Expert Systems With Applications 291 (2025) 128540

Contents lists available at ScienceDirect

Expert Systems With Applications

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

A robust machinery action recognition in construction using a mixed graph convolution block

Shuozhi Wang [a,b], Yitian Zhao [c], Zichao Zhang [b], Yifan Zhao [b,*]

[a] School of Aerospace Engineering, Xiamen University, Xiamen, China
[b] School of Aerospace, Transport and Manufacturing, Cranfield University, Bedfordshire, MK43 0AL, UK
[c] Ningbo Institute of Materials Technology and Engineering, Chinese Academy of Sciences, Ningbo, China

ARTICLE INFO

Keywords:
Activity classification
Computer vision
Graph convolutional networks
Keypoint extraction

ABSTRACT

Existing computer vision-based approaches struggle to identify machinery actions due to the challenges posed by environmental complexity and various obstructions in construction. This study introduces a novel two-stage framework that benefits from a newly proposed Residual Fusion Graph Convolution Network (RFGCN) to classify machinery actions with enhanced robustness and accuracy. The framework first extracts key machinery components from video data, subsequently transforming them into a graph-based representation. This spatio-temporal graph is then fed into the RFGCN model, specifically designed to overcome issues like partial obstructions and missing information common in busy construction sites. Experimental evaluations reveal the method's high efficacy, achieving an accuracy of up to 96.4 % and outperforming state-of-the-art. Additionally, the proposed RFGCN model achieved state-of-the-art performance on four established benchmark datasets for graph classification using spatial data only. These results suggest the potential of the proposed framework in facilitating the transition towards more intelligent and automated construction sites.

1. Introduction

Pose estimation and action recognition are critical tasks in numerous applications, including human-computer interaction (Huo et al., 2023), security and surveillance (Barra et al., 2018; Cormier et al., 2022), and industrial automation. This importance is accentuated in industrial contexts, where precise pose estimation of machinery is essential for tasks such as anomaly detection, preventive maintenance (Chen et al., 2022; Rao et al., 2022; Rasul et al., 2021), and enhancement of operational efficiency (Kassem et al., 2021). The advent of the digital twin concept, a virtual replica of the physical system, further underscores the necessity for accurate machine action/activity recognition, paving the way for predictive analysis and system optimisation (Broo et al., 2022; Chiachío et al., 2022; Jiang et al., 2022; Pregnolato et al., 2022).

Construction sites are dynamic environments that demand reliable and robust monitoring systems for tracking and predicting a multitude of activities, ensuring safety, timely project completion, cost-effectiveness, and overall operational efficiency (Geimer, 2020). A cornerstone of these systems is machinery activity classification - accurate and prompt classification is paramount for effective construction management (Lingard et al., 2013). Classifing activities based on video feeds or images from busy construction sites can be challenging, especially when it comes to detecting small objects (Kim et al., 2023; Zhang et al., 2023). There is a considerable distance between the camera and its targets in typical camera setups on large construction sites. As a result, machinery components, which are crucial for determining the type of activity taking place, often occupy only a tiny fraction of the overall image. Factors such as varying lighting conditions, weather, obstructions, and the inherent small size of these components in relation to the entire scene make this recognition task particularly challenging. The precision in detecting these parts directly affects the accuracy of activity classification. A misstep here could lead to the misidentification of operations, thereby affecting the overall misinterpretation of construction processes. Properly classifying these activities is also crucial for providing immediate response to potential safety risks and plays a pivotal role in enhancing planning, resource distribution, and operational insights.

The complexity of construction sites, including their interrelated activities and resources, can be modelled as a graph (Jia et al., 2023). In this model, nodes represent entities such as machinery or tasks, and edges denote relationships or interactions between them. To capture both local and global patterns within this graph during a construction process, we turn to Graph Convolutional Networks (GCNs). GCNs, a class

* Corresponding author.
E-mail addresses: Shuozhi.Wang@cranfield.ac.uk (S. Wang), yitian.zhao@nimte.ac.cn (Y. Zhao), Zichao.zhang.960@cranfield.ac.uk (Z. Zhang), yifan.zhao@cranfield.ac.uk (Y. Zhao).

https://doi.org/10.1016/j.eswa.2025.128540
Received 4 February 2024; Received in revised form 23 April 2025; Accepted 6 June 2025
Available online 10 June 2025
0957-4174/© 2025 The Author(s). Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

of deep learning algorithms specifically designed for graph data, provide an effective means to enhance the interpretation and prediction of activities on construction sites. Although Convolutional Neural Networks (CNNs) have been widely employed for image-based classification tasks, they have limitations in dealing with structured or relational data (Ren et al., 2022). CNNs primarily cater to grid-like data, such as images, but struggle to handle irregular structures or relational patterns directly. GCNs, on the other hand, are specifically designed for graph-structured data. The complex network of interrelationships at a construction site - between machinery, personnel, and tasks - can be naturally represented as a graph, allowing GCNs to capture relational patterns between entities. This capability is beneficial for tasks like action recognition, where the context and relation between entities can play a crucial role. GCNs also provide permutation invariance, i.e., the order of nodes in the input does not affect their performance, a significant advantage in dynamic scenarios like construction sites where the activity order may vary.

While GCNs have distinct advantages in handling structured data, applying these networks directly based on image information sometimes cannot fully exploit their potential for action categorisation tasks. Consequently, prior to deploying GCNs for these tasks, an intermediate feature extraction step to represent the action (e.g., pose or keypoint extraction) is required (Kim et al., 2023). However, a prevalent challenge is the incomplete pose information of machinery and equipment, which is predominantly caused by occlusions and relatively low-quality image quality. Such variations and obstructions can lead to obscured or missing parts, resulting in a scarcity of critical data for classification. The dynamic nature of construction sites, coupled with the multitude of interacting elements, amplifies the occurrence of these issues, necessitating the development of robust methodologies for effective action classification under such constraints.

In this paper, we introduce a novel video-based methodology integrating object detection with a new Graph Convolutional Networks (GCNs) to tackle the challenges of activity classification at construction sites. Different from the existing methods that rely on model-based keypoints extraction (complete pose information), a Residual Fusion Graph Convolution Networks (RFGCN) based on graph attention and the residual connection is presented to tackle the challenge of incomplete pose data of machinery. Another innovative aspect is that, unlike existing methods that often only focus on single machinery, primarily excavators, the proposed solution integrates other cooperating machinery into the introduced spatio-temporal mixed graphs to achieve more precise activity recognition. Our contributions to the field are fourfold:

- We propose a novel framework merging the strengths of small object detection and RFGCN. This fusion enables the effective handling of the intricate dynamics of construction sites, effectively managing the spatio-temporal graph-structured data inherent in these environments. The graph in the spatial domain is not only includes nodes extracted from various components of a single machinery, but also nodes from cooperating machinery working on the same construction task.
- The proposed methodology exhibits robust performance in scenarios of missing data, a critical aspect often overlooked in existing approaches. This capability significantly enhances the reliability and practical applicability of our approach in real-world, imperfect data conditions.
- Through rigorous experimentation and evaluation, we demonstrate that the proposed method exhibits superior performance in terms of accuracy, and resilience to missing data, against state-of-the-art.
- The proposed RFGCN is tested on four public datasets, including those in bioinformatics and social networks, underlining its versatility and potential for adaptation in various graph classification tasks.

2. Related work

2.1. Image-based methods

The quest to investigate excavator action estimation and classification began with image-based methodologies, predominantly focusing on pose estimation. Lundeen et al. (2016) laid the foundation by demonstrating the applicability of marker-based sensor technology. To improve upon this and eliminate the need for physical markers, Tang et al. (2023) advanced the field with a non-invasive onboard visual-inertial sensor system. This system aimed to combine data from various sensors, leading to enhanced accuracy. However, one primary concern was the discernibility of excavator components, especially in real-world scenarios where cameras were positioned at a long distance. This distance often compromised the clarity of the excavator components (Luo et al., 2018, 2019). Addressing this gap, Meng et al. (2020) proposed to employ UAVs coupled with YOLOv3 (Redmon & Farhadi, 2018), targeting distant action recognition. Yet, with the increasing need for computational efficiency in on-site applications, Guo et al. (2022) introduced a lightweight fully convolutional network (FCN), striving for a balance between accuracy and computational demands. More recently, Shin et al. (2025) proposed a 3D ResNet-based classifier enhanced with SHAP interpretability analysis and tailored image preprocessing, demonstrating improved excavator activity recognition performance under far-field camera conditions. Additionally, Pham and Han (2024) proposed a generative model to synthesise realistic excavator images with varied backgrounds, enabling improved 3D pose estimation without manual annotation and enhancing the generalisation of models trained in synthetic data.

2.2. Video-based methods

While image-based methods provided snapshots of actions, the dynamic nature of construction activities necessitated a temporal perspective, leading to the exploration of video-based methods. Initial forays into this domain were grounded in non-learning-based methods like video interpretation techniques (Gong & Caldas, 2010) and Histogram of Oriented Gradients (HOG) (Golparvar-Fard et al., 2013). However, the advent of deep learning provided tools to capture the richness of video data more effectively. Roberts and Golparvar-Fard (2019) capitalised on this by employing CNNs for a more nuanced detection and tracking of machinery. Recognising the dual need to understand both the visual and temporal aspects, Kim and Chi (2019) developed with a hybrid model, merging CNN and Long Short-Term Memory (Hochreiter & Schmidhuber, 1997) (LSTM). This combination was pivotal in modeling both the spatial features and the sequence of activities. As the field matured, researchers including Luo et al. (2021) explored more sophisticated temporal models like Gated Recurrent Units (GRUs) and Recurrent Neural Network (RNN), aiming to predict future equipment poses and safeguard site workers. While these methods advanced the field significantly, the challenge of recognising new or rare activities without extensive training data persisted, leading to the exploration of innovative techniques like zero-shot learning (Chen et al., 2023).

