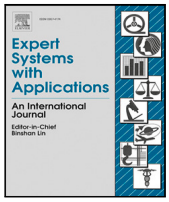
Expert Systems With Applications 237 (2024) 121525

Contents lists available at ScienceDirect

Expert Systems With Applications

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



Keypoints-based Heterogeneous Graph Convolutional Networks for construction

Shuozhi Wang, Lichao Yang, Zichao Zhang, Yifan Zhao *

School of Aerospace, Transport and Manufacturing, Cranfield University, Bedfordshire MK43 0AL, UK

ARTICLE INFO

Keywords:
Activity classification
Computer vision
Graph convolutional networks
Keypoint extraction

ABSTRACT

Artificial intelligence algorithms employed for classifying excavator-related activities predominantly rely on sensors embedded within individual machinery or computer vision (CV) techniques encompassing a large scene. The existing CV-based methods are often difficult to tackle an image including multiple excavators and other cooperating machinery. This study presents a novel framework tailored to the classification of excavator activities, accounting for both the excavator itself and the dumpers collaborating with the excavator during operations. Distinct from most existing related studies, this method centres on the transformed heterogeneous graph data constructed using the keypoints of all cooperating machinery extracted from an image. The resulting model leverages the relationships between the mechanical components of an excavator in varying activation states and the associations between the excavator and the collaborating machinery. The framework commences with a novel definition of keypoints representing different machinery relevant to the targetted activities. A customised Machinery Keypoint R-CNN method is then developed to extract these keypoints, forming the basis of graph notes. By considering the type, attribute and edge of nodes, a Heterogeneous Graph Convolutional Network is finally utilised for activity recognition. The results suggest that the proposed framework can effectively predict earthwork activities (with an accuracy of up to 97.5%) when the image encompasses multiple excavators and cooperating machinery. This solution holds promising potential for the automated measurement and management of earthwork productivity within the construction industry. Code and data are available at: https://github.com/gillesflash/Keypoints-Based-Heterogeneous-Graph-Convolutional-Networks.git.

1. Introduction

According to the 'Productivity in the construction industry, UK' from Census 2021 (Martin, 2021), average productivity levels in the construction industry have remained consistently below the UK average. Moreover, although the expenditure, skills and capital investment in this industry are growing, the productivity is not showing equal growth. This presents a lower conversion of resources to outputs, which requires more efficient resource allocation and management. Earthworks, crucial and resource-intensive components of construction projects, involve shaping target areas using mechanical equipment, and require optimisation to address the increasing demand for higher productivity and safety in infrastructure development (Parente et al., 2015). Building information modelling (BIM) (Eastman, 1974), a holistic process of creating and managing information about an asset to be built, has been growing to address this challenge. As a new technology that overturns the way to manage a construction project, BIM allows the team to capture and visualise the data they create during the process to benefit the coordination of operation and maintenance activities. However, most practical applications of BIM are currently concentrated in the design phase, and not always in the operations and maintenance phase (Lu et al., 2020). To assess specific construction progress and provide feedback to site managers, the current practices usually require people to update the information in this phase manually (e.g., machinery activity or weather), which is time-consuming, expensive, and error-prone work (Gong & Caldas, 2010; Kim, Ahn et al., 2018).

The application of digital twins on construction sites to accurately reconstruct and visualise building-related assets is a popular research topic. There are three major directions of digital twins in the construction phase. First, digital twins are developed and applied on infrastructures such as highways and bridges (Broo et al., 2022; Chiachío et al., 2022; Jiang et al., 2022a; Pregnolato et al., 2022). They provide a multidimensional view of how assets are designed and executed at the construction site, including staff behaviour, vehicle work

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

https://doi.org/10.1016/j.eswa.2023.121525
Received 6 May 2023; Received in revised form 29 August 2023; Accepted 7 September 2023
Available online 19 September 2023

phases, and space utilisation (Batty, 2018; Grieves, 2015). More specifically, Jiang et al. (2022b) designed and implemented a digital twin approach for highways only based on existing map data and then tested it in a section of the A1(M) motorway in the UK. They applied this digital twin for clearance checking on underpass roads in a road widening project. Pregnolato et al. (2022) developed a digital twins-based workflow for an existing asset in the built environment and tested it on the Clifton Suspension Bridge in Bristol (UK). In addition, it provides a means to test simulated scenarios, including the impact of design changes, weather disturbances, and safety events (Broo et al., 2022). As the second direction, the study of digital twins has focused on the analysis of human activities. Han and Lee (2013) developed a two-dimensional human skeleton detection, which could be reconstructed into three dimensions for improving safety management. Li et al. (2015) collected the information from the sensors on personal protective equipment to supervise the people's position and behaviour for training, analysing and evaluating for safety enhancement. Earthwork is one of the most essential and fundamental phases in most construction projects. Opoku et al. (2021) suggested that although digital twin technology has been used to assess the structural integrity of construction-related objects during the construction phase, there is a relative lack of real-time modelling and analysis of detailed processes. The application of digital twin technology to dynamic objects, such as machinery, holds significant promise in enabling potential real-time interventions to improve operational behaviours or fleet management.

