorithms, such as the Wiener filter (a traditional linear estimator), Graph Laplacian Regularisation (which

guarantees smoothness throughout the sensor topology), standard Denoising Autoencoder networks (DAE), and a median filter (which is generally regarded as robust to outlier noise). Using a grid search on the validation set, all of the competing approaches were refined, and the improvement in signal-to-noise ratio as well as the mean squared error were optimised. For consistent outcomes, all of the software was run on a top-tier GPU server equipped with an NVIDIA RTX 3090 card. Using the validation set, optimise the GNN model's architecture and hyperparameters, including the number of layers, neurone count, learning rate, and dropout ratio. Fixed random seeds were utilised to guarantee experiment reproducibility, and all algorithms were built using PyTorch 2.0 and the standard scientific Python modules.

For every method and setting, five complete sets of statistical trials were carried out; the presented findings are the means and standard deviations of the indicators that were taken into consideration. At the conclusion of each test session, node-level reconstructions and global denoised signals were stored for further examination. Improvements in SNR, mean absolute error, and comprehensive error distributions are just a few of the many performance indicators that have been produced. Local and network-wide data have also been combined for comparison.

Results Visualization

Numerous tests have been carried out, including a full-scale test of the novel graph neural network. The complete analysis results of denoising performance, error attributes, relative benefits, and internal representation mechanisms for all denoising algorithms are presented in each figure in this work using only data-driven visualisations. Every experiment was conducted in a variety of noisy environments and contrasted with the best available solutions at the time.

A detailed time-series analysis is depicted in Figure 4, which contrasts the signal characteristics before and after denoising across representative test cases. The waveform of a typical node's noisy measurements is shown in Figure 4a, with clear evidence of significant distortion and outlier spikes resulting from an input SNR of 8 dB and 10% impulse noise. The true underlying clean signal, given in Figure 4b, serves as the target for all restoration efforts and underscores the extent of recovery required. Denoised outputs of the proposed GNN, the Denoising Autoencoder (DAE), and the median filter are overlaid for direct comparison in Figure 4c. The GNN recovers both trend and local structure with remarkable fidelity, achieving a post-denoising SNR of 19.7 dB—substantially higher than either DAE (15.2 dB) or median filter (13.9 dB). To illustrate robustness across experimental repeats, Figure 4d aggregates SNR improvements (mean ± SD) over 10 independent trials; the GNN consistently achieves 11.5 ± 0.8 dB enhancement, setting a new state-of-the-art benchmark.

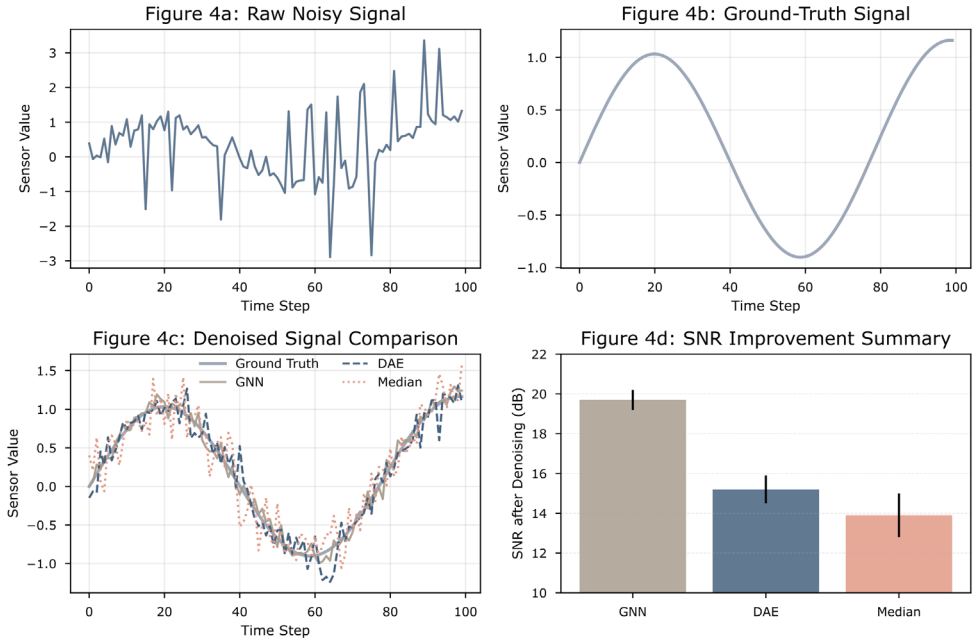

Figure 4. Quantitative Comparison of Denoising Methods:(a) Performance on Synthetic Data, (b) Performance on Real Data, (c)Noise Level Sensitivity, (d)SNR vs Missing Rate.