2.3. Graph-based approaches

With the increasing complexity of construction sites and the intricate interplay of entities, researchers began to look beyond traditional image and video data. This quest for a more holistic understanding led to the exploration of graph data structures. Graphs, by their very nature, excel in capturing relationships, making them ideal for representing construction sites with their myriad of interdependent entities. Jia et al. (2023) established a framework for leveraging graphs in construction. This methodology found applications in diverse areas, from crack detection to automating floor plan evaluations. A notable advancement was made by Kim and Chi (2022), who harnessed the power of GNNs

to unravel complex visual relationships on construction sites. Recognising the potential of an integrated approach, Wang et al. (2024) merged graph data with computer vision, offering a more comprehensive perspective to understand construction activities. However, the application of Graph Neural Network (GNN) in specific areas like excavator action classification needs further exploration, highlighting potential avenues for future research.

3. Methodology

In Fig. 1, we present a two-stage, video-based data analysis framework specifically designed to reliably and robustly classify construction machinery actions. The overarching aim is to enhance the practicability of AI for the construction sector by addressing challenges encountered during real-world deployments. The method consists of state-of-the-art object detection and graph neural network techniques, leveraging the spatio-temporal information present in video data. By exploring the spatial data correlations among collaborating machinery, our framework provides a precise depiction of their collective dynamics. Moreover, this framework aims to maintain a high classification accuracy, even for scenarios with missing data. This is a common occurrence in real-world deployments due to challenges such as occlusions, which can lead to the absence of keypoints crucial for machinery pose estimation. To tackle this challenge, we propose a novel graph classification model called Residual Fusion Graph Convolution Networks (RFGCN). Different from pose estimation methods, such as the paradigm in KAPAO research (McNally et al., 2021), the proposed method focuses on classification based on relatively simple pose-related features, even in the presence of missing keypoints.

3.1. Data processing

In support of the solution development, we created a mini-construction site within our lab to simulate five types of earthmoving activities during the construction process: idling, digging, lifting, swinging and loading. The involved machinery includes an excavator, a dumper, and a dozer. A dedicated dataset was collected at a frame rate of 10 frames per second (FPS) and a resolution of 3840×2160. The dataset consists of 15 classes of objects, including 3 types of machinery and 12 types of key components, all illustrated in Fig. 2. The selection of these components is mainly based on several literature on excavator pose estimation (Assadzadeh et al., 2022; Guo et al., 2022; Tang et al., 2023). A total of 17 objects are defined to form the 15 classes of nodes. The objects (except the whole machinery) have a pixel range from 0.1 % to 1 % of the original image size. Object detection was performed using YOLOv8 (Reis et al., 2023), known for its exceptional performance in small object detection in terms of both speed and accuracy. The dataset for object detection initially comprised 124 manually labelled samples, which were expanded to 372 through augmentation by adjusting brightness and flipping. A standard 80-20 split was employed for training and testing purposes.

After obtaining the bounding boxes for the 15 classes of objects from a single image, each bounding box is converted into a point using the center of the objects, which results in a spatial graph. Fig. 3 shows two examples of object detection results and their subsequent transformation into a spatial graph structure overlaid on the raw image. The top example illustrates a scenario where all key components were successfully detected, while the bottom example depicts a scenario with missing key components. It is worth noting that graph data offers notable advantages, particularly in its ability to effectively represent spatial relationships between objects. This spatial graph can proficiently describe the posture of a specific machinery and also represent the non-Euclidean spatial relationships among multiple cooperative plant (depicted by black solid curves in the right pictures of Fig. 3). This approach differs from certain model-based posture estimation techniques, which often aim to predict joints when they are obscured or undetectable due to factors such as occlusion or poor illumination. Typically, such predictions not only require substantial computational resources but also lack a high level of confidence. Our proposed framework offers a strategic alternative by leveraging RFGCN to achieve action classification tasks even without complete postures.

3.2. Graph data definition

Graphs, denoted as $G = (V, E)$, are mathematical structures designed to capture pairwise relationships among objects, where V is the node set and E is the edge set. The conversion from raw images or videos into graphs simplifies the description of objects, highlights their spatial and temporal relationships, and also reduces interference from image backgrounds, which are sensitive to factors such as illumination and weather.

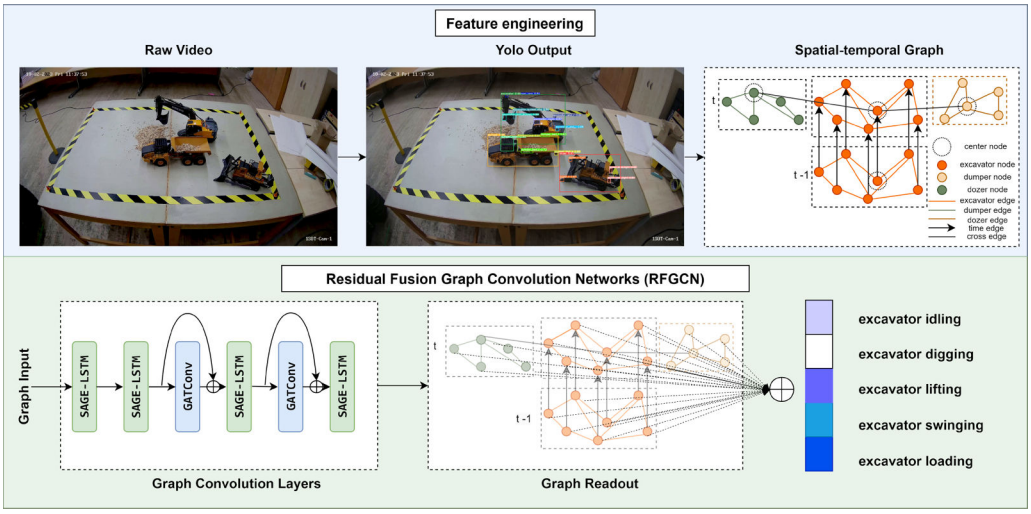

Fig. 1. The flowchart of the proposed framework. First, the raw video is used as input for an object detection model using YOLO v8 to identify key components of a specific plant. After obtaining the component class and bounding box information, these data are subsequently transformed into a graph data structure. This structure includes the location based on the centre of the bounding box, the class and the relationships with other components. Subsequently, the resulting graph datasets undergo grouping and labeling, and are then partitioned into training and testing sets. This division allows for the training and evaluation of the proposed graph classification model.

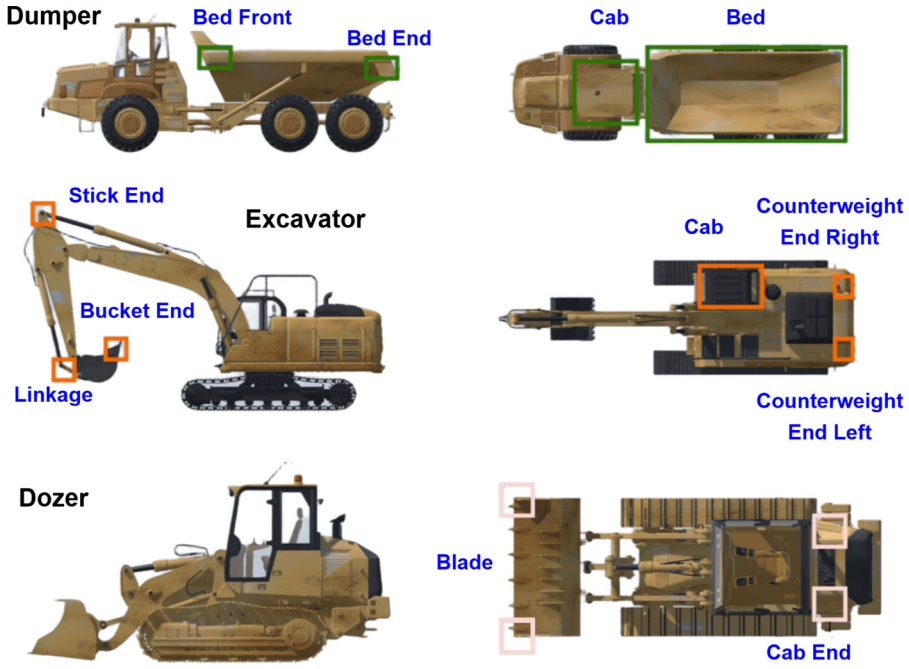

Fig. 2. An illustration of key components for three types of machinery. For the dumper, there are four distinct object types designated for parts, each delineated by green bounding boxes, and one category for the machinery itself (the black dash bounding box). For the excavator, we define six distinct object types, depicted by red boxes, for parts detection with one dedicated category for the machinery. For the dozer, we suggest two classes (each class includes two objects), illustrated by light orange boxes, and with one dedicated category for the machinery.

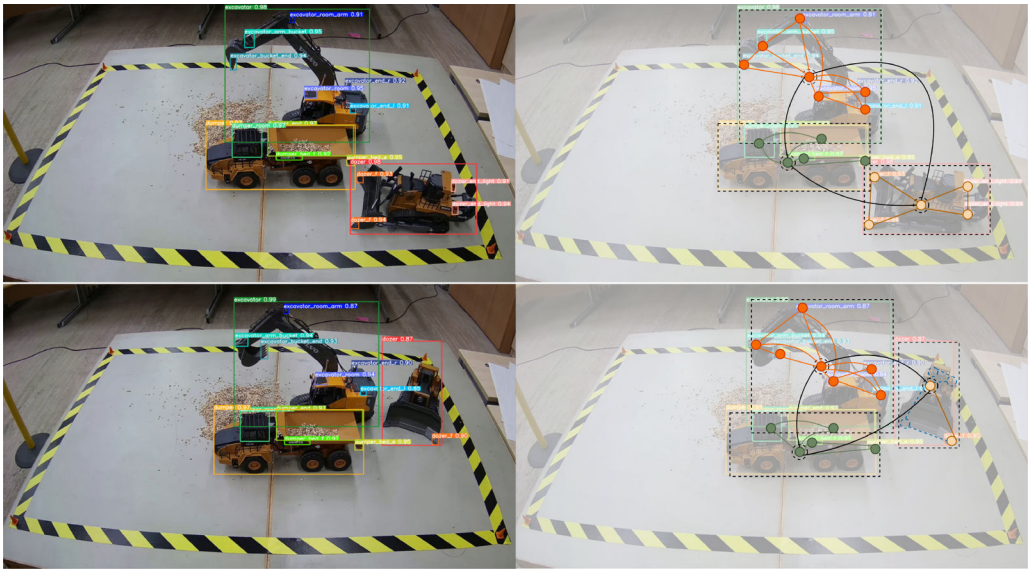

Fig. 3. Two examples for the outputs from YOLO v8 (left) along with an illustration of the graph structure from one image overlaid on the raw images (right). The orange dots represent nodes associated with the excavator, the green dots describe nodes for the dumper, and the yellow ones denote nodes for the dozer. The nodes enclosed by dashed circles represent the centres of the machinery's bounding boxes. The bottom part of the illustration demonstrates an example that includes missing nodes and edges, which are highlighted with blue dotted circles and blue dotted lines in the dozer.