As a typical earthwork machine, excavators are well used due to their advantages in flexibility and adaptability (He & Jiang, 2018). According to the type of data used, there are two major methods to estimate excavators' activity, either based on sensors in an individual machine or external cameras covering a large scene. For the first approach, data are collected by specific electronic sensors such as Bluetooth (Park et al., 2017), tape recorders (Akhavian & Behzadan, 2016; Cheng et al., 2017), inertial measurement units(IMU) (Kim, Ahn et al., 2018), etc. Based on this type of data, Rashid and Louis (2019) presented a recurrent neural network (RNN) based model for data augmentation for generating synthetic time-series training data about sensors on excavators. Bae et al. (2019) applied dynamic time warping (DTW) algorithm to determine similarities between reference signals to identify six excavator tasks. The research based on control signals Shi et al. (2021) adopted the Pulse-Width Modulation (PWM) technology and applied the Long-Short-Term Memory (LSTM) classifier, which produces an accuracy of 93.21% in pose estimation. Besides the information from internal machinery sensors, sound information are implemented within the construction site environment to facilitate the identification of equipment and tools present on site (Akbal et al., 2022; Scarpiniti et al., 2021). Although the accuracy of activity classification of the individual machines is high, one limitation of sensor-based approaches is the challenge of addressing group activities, which involve different types of machinery. Analysis of data from various machines could be difficult due to the difference in data format or sample rate etc. Furthermore, sensors could be costly, and the solution can have limited scalability. Computer vision (CV) methods are attractive in this application because image acquisition devices are less expensive while covering multiple machines at the same time.

In the CV-related studies, Lundeen et al. (2016) demonstrated the feasibility of marker-based sensor technology for excavator pose estimation. Soltani et al. (2017) used markers to recognise the parts of an excavator to extract the 2D skeleton of excavators based on videos. In the scope of deep learning methods, Kim, Chi et al. (2018) proposed a 2-stage (work or idle) algorithm based on the interrelationship between excavators and dumpers. Kim and Chi (2019) designed a Convolutional Neural Networks (CNN) and Double-layer LSTM for sequential pattern analysis. It was claimed that time information could significantly assist in the analysis of excavator movements. Roberts and Golparvar-Fard (2019) applied Hidden Markov Model to detect and track the excavators' activity based on a video dataset. Chen et al. (2020) applied a three-dimensional CNN on videos for classifying three excavator activities. Kim et al. (2021) combined the camera with the kinematic sensor as hybrid sensing to improve the recognition performance of three excavator activities. Additionally, a lightweight Fully Convolutional Network (FCN) was applied to achieve satisfactory accuracy with faster speed (Guo et al., 2022). Tang et al. (2023) proposed a data fusion strategy to utilise different types of onboard sensors for enhanced accuracy and robustness in full-body pose estimation of excavators. In the field of computer vision, two steps are typically involved in order to obtain the class of an action, namely feature extraction and classification based on the extracted features. There are some state-of-the-art in generic classification tasks, such as Resnet (He et al., 2015) and Vison Transformer (Dosovitskiy et al., 2020), could be also regarded as this process to some extents. Although these methods excel in classifying images, they usually are not straightforward to determine actions from a single image containing multiple object. Furthermore, certain one-stage action recognition models based on videos, such as ActionFormer (Zhang et al., 2022) and TriDet (Shi et al., 2023), are also difficult to output different actions based on different objects.

In modern construction sites, some specific tasks usually involve multiple types of machinery and multiple vehicles of the same type. Therefore, analysing the interaction between vehicles is beneficial to determine the work phase correctly. Kim, Chi et al. (2018) applied Tracking-Learning-Detection (TLD) to track heavy equipment and then analysed their interactions using a knowledge-based system based on the distance between the excavator and the dumpers. Focusing on the relationship of objects, Kim and Chi (2022) introduced a graph neural network-based model to detect entangled and intertwined visual relationships. The limitation of this approach is that it ignores the relationship between operational parts in one type of machinery (e.g., excavator), which could be necessary to accurately assess the productivity of such a vehicle (Sato et al., 2022).

The main research gap is that current activity identification methods for construction machinery either ignore the relationship between excavators and corporation machinery or the relationship among operational parts in individual machinery. Moreover, there is difference between the relationship between the joints on the same machinery and the relationship between different vehicles. Therefore, it is more logical to model the different relationships separately. This approach will also provide more information to improve the accuracy of the AI models than treat all relationships as the same.

In this study, the hypothesis is that the pose of machinery, which can be represented by a limited number of keypoints, is sufficient to determine the type of earthwork activity. Furthermore, it is assumed that the interaction between the excavator and cooperating dump trucks contributes to improving the performance of the classification model. This article proposes an end-to-end framework, called Construction Equipment Keypoints detection and Heterogeneous Graph Convolution Networks (CEKP-HGCNs), for classifying excavator-related machinery's activity. The keypoint-based 2D human pose estimation (Cao et al., 2021; Fang et al., 2017; Tompson et al., 2015; Wei et al., 2016) inspires the proposed framework. Next, the keypoints are grouped, satisfying that there is only one excavator in the group with all dumpers in the whole image. These groupings are then converted into heterogeneous graph structure data. Besides the node attribute (location and label), each group (or graph) also includes three types of relationship, which are 'relationship between keypoints on excavators', 'relationship between keypoints on dumpers' and 'relationship between keypoints on target excavator and dumpers'. Next, the graph dataset is applied to classify the actions via a heterogeneous graphical neural network.