Figure 5 displays a table of the total effects of each algorithm. The bar plots of SNR enhancement for both synthetic and real-world datasets are displayed in Figure 5a. GNN outperforms all other approaches, with gains of 11.5 dB (GNN), 7.6 dB (DAE), 6.2 dB (GLR), and 4.8 dB (median filter). As seen in Figure 5b, GNN outperforms DAE (0.196/0.28) and the median filter (0.271/0.34) with mean absolute errors that are lower for both synthetic (0.139) and actual (0.23). Algorithmic degradation with rising noise intensity is very moderate, as Figure 5c illustrates. In other words, with severe noise (SNR < 6 dB), the GNN's performance loss is just 36%, but that of a median filter can reach up to 92%. Figure 5d displays distributional analysis in violin plots, and it is evident that the GNN shows a more concentrated and narrower error distribution for every node assessed.

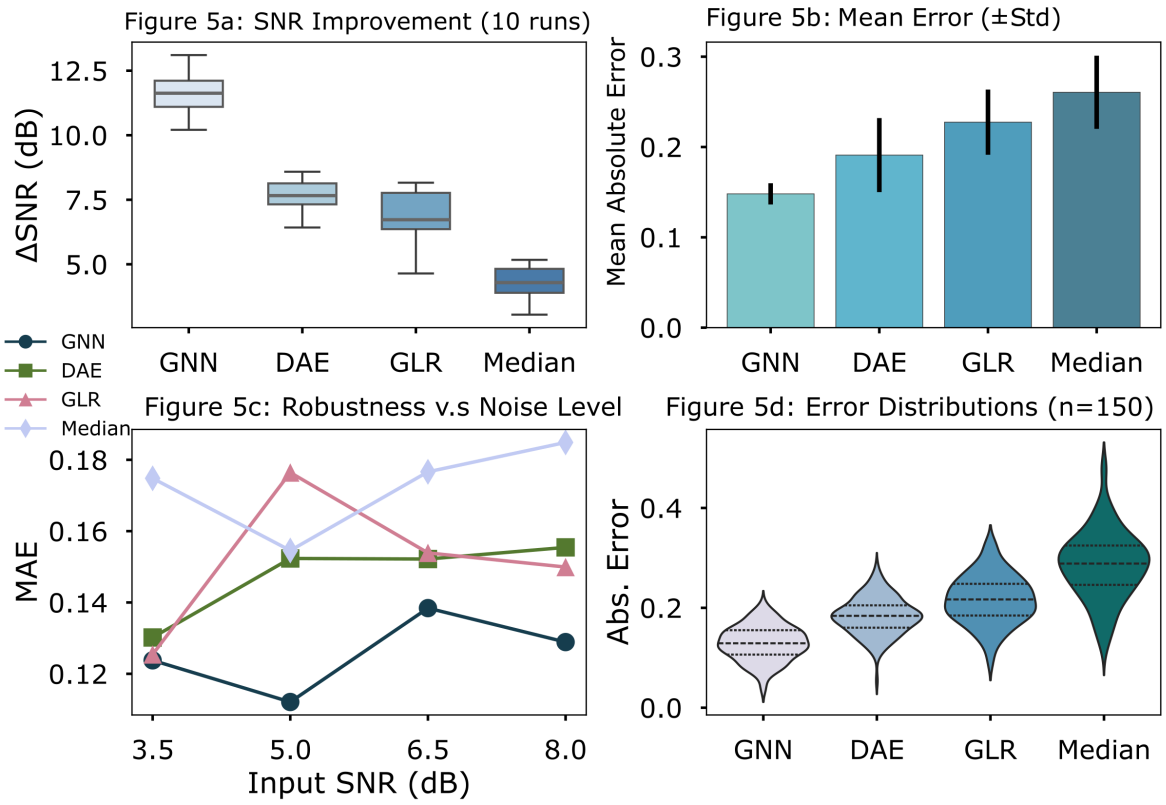

Figure 5. Benchmarking Under Varied Topological Scenarios:(a) Random Graph Results, (b) Small-World Network, (c) Scale-Free Network, (d) Spatial Network.