In this study, we construct a Spatio-temporal Mixed Graph G with N nodes and T frames featuring intra-machinery, cross-machinery and inter-frame connections. In this graph, the node set is defined as

$$V = \{V_{exc}, V_{dum}, V_{doz}, \dots\} = \{v_{t,i} | t = 1, 2, \dots, T, i = 1, 2, \dots, N\}, \quad (1)$$

where $V_{exc}, V_{dum}, V_{doz}, \dots$ denote the node set of excavator, dumper, dozer and other type of machinery respectively. It should be noted that in the following section, only excavator, dumper, and dozer are chosen as an example of cooperating machinery but the vector V can be extended to other types of machinery. In this study, N is chosen as 17. As the input of the proposed RFGCN, the feature vector $h_{v_{t,i}}$ of each node in V consists of three attributes, including the category and location (x and y). The edge set E is composed of three subsets $E = \{E_S, E_M, E_T\}$. Each edge is unweighted, indicating only connectivity (e.g., connected nodes belong to a single machinery) or relationships (e.g., multiple machinery work together as a team) without additional attributes. The subset

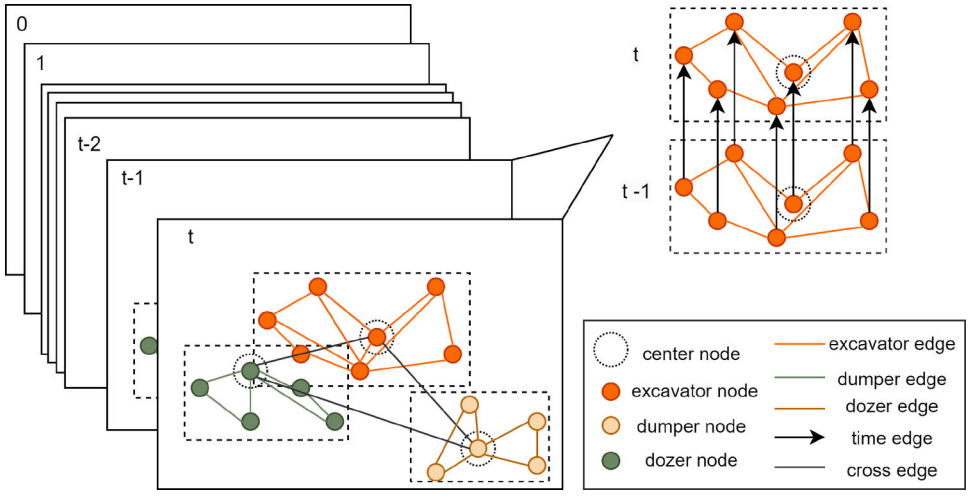

Fig. 4. An example of a spatial-temporal graph composed of two frames. The graph on the left illustrates the spatial distribution of notes from the excavator and other collaborating machinery, where each note connects to other notes within the same machinery. It should be noted that the central nodes are interconnected as well. The central nodes are represented by the center point of the bounding box of machinery. The right diagram depicts how the graph incorporates temporal information from two successive frames. There is a unidirectional edge from the previous frame to the current frame for each corresponding node in the excavator. It should be noted that the temporal connections of dumpers and dozers are not considered.

E_S depicts the spatial connection within each machinery at the time t, denoted as

$$\begin{aligned} E_S &= \{E_{S_{exc}}, E_{S_{dum}}, E_{S_{doz}}, \dots\}, \\ E_{S_{exc}} &= \{(v_{t,i}, v_{t,j}) | i \neq j, v_{t,i} \in V_{exc}, v_{t,j} \in V_{exc}\}, \\ E_{S_{dum}} &= \{(v_{t,i}, v_{t,j}) | i \neq j, v_{t,i} \in V_{dum}, v_{t,j} \in V_{dum}\}, \\ E_{S_{doz}} &= \{(v_{t,i}, v_{t,j}) | i \neq j, v_{t,i} \in V_{doz}, v_{t,j} \in V_{doz}\}, \\ &\vdots \end{aligned} \tag{2}$$

The subset E_M includes the spatial connections between each machinery's central node, written as

$$E_M = \{(v_{t,i_{exc}}, v_{t,i_{dum}}), (v_{t,i_{exc}}, v_{t,i_{doz}}), (v_{t,i_{doz}}, v_{t,i_{dum}}), \dots\}, \tag{3}$$

where $v_{t,i_{exc}}$, $v_{t,i_{dum}}$, and $v_{t,i_{doz}}$ denote the central nodes of the excavator, dumper, and dozer respectively. The subset E_M consists of the inter-frame edges of the excavator only, which connect the same nodes on the target machinery in consecutive frames, written as

$$E_T = \{v_{t-1,i} \rightarrow v_{t,i} | t = 2, \dots, T, v_{t,i} \in V_{exc}\}. \tag{4}$$

As illustrated in Fig. 4, the resulting mixed graph data retains spatio-temporal information with directed edges in the time domain and undirected edges in the space domain. The distribution of the nodes from the excavator and other cooperating machinery represents the spatial information. The change of each node's location with the excavator over time describes the temporal information. The number of frames to be considered in the time window, T, is a parameter which should be adjusted based on the complexity of targeted activities. A long time window can potentially capture the graph pattern of a complex activity with a significant computational cost. This graph includes component-based keypoints and the geometric centre of the entire machinery, referred to as the central node. E_S describes the pose of individual machinery; E_M represents the interactions between different types of machinery; E_T defines the movement of the excavator, which is directional.

It should be noted that this framework is inherently flexible and can be adapted to various machinery types beyond the excavator, which serves as the primary focus in this study. The design allows the excavator node set V_{exc} to be replaced by other types of machinery, such as cranes or loaders, without altering the underlying graph structure. Additionally, the number of cooperating machines can be adjusted by adding or removing nodes in the graph, enabling seamless scaling to suit different operational scenarios in practical applications. This adaptability stems from the framework's abstraction of the graph data, retaining only essential node and edge connections as input. As a result, the framework is highly modular, allowing it to generalize across different machinery configurations while maintaining core relational information.

The selection of the three edge types, including E_S (intra-machinery), E_M (inter-machinery), and E_T (temporal), is grounded to capture essential contextual dimensions for robust activity classification. E_S encodes the internal structural relationships within individual machinery, which is critical for modeling pose and configuration. E_M captures the cooperation between different types of machinery, such as excavators and dumpers, enabling the system to understand multi-agent collaborative tasks like loading. Finally, E_T introduces temporal consistency by linking the same components across consecutive frames, allowing the model to learn motion patterns and activity progression.

3.3. Model architecture

Our proposed classifier model, Residual Fusion Graph Convolution Networks (RFGCN), is based on Graph Convolutional Networks (GCNs) and is depicted in the green section in Fig. 1. The architecture consists of Graph Convolution layers, a graph readout, and soft classification. The model integrates a variety of layer types, incorporating both residual connections and attention-based layers, to bolster performance and ensure robustness.

3.3.1. Scalable graph convolution layer

To address the limitations of GCNs in handling unseen data, the proposed RFGCN leverages GraphSAGE, a framework for inductive representation learning on large graphs introduced by Hamilton et al. (2017). Fig. 5 depicts an example of the GraphSAGE process. This method enables inductive learning by using node feature information to generate embeddings for new data. The state-of-the-art (SOTA) offers three types of aggregators, Mean aggregators, LSTM aggregators, and Pooling aggregators. In mean aggregations, neighbouring node features are averaged, which can result in a loss of important information as it smooths out the features. Pooling aggregators can be too extreme, capturing only one aspect of the neighbouring nodes by taking the maximum value over each dimension from its neighbours. To accommodate a larger expressive capacity, this study employs a more complex aggregator based on an LSTM architecture (Hochreiter & Schmidhuber, 1997), as the aggregator for the GraphSAGE layer.

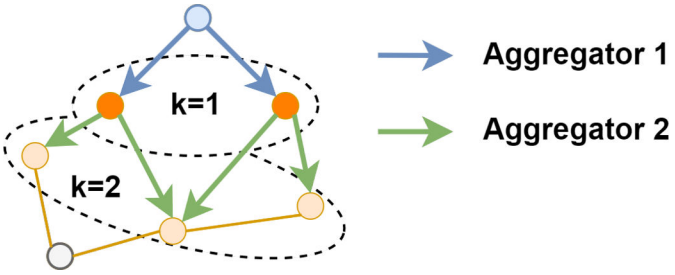

Fig. 5. An example of the GraphSAGE (Hamilton et al., 2017) process. The blue circle represents the object node, the orange ones represent the 1st-order neighbourhood, and the pink ones represent the 2nd-order neighbourhood.