The novelty of this work is highlighted by the introduction of a feature engineering approach to extract keypoints of machinery, which represent pose features, and the development of a customised heterogeneous graph neural network for activity classification using these features. Another distinctive aspect is that, unlike existing methods that often only focus on excavators, the proposed solution takes into

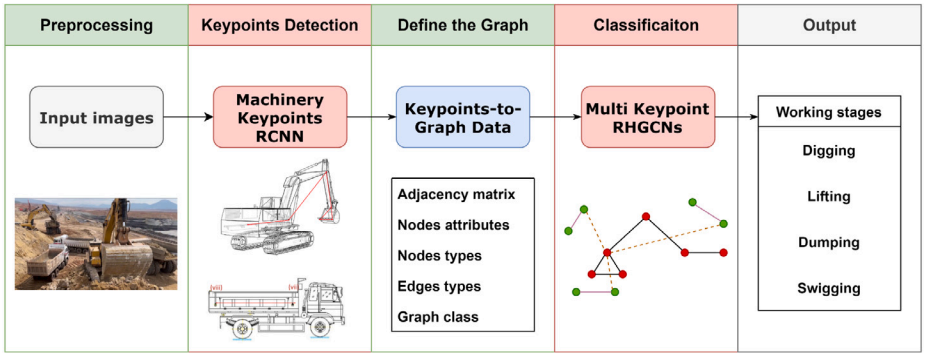


Fig. 1. The architecture of the proposed CEKP-HGCNs framework. It consists of four stages: (1) preprocessing image including data augmentation; (2) extraction of keypoints of all machinery in the image using a Machinery Keypoints RCNN model; (3) construction of graph data by assigning the type and attribute of nodes, the type of edges between notes of the same machinery and edges between notes from different machinery; (4) classification of earthwork activity based on the graph data using a Multi Keypoint RHGCNs model.

account other cooperating machinery to achieve more precise activity recognition. In contrast to state-of-the-art approaches that usually focus on image pixels, the proposed solution tends to be more transparent and more adaptive to variations of data capture environment, occlusion, and machinery models. The contribution of this study includes:

- Definition of keypoints and relationships among them to represent the pose of excavators and dumpers, as well as an extraction algorithm called Keypoint-RCNN.
- A Relationship Heterogeneous Graphical Convolutional Networks to analyse the relationships between the moving parts of an excavator and dumpers, which not only improves the performance of activity classification by grouping machinery, but also allows to recognise multiple group activities in a single image.

Fig. 2. The four investigated working stages/activities involved an excavator. The involvement of dumper is not essential.

2. Methodology

2.1. Overview of methodology

This paper introduces a novel classification framework, named Construction Equipment Keypoints Detection and Heterogeneous Graph Convolution Networks (CEKP-HGCNs), to recognise the activity of excavators involving multi-vehicles based on a keypoint detection algorithm and heterogeneous graph convolution networks (HGCNs). It should be noted that this method accommodates the activity of a single excavator without other vehicles. The proposed framework is illustrated in Fig. 1. First, a keypoint extraction model predicts the two-dimensional locations and labels of carefully defined keypoints for each excavator and dumper in the image. In this part, a keypoint detection algorithm for excavators and dumpers, named machinery keypoint-RCNN, is tailored and carried out to extract the keypoints of mechanical components. The output of machinery keypoint-RCNN is the type of the point and its pixels location X_i and Y_i. Six keypoints represent an excavator, while two keypoints represent a dumper. After obtaining the point location and label, we can transform the outcomes into Graph data, which includes adjacency matrix, nodes attributes and types, edges types and graph class. Next, this Graph data is passed through novel HGCNs to identify the activity. The detail of each step is presented below.

2.2. Machinery keypoints extraction

2.2.1. Machinery keypoints definition

Typical excavator's activity suggested in existing literature (Feng et al., 2016; Kim & Chi, 2019; Roberts & Golparvar-Fard, 2019; Shi et al., 2021, 2020) consist of digging, lifting, dumping, and swinging according to the tasks performed by mechanical components of the excavation, examples of which are shown in Fig. 2.

According to the major mechanical components and aiming to minimise the number of total keypoints, this paper defines six keypoints to describe the pose and activity of excavators, which are (i) body end; (ii) cab boom; (iii) boom arm; (iv) arm bucket; (v) left bucket end; (vi) right bucket end, as shown in Fig. 3. To be more specific, the points (i) and (ii) describe the direction of the excavator body; the pairs of (ii)–(iii) and (iii)–(iv) describe the poses of the boom and arm, respectively; the points (iv), (v) and (vi) could not only describe the pose and position of the bucket but also could describe the lateral orientation, which is beneficial for identifying body direction when the central axis is perpendicular to the image. Two keypoints are defined to represent a dumper: (vii) dumper body front: the middle point of the dumper body front, and (viii) dumper body end: the central point of the dumper body end. Compared to the existing methods using only one keypoint to represent a dumper (Luo et al., 2020), two keypoints can measure the direction and potentially describe an activity in more detail, such as unloading.

2.2.2. Machinery keypoint RCNN

In real-world applications where images are captured over a long working distance, the existing non-deep learning-based methods either identify key components of a vehicle by detecting artificial markers (Soltani et al., 2017) or extract the keypoints with a large margin of error due to a relatively small region of interest on the whole image. This paper proposes to use a deep learning-based method, machinery keypoint RCNN, to address this challenge. Established upon Mask-RCNN (He et al., 2017), Keypoint-RCNN has been used for human pose estimation but not for the keypoint extraction of excavators and dumpers. Mask-RCNN is widely implemented as a classic two-stage