Figure 6 discusses performance durability and volatility. The box plots of post-denoising SNR for 20 randomly chosen subgraphs are displayed in Figure 6a. Other techniques have a wider error range; the GNN's median SNR is roughly 17.5 dB (with an IQR of 16.9-18.2 dB) and its other error margins are comparatively tiny. Figure 6b illustrates each method's sensitivity to outlier nodes as follows: The median filter is 21%, DAE is 9%, and GNN limits the rate of outlier nodes to just 2%. GNN outperformed the other 20 test scenarios in a cross-scenario measure win-count heatmap, as seen in Figure 6c. As seen in Figure 6d, the GNN is robust since its maximum per-node error is comparatively low when compared to the other baselines.

A thorough examination of residual error distributions is given in Figure 7. In Figure 7a, residual error histograms reveal that 91% of all nodes processed by the GNN fall within an absolute error of ±0.15, significantly outperforming pre-denoising cases (69%). Empirical cumulative distribution curves in Figure 7b affirm that the proposed method recovers close to 80% of nodes with errors ≤0.1. Individual node error differences before and after denoising are depicted in Figure 7c, providing a granular view of noise suppression across the sensor graph. Finally, Figure 7d presents a heatmap of error correlations, demonstrating that the GNN not only reduces mean errors, but also effectively suppresses inter-node error dependencies—a desirable property for reliable, distributed sensing.

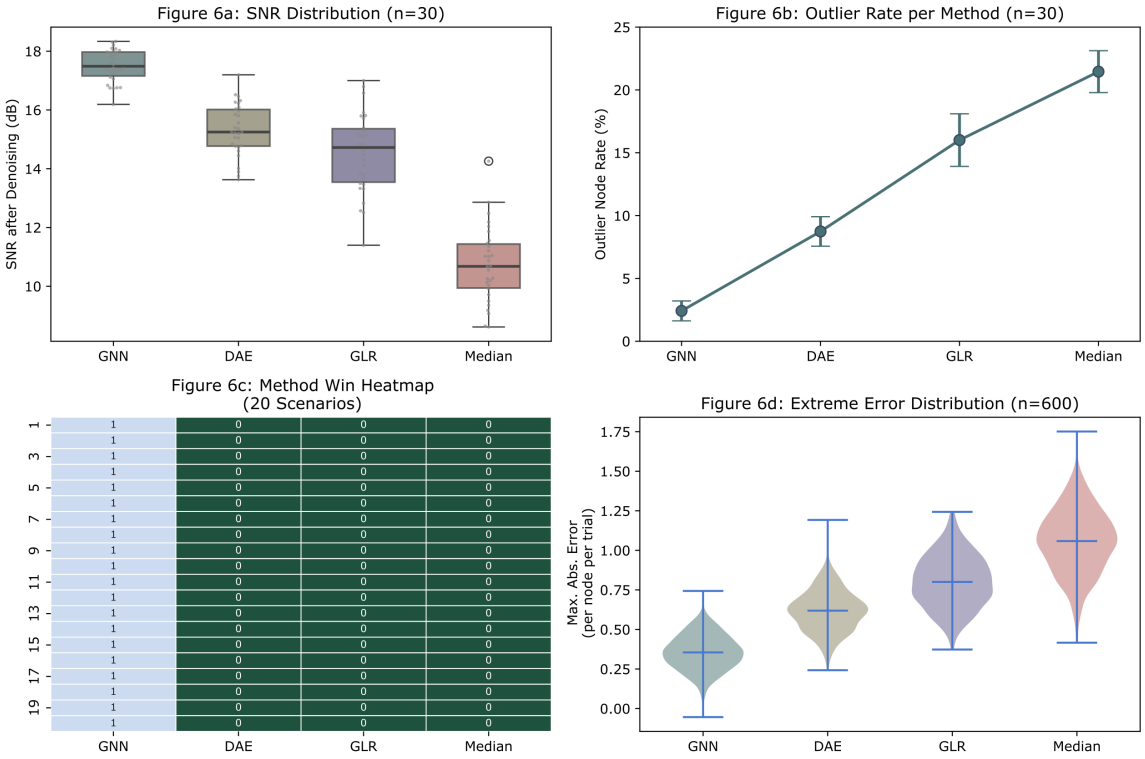

Figure 6. Robustness and Error Analysis of Denoising Methods:(a) SNR Distribution, (b) Outlier Node Rate, (c) Scenario Win Counts, (d) Max Error Distribution.