The operation of the GraphSAGE layer with an LSTM aggregator can be broken down as follows. First, neighbourhood sampling is carried out. For a specified node $v_{t,i}$, where the temporal frame is denoted by t ranging from 1 to T, and the nodal index is presented by i spanning 1 to N. The LSTM-based GraphSAGE includes three systematic phases, which are neighbourhood sampling, feature concatenation and LSTM aggregation mechanism. Each node $v_{t,i}$ in the graph is associated with a feature vector $h_{v_{t,i}}$. In the neighbourhood sampling, an initial step necessitates the extraction of a pre-defined neighbouring node-set, $V_{N(v_{t,i})}$, which contains all connections to $v_{t,i}$ within the graph G. For each node $v_{t,i}$, the feature vectors of its neighbouring nodes are denoted as $h_{V_{N(v_{t,i})}}$. In the second step, for each $v_{t,i}$, the feature vector is concatenated with the feature vectors of its neighbours, written as $[h_{v_{t,i}}, h_{V_{N(v_{t,i})}}]$. The third step sequentially processes the concatenated features using an LSTM aggregation, written as:

$$h^{(l+1)}_{V_{N(v_{t,i})}} = aggregate\left(\left\{h^{(l)}_{v_{t,j}}, \forall v_{t,j} \in V_{N(v_{t,i})}\right\}\right), \tag{5}$$

Subsequently, the node's feature is updated using the aggregated information:

$$h^{(l+1)}_{v_{t,i}} = \sigma\left(W \cdot concat\left(h^{(l)}_{v_{t,i}}, h^{(l+1)}_{V_{N(v_{t,i})}}\right)\right), \tag{6}$$

where $h^{(l)}_{v_{t,i}}$ is the hidden state of the node $v_{t,i}$ in the layer l of the neural network. W is a weight matrix and σ is the activation function.

3.3.2. GAT layer

The effectiveness of node embeddings often relies on their ability to seamlessly integrate both the local graph structure and node-specific features, as demonstrated by Kipf and Welling (2017). Traditional methods such as Graph Convolutional Networks (GCN), despite their effectiveness, exhibit an inherent structural dependency during aggregation, potentially leading to challenges in generalisation.

In contrast, the Graph Attention Network (GAT) offers a refined aggregation mechanism that is structurally agnostic. Unlike the previously discussed SAGE-LSTM, GAT emphasis on weighting neighboring features based on the principle of attention. This mechanism ensures that the weights are adjusted based on node features rather than the graph's structure. Following the aggregation, a LeakyReLU activation function is typically applied. Below are the equations to compute the node embedding $h^{(l+1)}_{v_{t,i}}$ of the layer $l+1$ from the embeddings of the layer l:

$$z^{(l)}_{v_{t,i}} = W^{(l)} h^{(l)}_{v_{t,i}}, \tag{7}$$

$$e^{(l)}_{v_{t,i},v_{t,j}} = \text{LeakyReLU}\left(\mathbf{a}^{(l)T}\left(z^{(l)}_{v_{t,i}} \parallel z^{(l)}_{v_{t,j}}\right)\right), \tag{8}$$

$$\alpha^{(l)}_{v_{t,i},v_{t,j}} = \frac{\exp(e^{(l)}_{v_{t,i},v_{t,j}})}{\sum_{v_{t,j}\in V_{N(v_{t,i})}} \exp(e^{(l)}_{v_{t,i},v_{t,j}})}, \tag{9}$$

$$h^{(l+1)}_{v_{t,i}} = \sigma\left(\sum_{v_{t,j}\in V_{N(v_{t,i})}} \alpha^{(l)}_{v_{t,i},v_{t,j}} z^{(l)}_{v_{t,j}}\right). \tag{10}$$

Eq. (7) represents a linear transformation of the lower layer embedding $h^{(l)}_{v_{t,i}}$ using $W^{(l)}$, a learnable weight matrix. Eq. (8) computes a pairwise unnormalised attention score between two neighbours. It begins by concatenating the z embeddings of the two nodes (where $\parallel$ denotes the concatenation operation), then calculating a dot-product with a learnable weight vector $\mathbf{a}^{(l)}$, and finally applies a LeakyReLU activation function. This form of attention is usually referred to as additive attention, in contrast to the dot-product attention found in the Transformer model. Eq. (9) applies a softmax function to normalise the attention scores for each node's incoming edges. Eq. (10) aggregates the embeddings from neighbours, scaled by the attention scores, similar to GCN.

3.3.3. Graph residual connection

Given the nature of incomplete spatio-temporal graphs, it is crucial to consider two points. First, sequential information could be beneficial for the temporal information on spatio-temporal graphs. Second, considering there are some missing nodes or edges on graphs, it is important to reduce the impact of missing data. GraphSAGE-LSTM layer and GAT layer could be applied to meet the two requirements. Combining GraphSAGE-LSTM and GAT in a layered architecture can potentially have the benefits of both models and mitigate some of their individual disadvantages. Additionally, combining these two distinct graph convolutional layers increases the overall number of layers in the neural network. This necessitates the introduction of residual connections to mitigate the effects of the increased layer count.

DeepGCNs (Li et al., 2019), inspired by ResNet (He et al., 2016) and DenseNet (Huang et al., 2017), apply residual connectivity across layers. This approach demonstrates that deepening the GCN network with specific strategies can enhance its effectiveness. In this study, we introduce residual connectivity between the outputs of the GraphSAGE and GAT layers, treating this as a block. The residual block can be expressed with the following function:

$$h^{(l+1)}_{Res} = \sigma\left(concat\left(h^{(l+1)}_{GAT}, h^{(l)}_{GraphSAGE-LSTM}\right)\right). \tag{11}$$

Unlike generic hybrid GNN models such as GatedGCN (Yang, 2023) or Graph Transformers (Müller et al., 2023), which typically integrate attention or gating mechanisms uniformly across all layers or nodes, our proposed RFGCN employs a modular residual fusion strategy that explicitly combines the outputs of heterogeneous graph convolution layers GraphSAGE-LSTM and GAT at a block level. This parallel fusion is designed to leverage the complementary strengths of sequential modeling (via LSTM-based aggregation) and structural feature weighting (via attention), rather than stacking them sequentially. Moreover, our residual design selectively connects the GraphSAGE-LSTM input to the output of the GAT layer, enabling effective gradient flow and robustness in deeper architectures. This is particularly beneficial for incomplete spatio-temporal graphs, where isolated attention or gating may fail to compensate for missing structural cues. The resulting architecture is not only more interpretable but also more efficient, as it avoids the complexity of global attention or full-layer self-attention mechanisms commonly used in transformer-based GNNs.

3.3.4. Graph readout

Each graph in the dataset has its unique structure and characteristics of nodes and edges. Therefore, to predict a single graph, it is common to aggregate as much information as possible in a single graph. This is a process known as 'Graph Readout'. Common aggregation methods include summing, averaging, or finding the maximum or minimum value across all node or edge features. In this study, we employ averaging as the aggregation method. Given a graph G, the average node feature readout is defined as:

$$h_G = \frac{1}{|V|}\sum_{v_{t,i}\in V} h_{v_{t,i}}, \tag{12}$$

where V is the set of the nodes and $h_{v_{t,i}}$ is the feature of the node $v_{t,i}$. In this model, once h_G is available, the data is passed through a *Dropout*

layer and a *Linear* layer for producing the classification result. Cross entropy is used as the *loss function*, denoted by L, which can be written as:

$$L = -\sum_{c=1}^{M} y_{o,c} \log\left(p_{o,c}\right), \tag{13}$$

where M is the number of classes; $y_{o,c}$ is a binary indicator (0 or 1) if the class label c is the correctly classified for the observation o; and $p_{o,c}$ is the predicted probability that the observation o is of the class c.

4. Results

The results are structured to first evaluate the performance of object detection, and then that of the proposed Residual Frame Graph Convolutional Network (RFGCN) in classifying excavator activities. Additionally, we assessed the model performance across different numbers of frames (T), investigating its adaptability in processing temporal data in comparison with SOTA.

4.1. Evaluation of object detection performance

To demonstrate the result of the first step, the outputs of YOLOv8 are presented in Fig. 6 for each of the five distinct actions. The solid circles denote the centre of each detected object and the double-circles represent the centre of individual machinery. It can be observed that, while most nodes are correctly identified, there are instances of failed recognition of certain objects when the machinery is partially occluded by itself or other machinery. For example, in the third column of images depicting the swinging action, the bucket's end and the right end of the excavator body are not detected, resulting in missing nodes. Moreover, illumination can affect the performance of object detection. In the third column of images showing the loading action, neither end of the excavator body is detected. The left end is obscured, while the right end is indistinct due to lighting issues such as shadows. In the digging action, the bucket end of the excavator submerged in the sand also failed to detect. Similar recognition problems are encountered with the dozer's front and back ends, as they are difficult to distinguish from each other. These observations demonstrate the real-world challenge that not all important nodes can be successfully detected for various reasons, which demands a flexible classification approach.

4.2. Performance of RFGCN in activity classification using spatial-temporal information

Fig. 7 represents the confusion matrix of the proposed model when applied on the dataset mentioned in the above section. The confusion matrix is normalised, which means that each value is divided by the number of samples in the corresponding class. Within the matrix, the diagonal values represent true positives, indicating cases where the predicted label matches the actual label. The off-diagonal values represent false positives and false negatives, presenting cases where the predicted label does not match the actual label. The confusion matrix reveals that our method achieves a high level of accuracy (> 94.8 % average accuracy) in classifying all five excavator activities. The accuracy in identifying the "loading" activity is perfect, which is very likely due to the consideration of additional information from other cooperating machinery, as it is a typical collaborative activity. The other four activities are solo based on the excavator without any contribution from other machinery. This is why they achieved relatively lower accuracy.