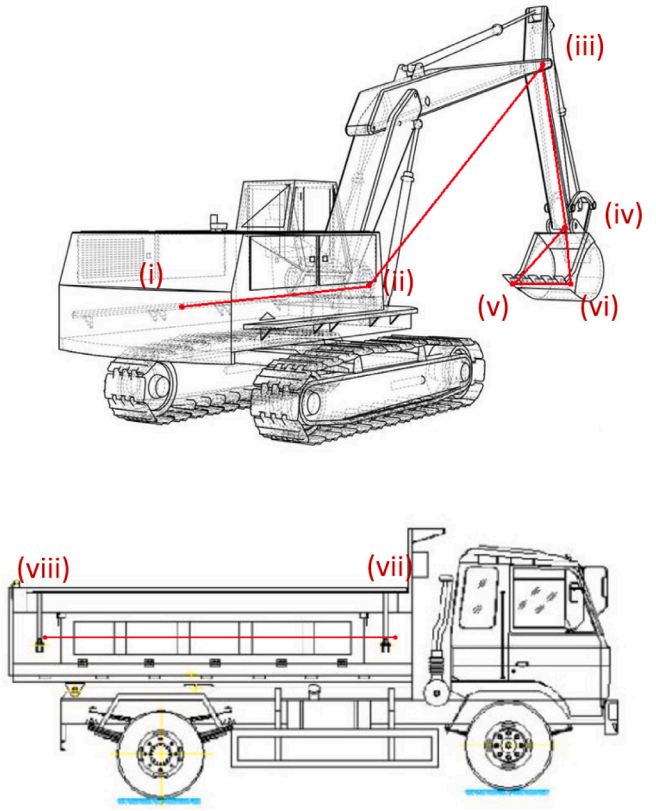


Fig. 3. The defined keypoints for an excavator and a dumper, (i) excavator's body end; (ii) excavator's cab boom; (iii) excavator's boom arm; (iv) excavator's arm bucket; (v) excavator's left bucket end; (vi) excavator's right bucket end; (vii) dumper body front and (viii) dumper body end.

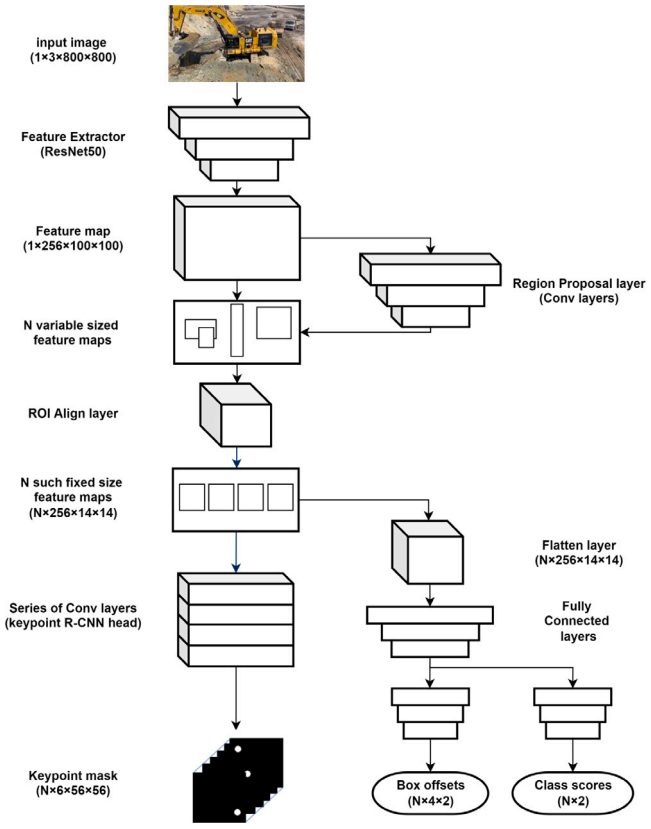


Fig. 4. The architecture of the machinery keypoint-RCNN for excavators, where N is the number of objects proposed by the Region-Proposal Layer. In this study, Resnet-50 is used as the backbone.

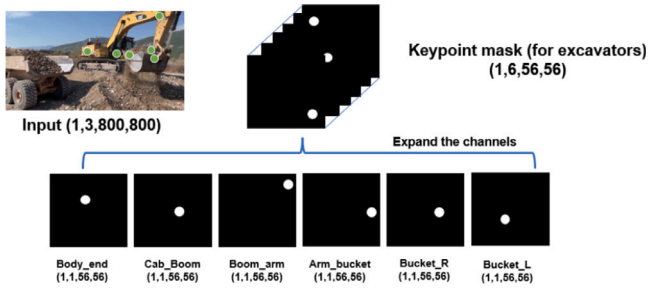


Fig. 5. An example for encoding the machinery keypoints in the output mask on an excavator. There are 6 (K=6) one-hot 56 × 56 binary masks, each of which has only a single pixel labelled as the object.

object detection model. The differences between Keypoint-RCNN and Mask-RCNN are the output size and the method of encoding keypoints in the mask. As shown in Fig. 4, the input image of machinery keypoint RCNN is resized as $1 \times 3 \times 800 \times 800$. The machinery keypoint RCNN model is built on the top of the feature extractor based on Feature Pyramid Network (Lin et al., 2016) (FPN), which is for fusing feature maps at multiple scales to preserve information at various levels. The *Region Proposal Layer* predicts the approximate location of N number of objects detected in the feature map. It is followed by a Region of Interest (ROI) Align Layer, which is an operation for extracting a small feature map from each ROI in detection and segmentation-based tasks (He et al., 2017). Instead of applying an ROI-Pooling layer in Faster-RCNN (Ren et al., 2015), ROI Align could solve the problem of quantisation of the ROI boundaries. The output of the ROI Align Layer is passed to two branches: a branch consisting of a flatten layer and a series of fully connected layers, and a branch called the keypoint RCNN head. In the first branch, a fully connected Layer is split into two separate blocks: one is for predicting the class scores for the object and background, and the output size is $[N, 2]$; another block is for predicting the bounding-box coordinates for the object, and the output size is $[N, 4 \times 2]$. The second branch is a series of convolution layers with an output size of $[N, K, 56, 56]$, where K is the number of the keypoints (e.g., 6 for excavators and 2 for dumpers). The ground truth of keypoints is encoded as a one-hot structure. For each visible ground truth, a channel-wise Softmax function from the final feature map $[K, 56, 56]$ is used to minimise the cross-entropy loss. Fig. 5 presents an example of encoding the keypoints in the output mask on excavators. The keypoint mask is expanded to K channel, and each channel corresponds to a specific keypoint.