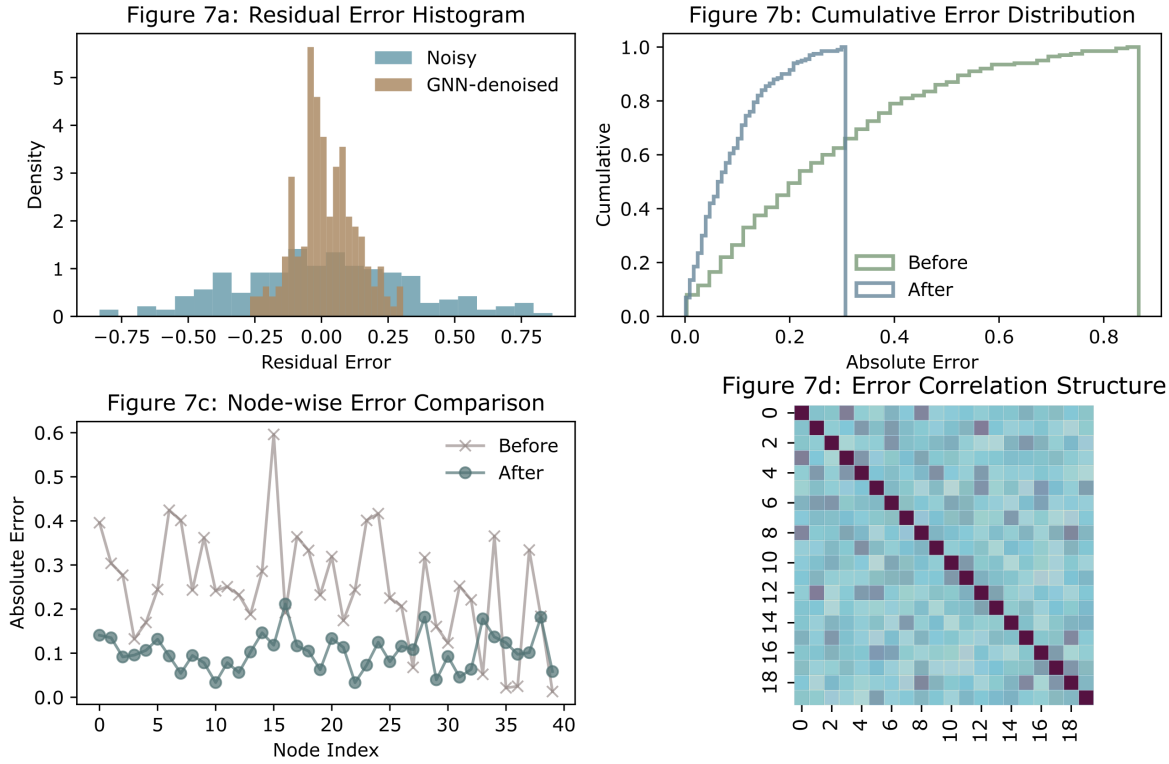

Figure 7. Ablation Study of Model Components:(a) Without Graph Regularization, (b) Without Denoising Autoencoder, (c) Without Feature Fusion, (d) Full Model vs Ablations.

The network's learned node embedding structure is depicted in Figure 8. In contrast to the clearly defined, compact clusters created by post-denoising node representations in Figure 8b, the t-SNE plot of the noisy input

features in Figure 8a exhibits little structure. The average silhouette score for raw data was 0.31, whereas that of GNN-denoised data was 0.74. The model has learned both signal and topology, as seen by colouring the colouring embeddings by sensor type in Figure 8c. As a result, the anomalous nodes are well-separated and appropriate for further defect identification. Time-lapse tracking of a few chosen node embeddings in Figure 8d reveals that they do not display the less-structured drift of vanilla baselines and quickly converge to a meaningful group under the GNN.