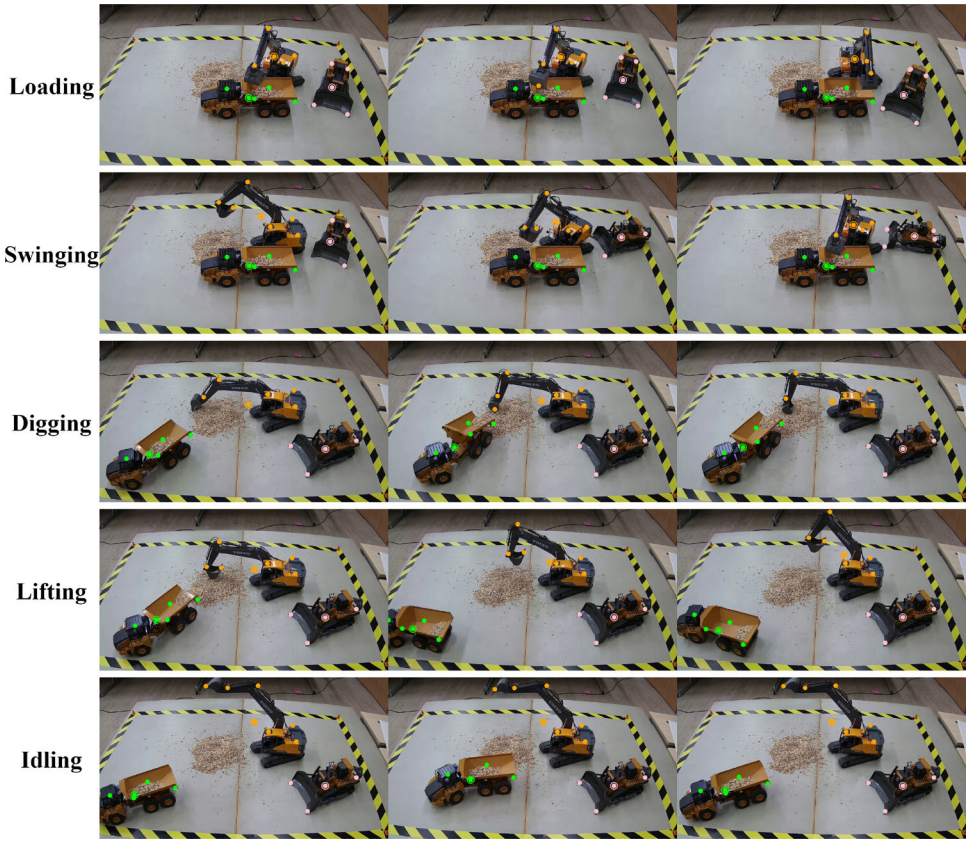

Fig. 6. The visualisation of the sample of the object detection output for five investigated actions. Points represent the centre of the bounding box. Green points describe the centre of the objects on the dumper. Orange points describe the ones on the excavator. Pink points show the ones for the dozer. The points with bounding cycles describe the centre points of the bounding box for the entire machinery.

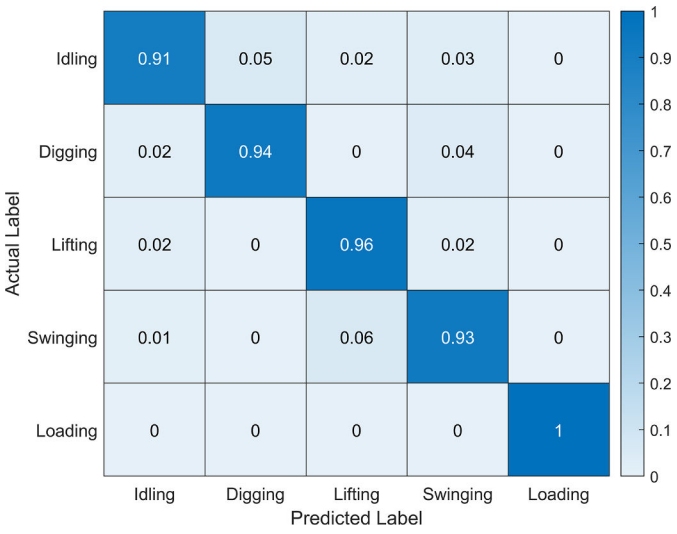

Fig. 7. The confusion matrix of one test using the proposed method, where $T = 3$.

4.3. Performance against state-of-the-art

Fig. 8 illustrates a comparative performance analysis of the proposed methods along with four selected SOTA methods, with variations in the number of adjacent frames used in the RFGCN T. The selected GraphSAGE-LSTM (Hamilton et al., 2017), GAT (Veličković et al., 2018), GCN (Kipf & Welling, 2017) and GIN-0 (Xu et al., 2019) methods used a single type of layer, while the proposed method uses two types of layers. Each of the five methods had a total of six layers. It can be observed that our method outperforms all SOTA for each number of used frames. Additionally, the performance of the proposed method improves steadily as the frame number increases. This observation suggests that this method effectively utilises additional temporal information for better classification while mitigating overfitting problems when considering more features. It should be noted that, as expected, the number of frames significantly affects computational cost. During real-world deployments, there is a trade-off between classification accuracy and real-time performance. GraphSAGE-LSTM exhibits relatively close performance but its accuracy experiences significant fluctuations when the number of frames exceeds 7. The proposed method, along with GraphSAGE-LSTM and GAT performs relatively well when the number of frames is between 5 and 7,

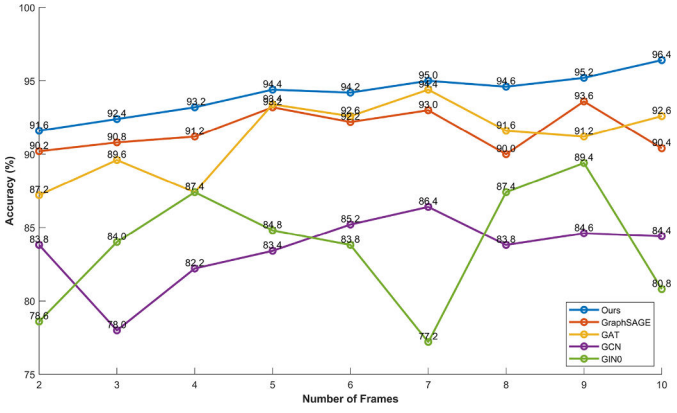

Fig. 8. Performance comparison of the proposed method with four selected state-of-the-art (SOTA) based on graph convolution networks and spatial-temporal information. The x-axis represents the number of frames used, T, including the target frame. Each shown accuracy is the mean accuracy of ten practice results.

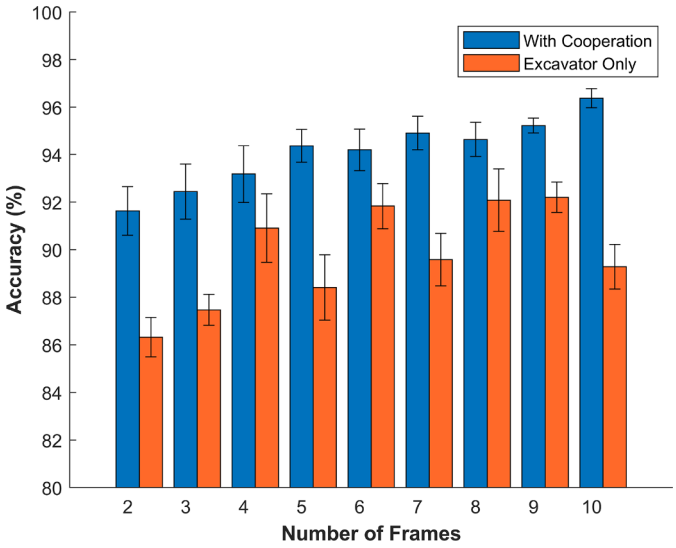

Fig. 9. The performance comparison of RFGCN for the scenario considering the cooperation machinery and the scenario without.

which suggests that a video clip with a duration of 0.5–0.7 s is the most appropriate for this classification task. Overall, GCN and GIN-0 achieved relatively lower performance in comparison to the other three methods in terms of both accuracy and stabilisation.

5. Discussion

5.1. Impact of cooperative machinery on classification accuracy

To demonstrate the additional contribution of cooperative machinery to the classification, Fig. 9 plots the performance comparison of RFGCN between the proposed solution and the scenario where only the excavator is considered. For the scenario where the cooperative machinery is considered, the performance commences at a high accuracy of approximately 92 %. Notably, this performance exhibits a stable and gradual increase with the incremental T, illustrating the positive impact of incorporating a broader contextual perspective. Moreover, when the frame number increases, the proposed solution exhibited a smaller variation of accuracy for a specific T. This observation suggests the proposed solution is more stable, and the trained model will be more repeatable when more temporal information is considered.

Conversely, the scenario featuring only the excavator achieves a much lower accuracy. It also exhibits more significant fluctuations across different values of T, suggesting potential overfitting. Moreover, in general, it has relatively larger variations of accuracy for a specific T, indicating a relatively compromised performance in repeatability. One reason is that the loading activity usually requires a collaboration between an excavator and a dumper. Considering the additional information from cooperative machinery could be beneficial to improve the overall accuracy.

5.2. Performance against missing nodes

To demonstrate the robustness of the proposed method with missing nodes, we conducted six additional tests, individually omitting each node in the excavator while preserving the other 5 nodes in the testing phase. Fig. 10 presents a compelling comparison with two selected SOTA methods, GraphSAGE-LSTM and GAT, where all three methods use 3 adjacent frames ($T = 3$).

The proposed algorithm, as visualised by the blue bars, exhibits remarkable resilience across various scenarios, significantly outperforming (> 10 %) other methods in all cases, with the exception of the case

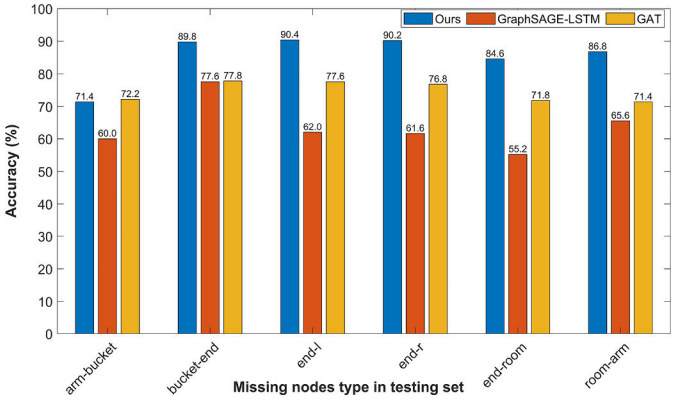

Fig. 10. A comparison of model performance with two SOTA methods when a certain node in the excavator is missing in the testing phase, where $T = 3$ for all three methods.

where the 'arm-bucket' node is missing. This observation demonstrates the superior performance of the proposed method in terms of inferring activities from partial data. It should be noted that the accuracy reflects the importance of a certain node to classification. The scenario where the "arm-bucket" node is missing achieves only an accuracy of 71.4 %, while the other five scenarios obtained an accuracy of more than 84 %. This observation suggests that the "arm-bucket" node in the excavator is critical to determine the activity while the other 5 nodes are relatively less important. When this important node is missing, the proposed method has a similar performance as GAT. Moreover, in general, GAT performs better than GraphSAGE-LSTM for all six tests.