2.3. Heterogeneous Graph convolution networks (HGCNs) for activity classification

Graph Convolution Networks (GCNs) (Kipf & Welling, 2016) have become a widely applied technology in recent years. For the classification applications of GCNs, there are three primary levels, concluded by Zhou et al. (2020): node-level, edge-level and graph-level, where node-level focuses on nodes, trying to categorise nodes into several classes; edge-level focuses on classifying edge types or predict whether the edge is existing or not between the nodes; graph-level requires the model to learn the graph representations. This framework introduces a GCNs-based model at the graph level. Besides, different nodes and edges have been defined according to their actual physical meaning to carry out a more logical and rigorous analysis. Besides the level of the tasks, the type of the graph also requires definition. As presented by Zhou et al. (2020), according to whether the type of the nodes and

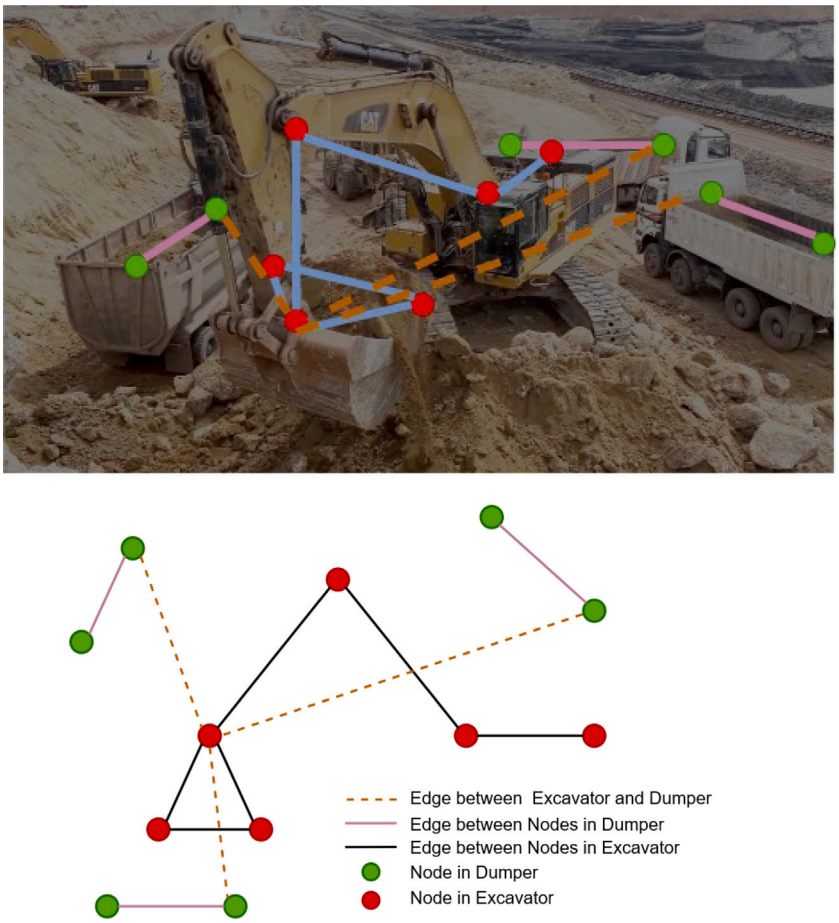


Fig. 6. An example for the definition of graph attributes overlaid on the image (top) and an abstract of the graph (bottom).

edges are the same, there are homogeneous graphs (with the same type of nodes and edge) and heterogeneous(nodes and edges have different types). In this study, the physical attributes of nodes on various machinery differ. Besides, the physical attributes of links between the nodes on the same machinery or cross-machinery are different. This means that the types of nodes and edges in the same graph are different. Therefore, the proposed graph is a heterogeneous graph representing multiple types of nodes or edges attributes. As shown in Fig. 7, the common practice is firstly to group nodes according to their types, and then perform graph convolution by each same type of node group and apply the graph readout function by aggregating over the nodes of different types. The final step is to perform the soft classification.

2.3.1. Graph attributes definition

As a data structure, Graph $G = (V, E, R)$ consists of Nodes V, Edges E, and relation types R. In this project, the nodes are keypoints detected by the machinery keypoints RCNN, and each keypoint's label and position are applied as Node Type V_{type} and Node Attribute V_{attr}. There are two types of nodes: nodes in excavators and nodes in dumpers. For the definition of the edge, logically, the connection between keypoint pairs on the same machinery differs from the connection between keypoints on different machinery. The graph has three edges: edges between the nodes in the same excavator, the edges between the nodes in the same dumper, and the edges between a node in the excavator and a node in the dumper. Fig. 6 shows an example of graph data in this study. It should be noticed that the nodes and edges from the same machinery use the same colour. In this study, the edge between the point of the excavator's arm bucket (point iv in Fig. 3) and the point of the dumper body front(the point vii in Fig. 3) is defined as the 'cross vehicles connection'.