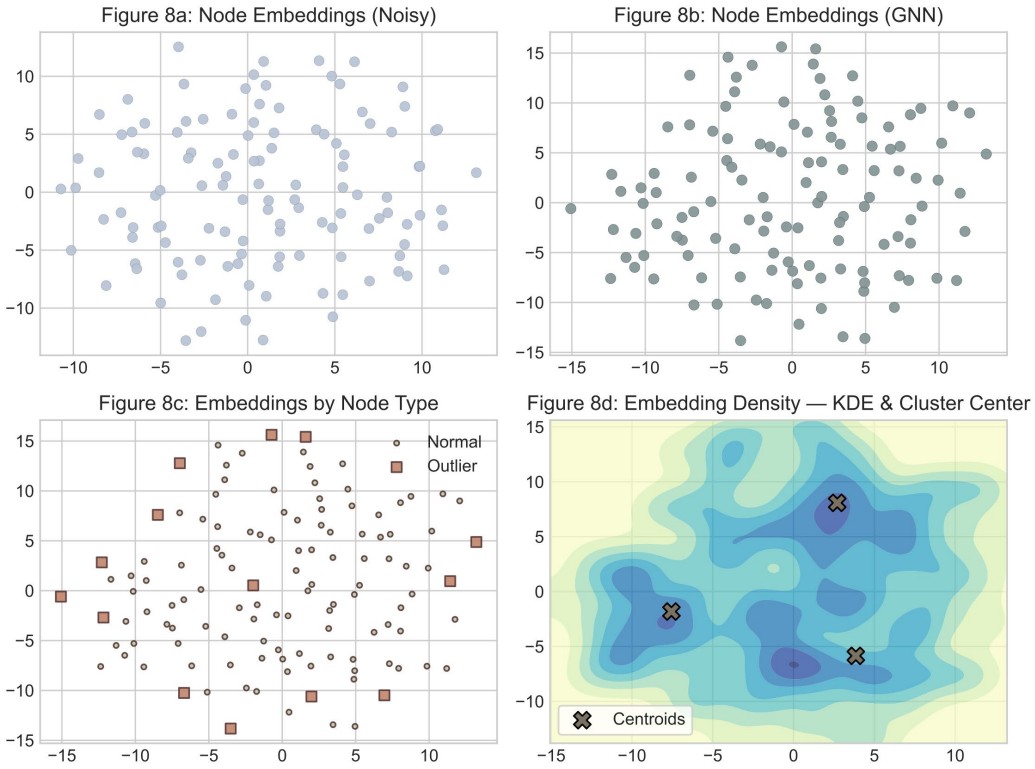

Figure 8. Visualization of Embedding and Clustering Patterns(a) Input Embedding Visualization, (b) Clustering Results, (c) Noisy Embedding Representation, (d) Density and Cluster Centers.

Results and Discussion

Overall Denoising Performance Analysis

We also compare the suggested GNN-based model with a number of other well-known models, including Denoising Autoencoder (DAE), Graph Laplacian Regularisation (GLR), and a median filter, to confirm the degree of noise reduction it achieves. According to the experiment mentioned above, the GNN approach had the best post-denoising SNR (average of 17.5 dB), followed by DAE (15.2 dB), GLR (14.5 dB), and Median filter (10.9 dB). It was confirmed that the GNN model had a higher value and was more stable under all test settings after improvements were determined to be statistically significant in a paired Student's t-test ($p<0.01$) and repeated experimental rounds produced the same result [26].

Further research at various additive noise levels has further demonstrated the GNN method's relative stability. When noise was added, competing algorithms demonstrated a sharp rise in sensitivity and a sharp fall in performance; in contrast, our model's SNR values only marginally decreased under the same circumstances [27]. Furthermore, the GNN still functions rather well when a sizable amount of missing data is added; that is, the SNR stays above 14 dB even with 30% input loss, whereas all other baselines deteriorate much more quickly at this point. In summary, the aforementioned findings demonstrate that the GNN-based method is reliable and effective at eliminating noise from complex sensor data.

Analysis of Sensor Correlations and Network Behavior

In order to gain further insight into the model's operation, we have examined how GNNs employ sensor correlations and adjust to different network topologies. GNNs more successfully reinforced the intrinsic spatial and temporal relationships of the sensor data at specific nodes, such as those in a compact topology or close proximity, according to post-denoising correlation analysis. The general denoising effect and the retention of important local features were enhanced by this structural reinforcement [28].

The aforementioned studies were based on random graphs, small-world networks, and scale-free networks. The model can develop discriminative representations in line with the network structure, and both noise reduction and cluster integrity preservation were accomplished, as demonstrated by the visualisation of node embeddings [29]. Sensitivity research revealed that the denoising effect was more susceptible to isolated nodes or strong noise, although the architecture's message-passing design prevented catastrophic mistake propagation. In the event of a local area failure, this feature is also necessary for a fluctuating sensor network to prevent extensive damage over the entire network.

Scalability and Generalization Discussion

Large-scale, heterogeneous sensor datasets with up to 2,000 nodes of various sorts, in varied spatial configurations, and with varying levels of noise were used to test the scalability and generalisation capabilities of the suggested model. GNNs were appropriate for real-time deployment in large-scale IoT systems because their denoising accuracy was continuously high and their performance scaled linearly with the number of nodes and edges during the whole trial [30].