5.3. Influence of residual terms on model performance

To explore the impact of various graph residual terms on model performance, we conducted a sensitivity analysis using four variations of graph neural network architectures. These residual connections are designed to enable the flow of gradients during training, potentially improving the learning process and enhancing model accuracy by mitigating issues such as the vanishing gradient problem. As illustrated by Fig. 11, the proposed model applies the residual terms from the input of the GAT layer to the output of the GAT layer. According to research by Zhang and Meng (2019), the residual terms can be carried out as 'full naive' and 'full raw'. Naive residual terms involve connecting the input and output for each layer, assuming their independence and determination by the current state. Raw (top to all) residual terms are assumed to be independent and determined by the features of the first layer output. We also conducted an experiment with Raw (all to end) residual terms, assuming their independence and determination by the output of all layers.

The experimental results are summarised in Table 1, where three adjacent frames were used for all five configurations of residual terms. The proposed model, which integrates a specific design of residual connections, demonstrates top performance with an overall accuracy of 92.4 %. This outcome not only confirms the efficacy of the proposed model but also highlights the value of a well-designed residual structure in graph neural networks. The performance differences between models employing various residual terms were not substantial, indicating that while the proposed model's residual design is effective, other configurations also contribute positively to model performance. However, the experiment revealed a notable drop in accuracy to 90.6 % when all residual connections were omitted. This significant decrease in performance emphasises the importance of residual connections in the architecture. Residual terms evidently play a pivotal role in stabilising and facilitating the training of deep neural networks, thereby contributing to the enhancement of classification accuracy. This result highlights the importance of incorporating residual connections in the design of graph convolutional networks, particularly when dealing with complex spatio-temporal data.

Table 1
Performance comparison of different residual configurations.

Residual configuration	Accuracy (%)
Ours	**92.4**
Full Naive	91.8
Full Raw (Top to All)	92.2
Full Raw (All to End)	91.2
No Residual Terms	90.6

Table 2
Comparison of model complexity and inference cost for different architectures (same layers).

Model	Trainable parameters	FLOPs
GAT	6.84M	3.38M
GraphSAGE-LSTM	5.92M	10.36M
RFGCN (Ours)	**5.78M**	**51.3M**
YOLOv8m (for object detection)	25.9M	78.9B

5.4. Ablation studies

To validate the contribution of each key component in the RFGCN framework, we conducted a series of ablation studies.

Residual Connections: As shown in Table 1, removing residual connections led to a 1.8 % drop in accuracy, highlighting their importance for stable training and deeper architecture effectiveness.

Fusion of GraphSAGE-LSTM and GAT Layers: Fig. 8 shows that the combined model consistently outperforms standalone GraphSAGE-LSTM and GAT implementations, verifying the effectiveness of the dual-layer design.

Cooperative Machinery Integration: As shown in Fig. 9, excluding cooperating machinery from the graph results in reduced accuracy and higher variability, demonstrating the value of inter-machinery connections in activity recognition.

Robustness to Missing Nodes: Fig. 10 demonstrates the model's superior performance when one key node is missing, compared with other models, confirming that RFGCN handles partial data robustly-an essential capability in real-world applications.

5.5. Practical considerations: computational efficiency

To assess the practical feasibility of deploying RFGCN in real-time construction environments, we analysed its computational efficiency. The model contains approximately 3.15 million trainable parameters, and a typical forward pass involving a graph of 31 nodes and 55 edges requires 51.1 million floating point operations (FLOPs) (based on input is a non-missing graph size, with 3 frames). As YOLOv8 is not the focus of our contribution, we have not included a detailed discussion of it in this section.

As shown in Table 2, RFGCN achieves a favorable trade-off between parameter size and inference cost compared to pure GAT and GraphSAGE-LSTM architectures. Despite using a residual fusion strategy, RFGCN maintains a low parameter count (5.78M) and a modest inference load (51.3M FLOPs), making it computationally efficient for real-time deployment. For the object detection phase, we employ YOLOv8m, which has a computational cost, approximately 25.9M parameters and 78.9 billion FLOPs per image, but remains one of the most optimised models for high-accuracy small object detection. Overall, the combined pipeline remains viable for real-time performance, especially when deployed on modern GPUs or edge devices capable of parallel processing of graph and image inference.

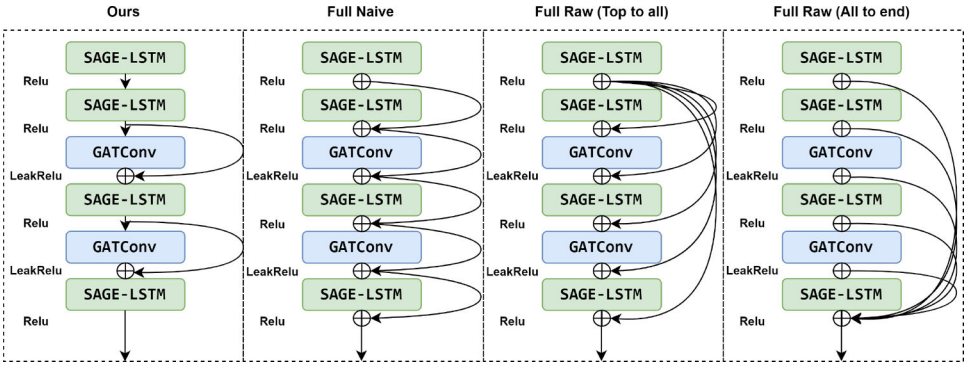

Fig. 11. A comparison of the network structure of our proposed model with different graph residual terms.

5.6. Performance on real-world datasets

This section presents the performance of the proposed framework on datasets from real construction scenarios. These datasets were collected from videos on YouTube. The dataset for the YOLOv8 model includes 265 manually labeled images sampled from these videos. Data augmentation was then applied through horizontal and vertical flips, as well as brightness adjustments ranging within ±25 %. This resulted in a dataset comprising a total of 795 images. In each image, there is at least one excavator and up to 4 dumpers. The specific models of excavators and dumpers are not fixed. The ratio of the training set to the testing set is 8:2. Roboflow was used as the image annotation tool, and the output is in the COCO format. Before resizing to $1 \times 3 \times 1920 \times 1080$, the original size of the images was $1 \times 3 \times 3840 \times 2160$. To demonstrate the result of the first step, the outputs of YOLOv8 on some testing images are presented in Fig. 12. Compared to the mini-construction dataset, the machinery in the real construction scenarios occupies a larger portion of the full image. However, the challenge of object recognition remains due to the varying appearances of the same type of machinery in complicated environment. For example, the topmost 'boom-arm' of the excavator in the first column of the third row is not recognised, and only one point on the tail of the excavator in the third column of the second row is recognized. Therefore, the flexibility to tackle missing node in the proposed solution is especially important and helpful for read world data.

In the next step, we used the same process to generate a dataset of graphs using the trained YOLOv8 model. The dataset consists of 3 videos, each containing 330 image sequences, totaling 990 frames. These 990 frames are grouped into graphs, with each graph consisting of three consecutive frames ($T = 3$). For each video, 328 graphs can be generated using a sliding windows with a window size of three consecutive frames, resulting in a total of 884 graphs. The ratio of the training set to the testing set is 8:2. The actions are categorised into five categories: idling, digging, lifting, swinging, and loading. Fig. 13 represents the normalised confusion matrix of the proposed model when applied to the testing dataset from real construction scenarios. Overall, the accuracy for each class was above 90 %, which demonstrates the effectiveness of the proposed solution for real world data.

5.7. Performance evaluation on public datasets

The proposed solution aims to excel not only in activity recognition based on spatial-temporal information but also spatial information only. To validate this hypnosis, we conducted a test using the proposed method on four well-known datasets. They consist of two social network datasets (IMDB-M and IMDB-B) and two bioinformatics datasets (MUTAG and PTC), which are commonly used for evaluating graph classification algorithms. There are no node features in both social network datasets. Therefore, we followed the approach proposed by Niepert et al. (2016); Zhang et al. (2018), setting the node degrees as the features. Detailed descriptions of each dataset are provided below.

- The MUTAG (Kersting et al., 2016) dataset consists of 88 chemical compounds, each of which is represented as a graph where vertices correspond to atoms and edges indicate the chemical bonds between them. Originally designed to assess the mutagenic effects

Fig. 12. The visualisation of the object detection output on the testing dataset from real construction scenarios.

Table 3
Comparison of classification accuracy (%) with state-of-the-art methods.

Model	IMDB-B	IMDB-M	MUTAG	PTC
GCN (Kipf & Welling, 2017)	73.30 ± 5.29	51.20 ± 5.13	87.20 ± 5.11	–
GAT (Veličković et al., 2018)	70.50 ± 2.30	47.80 ± 3.10	89.40 ± 6.10	66.70 ± 5.10
DGCNN (Zhang et al., 2018)	70.03 ± 0.86	47.83 ± 0.85	85.83 ± 1.66	58.59 ± 2.47
GraphSAGE (Hamilton et al., 2017)	72.40 ± 3.60	49.90 ± 5.00	79.80 ± 13.9	–
GIN-0 (Xu et al., 2019)	75.10 ± 5.10	52.30 ± 2.80	89.40 ± 5.60	64.60 ± 7.00
U2GNN (Nguyen et al., 2022)	77.04 ± 3.45	53.60 ± 3.53	89.97 ± 3.65	69.63 ± 3.60
CIN + + (Giusti et al., 2023)	–	–	94.4 ± 3.7	–
GIUNet (Amouzad et al., 2024)	79.0	52.6	94.7	85.7
ESA (Buterez et al., 2024)	96.9 ± 1.3	48.7 ± 1.1	–	–
Ours	71.05 ± 2.15	56.50 ± 2.00	94.47 ± 1.90	69.57 ± 2.10

Note: Standard deviation for GIUNet is not available.