2.3.2. Heterogeneous Graph Convolution Networks (HGCNs)

This project defines the task as a heterogeneous graph classification task. After applying mini-batching, the proposed classification framework contains, Relational-GCNs layers, activation layers, readout layer, dropout and classifier. In this model, the first step is to perform message passing on this batch of graphs by updating the features of nodes or edges. Next, aggregate the features of the same nodes or edges in the graph, and then Aggregate different types of nodes and edges in the readout function. In the final step, classify the graph based on graph-level representations. The general structure of the Relationship Heterogeneous Graph Convolution Networks (HGCNs) model is shown in the algorithm 1, where the input: V_{attr} is the node attributes (keypoint's location); G_{label} is the graph label; E_{type} is the edge type; v_{type} is the node type, and A is the adjacency matrix which represents the adjacency between nodes.

This HGCNs-based model for graph classification follows three stages: 1. Embedding nodes by performing message-passing roundly; 2. Executing the Readout layer, which aggregates node embedding into a unified graph embedding; 3. Training a final classifier on the graph embedding.

This HGCNs-based model includes graphs convolution layers in the embedding nodes step, followed by the **ReLU** activation. In the Readout layer, Henaff et al. (2015) suggested that performing max or mean pooling is essential to reduce the dimensionality in the graph domain, and the mean pooling is selected as it is the most common one in existing HGCNs-based models. The last parts are a dropout process and a linear classifier. We conducted tests on the selection of the number of HGCN layers, specifically testing 2, 3, and 4 layers. No significant difference was observed among these options. Consequently, according

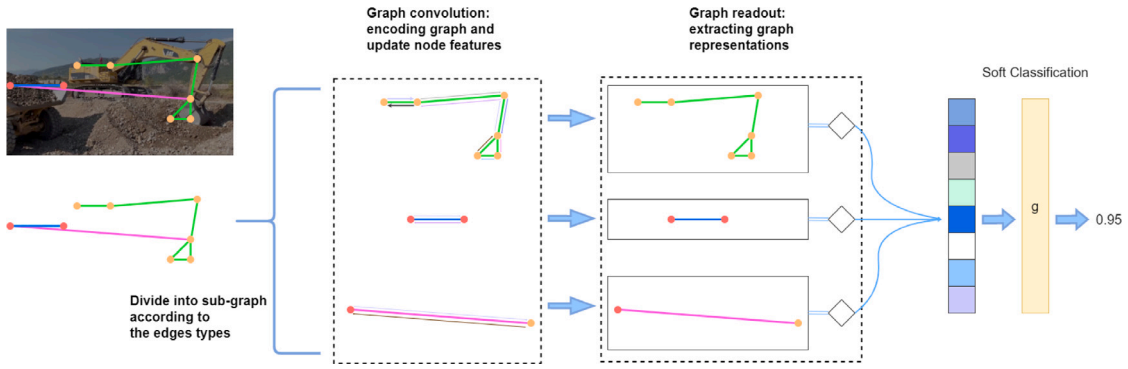


Fig. 7. An example for the graph classification pipeline involving 4 stages. From left to right, (1) dividing the input batched graphs into sub-graph according to the edges types; (2) performing message passing on each type of edge to update features; (3) aggregating over the edges of different types in the readout function; (4) classifying graphs based on graph-level representations.

Algorithm 1 Graph-level Convolution Network Classifier

```
Random Shuffle Graph
Data splitting
Mini-batching of graphs
for graph in batch do
    for nodes, edges with same types do
        Graph convolution
        Relu
    end for
    Readout(average)
    Dropout layer
    Linear layer
end for
```

to the research (Li et al., 2018), a structure with two HGCN layers has been employed to achieve a relatively low computational complexity. Additionally, the hidden layer consisted of 256 neurons.

Relational Graph Convolution Network

Motivated by messages-passing architectures (Gilmer et al., 2017), Schlichtkrull et al. (2017) define the propagation model for computing the forward update of an entity:

$$h_i^{(l+1)} = \sigma\Big(\sum_{r\in R}\sum_{j\in N_i^r}\frac{1}{c_{i,r}}W_r^{(l)}h_j^{(l)} + W_0^{(l)}h_i^{(l)}\Big), \tag{1}$$

where $h_i^{(l)}$ is hidden state of node v_i in the layer l of the neural network. N_i^r describes the set of neighbour indices of node i with the relation $r \in R$, $c_{i,r}$ is a normalisation constant. $W_r^{(l)}$ is a linear transformation function that uses a parameter matrix to transform the neighbour nodes of the same edge type. σ is the activation function, where we apply the ReLU activation ($f(x) = max(0, x)$). Intuitively, the function (1) accumulates the transformed feature vectors by a normalised sum. Unlike the GCNs, the relation-specific transformations depend on the type and orientation of the nodes and edges. And this function describes that the different type of relationship is aggregated separately. It is noticed that a single self-connection of a special relation type to each node is added to the representation of the node at layer $l + 1$ could be informed by the corresponding representation at layer l.

To avoid the number of parameters proliferating due to applying a function (1), Schlichtkrull et al. (2017) provides two methods for weight regularisation, which are *basis-decomposition* and *block-diagonal-decomposition*. To alleviate the overfitting problem, the *basis-decomposition* is carried out in this project, each weight $W_r^{(l)}$ is defined as the following function:

$$W_r^{(l)} = \sum_{b=1}^{B} a_{rb}^{(l)} V_b^{(l)}, \tag{2}$$

where $V_b^{(l)} \in \mathbb{R}^{d^{(l+1)}\times d^{(l)}}$, and B is a hyperparameter to control the number of $V_b^{(l)}$. According to the function (2), for different types of relation r, its parameter matrix $W_r^{(l)}$ is linear combination of $V_b^{(l)}$ and coefficient $a_{rb}^{(l)}$. Therefore only $a_{rb}^{(l)}$ is related to the relation type r. Meanwhile, effective weight sharing between different relation types is achieved for all $V_b^{(l)}$, which could be trained on data with frequent relations. Because the sparse relation matrix is composed of $V_b^{(l)}$ shared parameters, this regularisation method is beneficial for alleviating the overfitting problem.