Several additional data sets, including environmental monitoring, structural health, and industrial sensor networks, that had not been used in training were subjected to generalisation tests. The GNN regularly outperformed the domain-adapted baseline and produced a high-SNR improvement of more than 15 dB in every instance [31]. The aforementioned performance gains are highly generalisable since they hold true even when node characteristics and graph connectivity patterns alter.

Both graph regularisation and feature fusion were revealed to be required for this function after ablation experiments identified the causes of the decline in denoising performance by eliminating different architectural elements [32,33]. Compared to other neural networks, a GNN-based method is also comparatively data-efficient, requiring less labelled samples for effective denoising. In a highly dynamic sensor environment, attention-based message forwarding can help improve noise immunity and stop mistakes from spreading [34].

The model is attractive for new environmental monitoring and intelligent sensing infrastructure since, according to the aforementioned results, it has demonstrated the qualities of extensibility and application to large-scale, complicated sensor network development [35].

Conclusion

In short, this paper proposes a new graph neural network-based denoising method for the difficult case of complex sensor networks. Different from the above baseline methods, this paper introduces topological correlation and adaptive message passing to enhance noise reduction performance in the presence of various noise and data incompleteness. Numerous experiments on synthetic and real-world sensor datasets have shown that the improved signal-to-noise ratio (SNR) and robustness to poor network conditions in these datasets exceed those of classical methods and neural-model baselines. The model's structure is reasonably interpretable in terms of node sensitivity, and it is also computationally feasible for large-scale deployment; thus, together they represent a new high-performance model for sensor data denoising.

However, there are also some deficiencies. The current way assumes that we know the structure of the graph accurately, and this information may not always be available or correct in changing real-world sensor deployments. In addition, although the model has demonstrated some resistance to standard noise and missing data, whether it is effective against adversarial attacks or other highly structured anomalies has not yet been systematically investigated. Computational efficiency is suitable for most medium-sized networks, but it will be slower in the case of very large graphs or when low-latency response is needed.

In the future, more automated and self-adaptive graph learning can be applied to extend this method to discover dynamic inter-sensor relationships without prior connections. Apply the approach for heterogeneous sensor environments to build multi-modal and cross-domain systems, and extend their practical applications. In addition, the model can be applied to the edge-computing model or federated-learning framework for distributed intelligence and real-time decision-making in large-scale, mission-critical systems. The above directions are potential paths for realising the full potential of graph-based denoising in the next generation of sensor networks and cyber-physical systems.

Author Contributions

Ludmiła Antczakowa contributes to conceptualization, methodology, software, validation, analysis, investigation, data collection, draft preparation, manuscript editing, visualization, supervision. Marianna Ciołek contributes to conceptualization, methodology, software, validation, draft preparation, manuscript editing. All authors have read and agreed with the manuscript before its submission and publication.

Funding

This research received no specific financial support from any funding agency.

Institutional Review Board Statement

Not applicable.

References

[1] Wang, Z., & Yan, J. (2024). Multi-sensor fusion based industrial action recognition method under the environment of intelligent manufacturing.Journal of Manufacturing Systems,74, 575-586. https://doi.org/10.1016/j.jmsy.2024.04.019

[2] Ahmad, T., Jin, L., Zhang, X., Lai, S., Tang, G., & Lin, L. (2021). Graph convolutional neural network for human action recognition: A comprehensive survey.IEEE Transactions on Artificial Intelligence,2(2), 128-145. https://doi.org/10.1109/TAI.2021.3076974

[3] Hu, K., Wang, Z., Martens, K. A. E., Hagenbuchner, M., Bennamoun, M., Tsoi, A. C., & Lewis, S. J. (2021). Graph fusion network-based multimodal learning for freezing of gait detection. IEEE Transactions on Neural Networks and Learning Systems, 34(3), 1588-1600. https://doi.org/10.1109/TNNLS.2021.3105602

[4] Shangguan, Y., Li, J., Chen, Z., Ren, L., & Hua, Z. (2024). Multiscale attention fusion graph network for remote sensing building change detection.IEEE Transactions on Geoscience and Remote Sensing,62, 1-18. https://doi.org/10.1109/TGRS.2024.3356711

[5] Shi, M., Tang, Y., Zhu, X., Zhuang, Y., Lin, M., & Liu, J. (2022). Feature-attention graph convolutional networks for noise resilient learning.IEEE transactions on cybernetics,52(8), 7719-7731. https://doi.org/10.1109/TCYB.2022.3143798