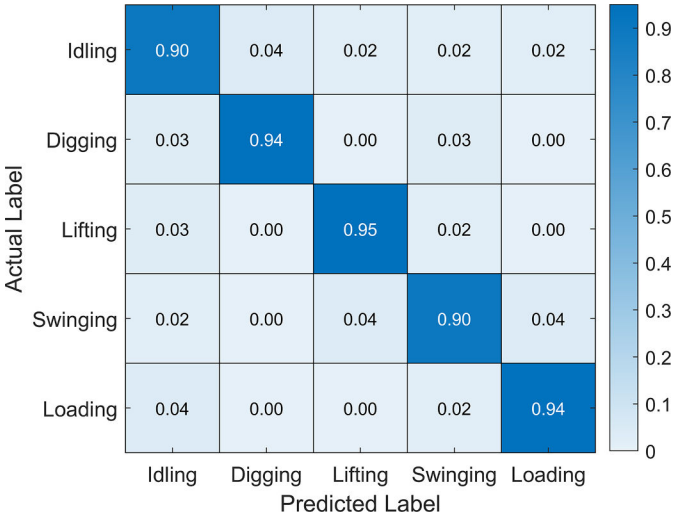

Fig. 13. The confusion matrix of testing image sequences from real construction scenarios using the proposed method, where $T = 3$.

of these compounds, the dataset have two class: mutagenic and non-mutagenic.

- The IMDB (Yanardag & Vishwanathan, 2015) dataset includes two datasets: IMDB-B and IMDB-M. They serve as a benchmark for two-class and multi-class graph classification tasks, respectively. Each dataset includes approximately 1000 graphs and 1500 graphs respectively. Each graph in this dataset represents an ego-network of an actor or actress, with nodes representing the actors or actresses themselves. The edges between nodes are indicative of co-appearances in movies. Each graph is derived from a pre-specified genre of movies. The graphs in the IMDB-B dataset are derived from the Action and Romance genres. The IMDB-M dataset is categorised into three distinct classes: Action, Comedy, and Drama. Node features in this dataset may vary and include attributes such as actor popularity or average movie rating. Edge features, although not consistent across all versions, generally represent the co-appearance of actors in the same movie.
- The Predictive Toxicology Challenge (PTC) (Toivonen et al., 2003) dataset is designed for binary graph classification and consists of 344 graphs. Each graph represents a chemical compound, with nodes representing atoms and edges denoting chemical bonds between them. The dataset is bifurcated into two classes: Carcinogenic and Non-carcinogenic. Node features commonly include atom type, atomic number, and other relevant chemical properties. Similarly, edge features often include the type of chemical bonds (single, double, or triple).

Table 3 presents the classification results of the proposed method along with seven state-of-the-art methods for the four selected public datasets. The proposed method achieved an accuracy of 56.50 % ± 2.00 % for the IMDB-M dataset, outperforming all other methods. Similarly, for the MUTAG dataset, our method attained an accuracy of 94.47 % ± 1.90 %, matching or surpassing the best reported performances such as those achieved by GIUNet (94.7 %) and U2GNN (89.97 %). For the PTC dataset, our method achieved an accuracy of 69.57 % ± 2.10 %, which is very close to the top accuracy of 69.63 % ± 3.60 % reported by U2GNN. The performance on the IMDB-M dataset was also competitive, achieving the highest accuracy among all tested methods. These results indicate that the proposed method has superior performance in graph-based classification tasks, demonstrating robustness across diverse datasets with varying graph structures.

6. Conclusion

To address the challenge of the classification of excavator-involved activities, especially when the target parts are occluded or hard to detect, this paper reports a robust Residual Fusion Graph Convolution Network (RFGCN) based framework for inferring earth-moving activities from machinery keypoints extracted from videos. First, we introduced a Yolov8-based model to extract the location and type of machinery keypoints. To handle information on keypoint relationships within the same machinery or across machines at different times, we introduced the Residual Fusion Graph Convolution Network (RFGCN) model, which focuses on processing and aggregating keypoint attributes of action parts and the relationships between them. With such a robust model, spatial-temporal information hidden in the keypoints can be learned, which enhances the accuracy of graph classification. Another key feature of this approach is that this approach is capable of effectively handling missing nodes or edges.

The main research findings include:

- The proposed RFGCN model effectively predicts the activity class of target machinery, even when nodes or edges are missing;
- An architecture with mixed graph convolution layers is a more appropriate choice than one with a single type layer for this specific application;
- The inclusion of nodes and relationships with cooperating machinery can significantly improve classification accuracy;
- Effective utilisation of both spatial and temporal information in the architecture can significantly enhance the performance of activity classification.

It is important to note that the proposed solution has certain limitations. As a computer vision-based method, it still heavily relies on high-quality video data even it has a high robustness against missing data. A potential solution for future studies could involve the integration of data from other types of sensors. Furthermore, the current approach may not adequately adapt to new or previously unseen construction scenarios without retraining or fine-tuning. To enhance the model's adaptability, future work could explore transfer learning or meta-learning strategies, which would allow the model to generalise more efficiently to new tasks

with minimal data requirements. Lastly, while the proposed method demonstrates robustness against partial data occlusion, it may still encounter challenges when faced with servere occlusions or very sparse data.

The precise classification of excavator actions promotes automation and digitisation within the construction sector. These advancements are crucial as they help establish safer, more efficient, and sustainable construction processes. By establishing a symbiotic relationship between the physical tasks of excavators and their digital counterparts, this research contributes to the foundation for intelligent construction ecosystems.

CRediT authorship contribution statement

Shuozhi Wang: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Validation, Visualization, Writing – original draft, Writing – review & editing; **Yitian Zhao:** Validation, Writing – review & editing; **Zichao Zhang:** Validation, Writing – review & editing; **Yifan Zhao:** Conceptualization, Supervision, Funding acquisition, Project administration, Resources, Writing – review & editing.

Data availability

Data will be made available on request.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Amouzad, A., Dehghanian, Z., Saravani, S., Amirmazlaghani, M., & Roshanfekr, B. (2024). Graph isomorphism U-Net. *Expert Systems with Applications*, *236*, 121280. https://doi.org/10.1016/j.eswa.2023.121280

Assadzadeh, A., Arashpour, M., Brilakis, I., Ngo, T., & Konstantinou, E. (2022). Vision-based excavator pose estimation using synthetically generated datasets with domain randomization. *Automation in Construction*, *134*, 104089. https://doi.org/10.1016/j.autcon.2021.104089

Barra, P., Bisogni, C., Nappi, M., & Ricciardi, S. (2018). Fast quadtree-based pose estimation for security applications using face biometrics. In M. H. Au, S. M. Yiu, J. Li, X. Luo, C. Wang, A. Castiglione, & K. Kluczniak (Eds.), *Network and system security* (pp. 160–173). Cham: Springer International Publishing.

Broo, D. G., Bravo-Haro, M., & Schooling, J. (2022). Design and implementation of a smart infrastructure digital twin. *Automation in Construction*, *136*, 104171. https://doi.org/10.1016/J.AUTCON.2022.104171

Buterez, D., Janet, J. P., Oglic, D., & Lio, P. (2024). An end-to-end attention-based approach for learning on graphs. https://arxiv.org/abs/2402.10793

Chen, C., Xiao, B., Zhang, Y., & Zhu, Z. (2023). Automatic vision-based calculation of excavator earthmoving productivity using zero-shot learning activity recognition. *Automation in Construction*, *146*, 104702. https://doi.org/10.1016/j.autcon.2022.104702

Chen, C., Zhu, Z., & Hammad, A. (2022). Critical review and road map of automated methods for earthmoving equipment productivity monitoring. *Journal of Computing in Civil Engineering*, *36*(3), 03122001. https://doi.org/10.1061/(ASCE)CP.1943-5487.0001017

Chiachío, M., Megía, M., Chiachío, J., Fernandez, J., & Jalón, M. L. (2022). Structural digital twin framework: Formulation and technology integration. *Automation in Construction*, *140*, 104333. https://doi.org/10.1016/J.AUTCON.2022.104333

Cormier, M., Clepe, A., Specker, A., & Beyerer, J. (2022). Where are we with human pose estimation in real-world surveillance? In *Proceedings of the IEEE/CVF winter conference on applications of computer vision (WACV) workshops* (pp. 591–601).

Geimer, M. (2020). Mobile working machines. SAE International.

Giusti, L., Reu, T., Ceccarelli, F., Bodnar, C., & Liò, P. (2023). Cin++: Enhancing topological message passing. https://arxiv.org/abs/2306.03561

Golparvar-Fard, M., Heydarian, A., & Niebles, J. C. (2013). Vision-based action recognition of earthmoving equipment using spatio-temporal features and support vector machine classifiers. *Advanced Engineering Informatics*, *27*(4), 652–663. https://doi.org/10.1016/j.aei.2013.09.001

Gong, J., & Caldas, C. H. (2010). Computer vision-based video interpretation model for automated productivity analysis of construction operations. *Journal of Computing in Civil Engineering*, *24*(3), 252–263. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000027

Guo, Y., Cui, H., & Li, S. (2022). Excavator joint node-based pose estimation using lightweight fully convolutional network. *Automation in Construction*, *141*, 104435. https://doi.org/10.1016/j.autcon.2022.104435

Hamilton, W. L., Ying, R., & Leskovec, J. (2017). Inductive representation learning on large graphs.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 770–778). https://doi.org/10.1109/CVPR.2016.90

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. *Neural Computation*, *9*, 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Huang, G., Liu, Z., Van Der Maaten, L., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In *2017 IEEE conference on computer vision and pattern recognition (CVPR)* (pp. 2261–2269). https://doi.org/10.1109/CVPR.2017.243

Huo, R., Gao, Q., Qi, J., & Ju, Z. (2023). 3D human pose estimation in video for human-computer/robot interaction. In H. Yang, H. Liu, J. Zou, Z. Yin, L. Liu, G. Yang, X. Ouyang, & Z. Wang (Eds.), *Intelligent robotics and applications* (pp. 176–187). Singapore: Springer Nature Singapore.