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 operation is called 'Graph Readout'. Common aggregation methods include summing over all node or edge features, averaging, and finding the maximum or minimum value element by element. In this model, we carry out averaging as an aggregation method. Given a graph g, the average node feature readout could be defined as:

$$h_g = \frac{1}{|V|}\sum_{v\in V} h_v, \tag{3}$$

where h_g is the representation of the graph, V is the set of the nodes, h_v is the feature of certain node. In this model, once the h_g is available, the data is passed through a $Dropout\ layer$ and $Linear$ layer for classification output. The $loss\ function$, L, is cross entropy and can be written as:

$$L = -\sum_{c=1}^{M} y_{o,c}\log(p_{o,c}), \tag{4}$$

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

2.4. Evaluation methods

This framework includes two learning-based models: machinery keypoint RCNN and multi-keypoints HGCN. To evaluate the Keypoint detection model, we used Object keypoint similarity (OKS), defined by COCO. It applies the mean average precision (AP) over 10 OKS thresholds as a standard evaluation metric that considers the Euclidean distance and the scale effect (Anon, 2016). The OKS plays a similar role as the IoU in object detection. It is calculated from the scale and the distance between predicted and ground truth points. We usually refer to the AP as the quantitative evaluation standard.

Four different standards are carried out to evaluate the performance of the HGCN classification model in the identification of the activity of excavators. Accuracy represents the ratio of correctly classified samples. Precision and recall describe the ratio of correctly predicted

and identified retrieved samples, respectively. Besides, precision and recall are interdependent, and the F1-score is the harmonic mean of these two metrics. The calculation formulas are as follows (Koo et al., 2019; Shi et al., 2021):

$$Accuracy = \frac{TP + TN}{TP + FP + FN + TN} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$Precision = \frac{TP}{TP + FP} \tag{7}$$

$$F - score = \frac{2 \times precision \times recall}{precision + recall} \tag{8}$$

3. Experiment and result

3.1. Experiment environment

To evaluate the performance of the proposed framework, we constructed an image dataset from six YouTube videos. Each video includes different numbers and types of excavators and dumpers. Besides, the construction site environment in each video is also different from the others. The total dataset for machinery keypoint RCNN includes 465 manually labelled images. Data augmentation was then applied through horizontal and vertical flips, as well as brightness adjusting ranging within ±30%. This resulted in an image dataset comprising a total of 1395 images. Roboflow was used as the image annotation tool, and the output is in the COCO format. Before resizing into $1 \times 3 \times 800 \times 800$, the original size of images is $1 \times 3 \times 1920 \times 1080$. For the HGCNs, the dataset was produced by a trained machinery keypoint RCNN, including 1185 graphs, divided into 5-fold for cross-validation (237 graphs in each fold). The value for each node attribute, representing the point location, is based on the orthogonal coordinate system, where the centre point is $[0, 0]$ according to the original image. We tried to make the proportion of each category of activity as even as possible (27.8% for Digging, 21.5% for Dumping, 29.5% for Lifting and 21.1% for swinging). The developed codes were performed on the Operation System of Ubuntu 20.04. The programming language is Python, and the deep learning environment is PyTorch 1.10. Deep Graph Library with CUDA 11.3 is the main library for HGCNs. CPU and GPU are Intel i9-9900k and Nvidia 3090, respectively. Both keypoint detection and classification networks were trained using the Adam optimiser, whose capability has been well demonstrated in Kingma and Ba (2017). The initial learning rate for both models was set as 0.001. It was adjusted dynamically based on the loss of the validation dataset. During the training process, early stopping was carried out to avoid over-fitting for both deep learning models. The patience was set as 50 epochs, which means the training would be stopped when the current validation loss is not less than the one in the last 50 epochs.

The graphs are batched before being fed into the HGCNs to reduce the computational cost so that the framework can work in near real-time. In the GCN, some typical procedures achieved by rescaling or padding each data into a set of the same shapes are hardly feasible or may lead to unnecessary memory consumption. To address this point, adjacency is stacked diagonally, and the node and object features are connected directly in the node dimension. This procedure has some crucial advantages over other batching approaches. Firstly, the Relational Graph convolution operators that rely on a message-passing scheme do not need to be modified since messages are not exchanged between two nodes that belong to different graphs. Secondly, there is no computational or memory overhead since adjacency matrices are saved in a sparse fashion holding only non-zero entries such as the edges.

Table 1
Model performance comparison of different keypoint extraction methods and different backbones.

Object	Methods	AP	$AP_{0.50}$	$AP_{0.75}$
Excavator	MobileNet (Sandler et al., 2018) backbone	0.814	0.873	0.827
	Hourglass (Liang et al., 2019)	0.827	0.913	0.871
	Proposed (Resnet18)	0.817	0.887	0.863
	Proposed (Resnet34)	0.831	0.925	0.879
	Proposed (Resnet50)	**0.838**	**0.935**	**0.897**
Dumper	MobileNet (Sandler et al., 2018) backbone	0.913	0.939	0.921
	Hourglass (Liang et al., 2019)	0.881	0.943	0.937
	Proposed (Resnet18)	0.933	0.974	0.954
	Proposed (Resnet34)	0.957	0.981	0.970
	Proposed (Resnet50)	**0.984**	**0.987**	**0.986**

Table 2
Performance comparison of different numbers of neurons.