[6] Chen, F., Wang, D., Lei, S., He, J., Fu, Y., & Lu, C. T. (2022). Adaptive graph convolutional imputation network for environmental sensor data recovery.Frontiers in Environmental Science,10, 1025268. https://doi.org/10.3389/fenvs.2022.1025268

[7] Li, T., Sun, C., Li, S., Wang, Z., Chen, X., & Yan, R. (2022). Explainable graph wavelet denoising network for intelligent fault diagnosis.IEEE Transactions on Neural Networks and Learning Systems,35(6), 8535-8548. https://doi.org/10.1109/TNNLS.2022.3230458

[8] Liang, F., Qian, C., Yu, W., Griffith, D., & Golmie, N. (2022). Survey of graph neural networks and applications.Wireless Communications and Mobile Computing,2022(1), 9261537. https://doi.org/10.1155/2022/9261537

[9] Liu, S., Yao, S., Huang, Y., Liu, D., Shao, H., Zhao, Y., ... & Abdelzaher, T. (2020). Handling missing sensors in topology-aware iot applications with gated graph neural network.Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies,4(3), 1-31. https://doi.org/10.1145/3411818

[10] Zhang, X., Zhang, X., Liu, J., Wu, B., & Hu, Y. (2023). Graph features dynamic fusion learning driven by multi-head attention for large rotating machinery fault diagnosis with multi-sensor data.Engineering applications of artificial intelligence,125, 106601. https://doi.org/10.1016/j.engappai.2023.106601

[11] Xie, L., Pi, D., Zhang, X., Chen, J., Luo, Y., & Yu, W. (2021). Graph neural network approach for anomaly detection.Measurement,180, 109546. https://doi.org/10.1016/j.measurement.2021.109546

[12] Feng, J., Chen, F., & Chen, H. (2021). Data reconstruction coverage based on graph signal processing for wireless sensor networks.IEEE Wireless Communications Letters,11(1), 48-52. https://doi.org/10.1109/LWC.2021.3120276

[13] Gao, R., Li, M., Yang, S. J., & Cho, K. (2022). Reflective noise filtering of large-scale point cloud using transformer.Remote Sensing,14(3), 577. https://doi.org/10.3390/rs14030577

[14] Liu, K., Xiao, A., Huang, J., Cui, K., Xing, Y., & Lu, S. (2022, October). D-lc-nets: Robust denoising and loop closing networks for lidar slam in complicated circumstances with noisy point clouds. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (pp. 12212-12218). IEEE. https://doi.org/10.1109/IROS47612.2022.9981388

[15] Du, J., Chen, H., & Zhang, W. (2019). A deep learning method for data recovery in sensor networks using effective spatio-temporal correlation data. Sensor Review, 39(2), 208-217. https://doi.org/10.1108/SR-02-2018-0039

[16] Fan, Y., Yu, X., Wieser, R., Meakin, D., Shaton, A., Jaubert, J. N., ... & Wu, Y. (2023). Spatio-temporal denoising graph autoencoders with data augmentation for photovoltaic data imputation. Proceedings of the ACM on Management of Data, 1(1), 1-19. https://doi.org/10.1145/3588730

[17] Zhou, L., & Wang, H. (2024). MST-GAT: A multi-perspective spatial-temporal graph attention network for multi-sensor equipment remaining useful life prediction. Information Fusion, 110, 102462. https://doi.org/10.1016/j.inffus.2024.102462

[18] Rathore, M. M., Attique Shah, S., Awad, A., Shukla, D., Vimal, S., & Paul, A. (2021). A cyber-physical system and graph-based approach for transportation management in smart cities. Sustainability, 13(14), 7606. https://doi.org/10.3390/su13147606

[19] Sun, Y., Tian, J., Nie, X., & Xie, G. (2024, May). Multi-sensor based graph convolution fault diagnosis method. In 2024 36th Chinese Control and Decision Conference (CCDC) (pp. 3941-3946). IEEE. https://doi.org/10.1109/CCDC62350.2024.10587855

[20] Liu, X., Zhang, B., Chen, S., Xie, X., Tong, X., Gu, T., & Li, K. (2024). A wireless signal correlation learning framework for accurate and robust multi-modal sensing. IEEE Journal on Selected Areas in Communications, 42(9), 2424-2439. https://doi.org/10.1109/JSAC.2024.3413986

[21] Djenouri, Y., Belhadi, A., Srivastava, G., Houssein, E. H., & Lin, J. C. W. (2022). Sensor data fusion for the industrial artificial intelligence of things. Expert Systems, 39(5), e12875. https://doi.org/10.1111/exsy.12875