Jia, Y., Wang, J., Shou, W., Hosseini, M. R., & Bai, Y. (2023). Graph neural networks for construction applications. *Automation in Construction*, *154*, 104984. https://doi.org/10.1016/j.autcon.2023.104984

Jiang, F., Ma, L., Broyd, T., Chen, K., & Luo, H. (2022). Underpass clearance checking in highway widening projects using digital twins. *Automation in Construction*, *141*, 104406. https://doi.org/10.1016/J.AUTCON.2022.104406

Kassem, M., Mahamedi, E., Rogage, K., Duffy, K., & Huntingdon, J. (2021). Measuring and benchmarking the productivity of excavators in infrastructure projects: A deep neural network approach. *Automation in Construction*, *124*, 103532. https://doi.org/10.1016/j.autcon.2020.103532

Kersting, K., Kriege, N. M., Morris, C., Mutzel, P., & Neumann, M. (2016). Benchmark data sets for graph kernels.

Kim, J., & Chi, S. (2019). Action recognition of earthmoving excavators based on sequential pattern analysis of visual features and operation cycles. *Automation in Construction*, *104*, 255–264. https://doi.org/10.1016/j.autcon.2019.03.025

Kim, J., & Chi, S. (2022). Graph neural network-based propagation effects modeling for detecting visual relationships among construction resources. *Automation in Construction*, *141*, 104443. https://doi.org/10.1016/J.AUTCON.2022.104443

Kim, S., Hong, S. H., Kim, H., Lee, M., & Hwang, S. (2023). Small object detection (SOD) system for comprehensive construction site safety monitoring. *Automation in Construction*, *156*, 105103. https://doi.org/10.1016/j.autcon.2023.105103

Kipf, T. N., & Welling, M. (2017). Semi-supervised classification with graph convolutional networks.

Li, G., Müller, M., Thabet, A., & Ghanem, B. (2019). DeepGCNs: Can GCNs go as deep as CNNs? In *2019 IEEE/CVF international conference on computer vision (ICCV)* (pp. 9266–9275). https://doi.org/10.1109/ICCV.2019.00936

Lingard, H., Wakefield, R., & Blismas, N. (2013). If you cannot measure it, you cannot improve it: Measuring health and safety performance in the construction industry. In *the 19th triennial CIB world building congress, Queensland university of technology, Brisbane, Queensland, Australia*.

Lundeen, K. M., Dong, S., Fredricks, N., Akula, M., Seo, J., & Kamat, V. R. (2016). Optical marker-based end effector pose estimation for articulated excavators. *Automation in Construction*, *65*, 51–64. https://doi.org/10.1016/j.autcon.2016.02.003

Luo, H., Wang, M., Wong, P. K.-Y., Tang, J., & Cheng, J. C. P. (2021). Vision-based pose forecasting of construction equipment for monitoring construction site safety. In E. Toledo Santos, & S. Scheer (Eds.), *Proceedings of the 18th international conference on computing in civil and building engineering* (pp. 1127–1138). Cham: Springer International Publishing.

Luo, X., Li, H., Cao, D., Yu, Y., Yang, X., & Huang, T. (2018). Towards efficient and objective work sampling: Recognizing workers' activities in site surveillance videos with two-stream convolutional networks. *Automation in Construction*, *94*, 360–370. https://doi.org/10.1016/j.autcon.2018.07.011

Luo, X., Li, H., Yang, X., Yu, Y., & Cao, D. (2019). Capturing and understanding workers activities in far-field surveillance videos with deep action recognition and Bayesian nonparametric learning. *Computer-Aided Civil and Infrastructure Engineering*, *34*(4), 333–351. https://doi.org/10.1111/mice.12419

McNally, W., Vats, K., Wong, A., & McPhee, J. (2021). Rethinking keypoint representations: Modeling keypoints and poses as objects for multi-person human pose estimation. arXiv preprint arXiv:2111.08557

Meng, L., Peng, Z., Zhou, J., Zhang, J., Lu, Z., Baumann, A., & Du, Y. (2020). Real-time detection of ground objects based on unmanned aerial vehicle remote sensing with deep learning: Application in excavator detection for pipeline safety. *Remote Sensing*, *12*(1). https://doi.org/10.3390/rs12010182

Müller, L., Galkin, M., Morris, C., & Rampášek, L. (2023). Attending to graph transformers. arXiv preprint arXiv:2302.04181

Nguyen, D. Q., Nguyen, T. D., & Phung, D. (2022). Universal graph transformer self-attention networks. In *Companion proceedings of the web conference 2022 (WWW '22 companion)*.

Niepert, M., Ahmed, M., & Kutzkov, K. (2016). Learning convolutional neural networks for graphs. In *Proceedings of the 33rd international conference on international conference on machine learning - volume 48* ICML'16 (p. 2014–2023). JMLR.org.

Pham, H. T. T. L., & Han, S. (2024). Generating realistic training images from synthetic data for excavator pose estimation. *Automation in Construction*, *167*, 105718. https://doi.org/10.1016/j.autcon.2024.105718

Pregnolato, M., Gunner, S., Voyagaki, E., De Risi, R., Carhart, N., Gavriel, G., Tully, P., Tryfonas, T., Macdonald, J., & Taylor, C. (2022). Towards civil engineering 4.0: Concept, workflow and application of digital twins for existing infrastructure. *Automation in Construction*, *141*, 104421. https://doi.org/10.1016/J.AUTCON.2022.104421

Rao, A. S., Radanovic, M., Liu, Y., Hu, S., Fang, Y., Khoshelham, K., Palaniswami, M., & Ngo, T. (2022). Real-time monitoring of construction sites: Sensors, methods, and applications. *Automation in Construction*, *136*, 104099. https://doi.org/10.1016/j.autcon.2021.104099

Rasul, A., Seo, J., & Khajepour, A. (2021). Development of integrative methodologies for effective excavation progress monitoring. *Sensors*, *21*(2). https://doi.org/10.3390/s21020364

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement.

Reis, D., Kupec, J., Hong, J., & Daoudi, A. (2023). Real-time flying object detection with YOLOv8.

Ren, H., Lu, W., Xiao, Y., Chang, X., Wang, X., Dong, Z., & Fang, D. (2022). Graph convolutional networks in language and vision: A survey. *Knowledge-Based Systems*, *251*, 109250. https://doi.org/10.1016/j.knosys.2022.109250

Roberts, D., & Golparvar-Fard, M. (2019). End-to-end vision-based detection, tracking and activity analysis of earthmoving equipment filmed at ground level. *Automation in Construction*, *105*, 102811. https://doi.org/10.1016/j.autcon.2019.04.006

Shin, Y., Seo, S., & Koo, C. (2025). Deep learning-based automated method for enhancing excavator activity recognition in far-field construction site surveillance videos. *Automation in Construction*, *173*, 106099. https://doi.org/10.1016/j.autcon.2025.106099

Tang, J., Wang, M., Luo, H., Wong, P. K.-Y., Zhang, X., Chen, W., & Cheng, J. C. P. (2023). Full-body pose estimation for excavators based on data fusion of multiple onboard sensors. *Automation in Construction*, *147*, 104694. https://doi.org/10.1016/j.autcon.2022.104694

Toivonen, H., Srinivasan, A., King, R., Kramer, S., & Helma, C. (2003). Statistical evaluation of the predictive toxicology challenge 2000–2001. *Bioinformatics (Oxford, England)*, *19*, 1183–1193. https://doi.org/10.1093/bioinformatics/btg130

Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., & Bengio, Y. (2018). Graph attention networks.

Wang, S., Yang, L., Zhang, Z., & Zhao, Y. (2024). Keypoints-based heterogeneous graph convolutional networks for construction. *Expert Systems with Applications*, *237*, 121525. https://doi.org/10.1016/j.eswa.2023.121525

Xu, K., Hu, W., Leskovec, J., & Jegelka, S. (2019). How powerful are graph neural networks? In *International conference on learning representations*.

Yanardag, P., & Vishwanathan, S. V. N. (2015). Deep graph kernels. In *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining* KDD '15 (p. 1365–1374). New York, NY, USA: Association for Computing Machinery. https://doi.org/10.1145/2783258.2783417

Yang, H. (2023). GatedGCN with graphsage to solve traveling salesman problem. In *International conference on artificial neural networks* (pp. 377–387). Springer.

Zhang, H., Li, M., Miao, D., Pedrycz, W., Wang, Z., & Jiang, M. (2023). Construction of a feature enhancement network for small object detection. *Pattern Recognition*, *143*, 109801.

Zhang, J., & Meng, L. (2019). Gresnet: Graph residual network for reviving deep gnns from suspended animation. CoRR, abs/1909.05729

Zhang, M., Cui, Z., Neumann, M., & Chen, Y. (2018). An end-to-end deep learning architecture for graph classification. In *Proceedings of the thirty-second AAAI conference on artificial intelligence and thirtieth innovative applications of artificial intelligence conference and eighth AAAI symposium on educational advances in artificial intelligence* AAAI'18/IAAI'18/EAAI'18. AAAI Press.

https://dspace.org/

A robust machinery action recognition in construction using a mixed graph convolution block

Wang, Shuozhi; Zhao, Yitian; Zhang, Zichao; Zhao, Yifan

2025-10-01
Elsevier
https://doi.org/10.1016/j.eswa.2025.128540
Wang S, Zhao Y, Zhang Z, Zhao Y. (2025) A robust machinery action recognition in construction using a mixed graph convolution block. Expert Systems with Applications, Volume 291, October 2025, Article number 128540

Downloaded from a DSpace repository. https://dspace.org/