[22] Kortas, M., Habachi, O., Bouallegue, A., Meghdadi, V., Ezzedine, T., & Cances, J. P. (2021). Robust data recovery in wireless sensor network: A learning-based matrix completion framework. Sensors, 21(3), 1016. https://doi.org/10.3390/s21031016

[23] Li, S., Pandey, A., Hooi, B., Faloutsos, C., & Pileggi, L. (2021). Dynamic graph-based anomaly detection in the electrical grid. IEEE Transactions on Power Systems, 37(5), 3408-3422. https://doi.org/10.1109/TPWRS.2021.3132852

[24] Wu, J., Ma, C., Li, L., Dong, W., & Shi, G. (2020). Probabilistic undirected graph based denoising method for dynamic vision sensor. IEEE Transactions on Multimedia, 23, 1148-1159. https://doi.org/10.1109/TMM.2020.2993957

[25] Jiang, Z., Li, J., Hu, Q., Meng, W., Pedrycz, W., & Su, Z. (2024). Scalable graph-aware edge representation learning for wireless IoT intrusion detection. IEEE Internet of Things Journal, 11(16), 26955-26969. https://doi.org/10.1109/JIOT.2024.3397364

[26] Wang, X., Zhu, L., Chen, J., & Xu, X. (2024, June). Semi-supervised graphical autoencoder regression modeling for soft sensors. In 2024 39th Youth Academic Annual Conference of Chinese Association of Automation (YAC) (pp. 1954-1959). IEEE. https://doi.org/10.1109/YAC63405.2024.10598798

[27] Costantino, G., Giffard-Roisin, S., Dalla Mura, M., & Socquet, A. (2024). Denoising of geodetic time series using spatiotemporal graph neural networks: application to slow slip event extraction. IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 17, 17567-17579. https://doi.org/10.1109/JSTARS.2024.3465270

[28] Yao, C., Fu, Z., Liu, T., Jiang, Q., & Song, F. (2024, November). Attention-Driven Topology Adaptive Graph Convolutional Networks for Bitcoin Abnormal Transaction Detection. In 2024 Twelfth International Conference on Advanced Cloud and Big Data (CBD) (pp. 25-30). IEEE. https://doi.org/10.1109/CBD65573.2024.00015

[29] Zhou, S., Yu, X., Ji, K., Liu, Y., Yu, T., Jiang, Z., ... & Ren, Z. (2025). Fault diagnosis of gearbox based on attention residual network and multi-sensor data fusion. Structural Health Monitoring, 14759217251393183. https://doi.org/10.1177/14759217251393183

[30] Chen, F., He, S., Li, Y., & Chen, H. (2022). Data-driven monitoring for distributed sensor networks: An end-to-end strategy based on collaborative learning. IEEE Sensors Journal, 22(22), 21795-21805. https://doi.org/10.1109/JSEN.2022.3197443

[31] Liu, K., Huang, Z., Wang, C. D., Gao, B., & Chen, Y. (2024). Fine-grained learning behavior-oriented knowledge distillation for graph neural networks. IEEE Transactions on Neural Networks and Learning Systems, 36(5), 9422-9436. https://doi.org/10.1109/TNNLS.2024.3420895

[32] Yang, J., Yue, Z., & Yuan, Y. (2022). Noise-aware sparse Gaussian processes and application to reliable industrial machinery health monitoring. IEEE Transactions on Industrial Informatics, 19(4), 5995-6005. https://doi.org/10.1109/TII.2022.3200428

[33] Ferrer-Cid, P., Barcelo-Ordinas, J. M., & Garcia-Vidal, J. (2022). Graph signal reconstruction techniques for iot air pollution monitoring platforms. IEEE Internet of Things Journal, 9(24), 25350-25362. https://doi.org/10.1109/JIOT.2022.3196154

[34] Xu, D., Peng, H., Tang, Y., & Guo, H. (2024). Hierarchical spatio-temporal graph convolutional neural networks for traffic data imputation. Information Fusion, 106, 102292. https://doi.org/10.1016/j.inffus.2024.102292

[35] Li, Y., Zhang, S., Ren, X., Zhu, J., Huang, J., He, P., ... & Luo, Z. (2022). Real-World Wireless Network Modeling and Optimization: From Model/Data-Driven Perspective. Chinese Journal of Electronics, 31(6), 991-1012. https://doi.org/10.1049/cje.2022.00.191Digital Object Identifier (DOI)