Neurons	Accuracy
32	95.1%
64	95.3%
128	95.6%
256	97.5%
512	97.1%

3.2. Result and analysis

For the proposed machinery keypoint RCNN, Fig. 8 shows the visualisation of the mode output. It includes six scenes containing at least an excavator in operation. The objects and the mechanical keypoints are well identified in scenarios (1) and (5), which have a single excavator and dumper. In scenarios (3), (4) and (6), multiple excavators and dumpers are well identified and represented. Although sometimes the left and right ends of the digger bucket are obscured in the diagram, they still can be identified. Fig. 9 shows the prediction and ground truth of keypoints in one image. In general, the keypoints in the moving parts, i.e. from the boom to the buckets, were identified with no significant deviation. The predictions provide a more accurate representation of the location of the keypoints of the excavator's moving parts. Table 1 shows a performance comparison for each evaluation metric between the proposed keypoint detection model, methods based on stacked Hourglass networks (Liang et al., 2019) and applying MobileNet (Sandler et al., 2018) as backbone. Overall, the proposed model performs well on both excavators and dumpers. Hourglass performs better compared to using MobileNet as the backbone. In terms of accuracy, the state-of-the-art and the machinery keypoints RCNN performed well from the AP perspective, while the machinery keypoint RCNN improved AP by 0.011 and 0.103 on excavators and dumpers, respectively. The comparison of applying MobileNet (Sandler et al., 2018) and different layers of ResNet for the proposed machinery keypoint RCNN is also shown in Table 1. ResNet50 has exhibited the best performance for all matrices.

For the HGCNs model, Fig. 6 represents the visualisation of the HGCNs classification process. This figure suggests that one image could include not only one graph. As mentioned before, the graph is grouped according to only one excavator and all the dumpers in the image. Each graph contains different types of points and relationships. By performing the HGCNs model, the class of each graph is predicted. This indicates that the proposed framework has the ability to identify the class even though the image contains multiple excavators, which differs from the state-of-the-art. We first evaluated the performance using different numbers of neurons in the hidden layer. Various neuron counts were tested and an accuracy of approximately 95% was observed even when the number of neurons is as small as 32. After settling on 256 neurons, we found that increasing the number further did not yield significant improvement. Ultimately, we selected 256 neurons for the proposed model. Table 2 presents the results of this sensitivity test.

Fig. 8. The visualisation of the output of the machinery keypoint-RCNN model. Scenarios (1) and (5) contain an excavator and a dumper; Scenario (2) includes an excavator and multiple dumpers. Scenarios (3), (4) and (6) contain multiple excavators and multiple dumpers. In each example, the green rectangle describes the bounding box of excavators, and the red one describes that of dumpers. The green dots represent the keypoints of excavators, and the red dots represent the keypoints of dumpers.

Fig. 9. The comparison of the output of the machinery keypoint-RCNN model (orange dots) and ground truth (blue dots).

The training log of the HGCNs is shown in Fig. 11. The image shows that before 200 epochs, the loss of the model presents a remarkable decrease, while the training accuracy shows a corresponding increase. For the validation loss, the general tendency suggests the same increase even if there is a significant fluctuation. After training 200 epochs, the model converges with a training accuracy close to 98% and a validation loss close to 96%. The loss degree to 0.2 after 100 epochs remains to decrease until the end while the training and validation loss does not show a significant change. This figure suggests that after training epoch 200, the model could be regarded as convergence and able to produce an effective prediction. Besides, as shown in Fig. 10, when the ground truth of the keypoint is used as the training data for the HGCNs model instead of the machinery keypoint RCNN, the training log does not show a significant difference. This could be further illustrated by the fact that the output of mechanical key point detection also has a strong ability to represent mechanical poses.

Fig. 13 shows the Confusion Matrix, which is utilised for the performance evaluations of the methods after the classification. The total number of test datasets for HGCN is 237. Generally, the prediction accuracy is 97.5%. In terms of precision, the Digging class has the highest percentage, which is 98.5%. According to Fig. 13, the F-score for each class and weight F-score could be calculated, shown in Table 3. It is noticed that the digging class is not the most of the training dataset (as mentioned before, the training dataset and testing dataset are almost the same). However, the lifting class takes up the most part while having less precision than the digging class. This indicates that the performance for a particular category is less influenced by the amount of data in the corresponding class. Moreover, as shown in Table 3, the F-socre of dumping is the highest one even dumping class takes up 21.5% in the training set. This also means that each class that is not exactly equal in the training set will not significantly impact the model performance. In terms of recall, the lifting class owns the highest percentage of 98.6%. There are more mispredictions of swinging and digging, which are 4% and 3%. This is most likely caused by the multi-machinery scenario where the bucket's position is similar in both categories. To justify the selection of the heterogeneous graph, we conducted an additional sensitivity analysis that utilised a homogeneous graph, allowing only one type of edge. The training log of this test is displayed in Fig. 12, which suggests that the model exhibited slower convergence within 1000 epochs compared to using a heterogeneous graph, as shown in Fig. 11. Both the training and validation accuracy (around 80%) are significantly lower than that of the heterogeneous graph (>90%). This result suggests that heterogeneous graph datasets contain more information by considering different edge types, leading to better classification performance.

To further evaluate the performance of the proposed algorithm against state-of-the-art methods, we applied Vision Transformer (Dosovitskiy et al., 2020) (ViT), VGG-16 (Simonyan & Zisserman, 2015) and ResNet-34 (He et al., 2015) to the same set of images used in this research. Since these methods can only estimate one activity for

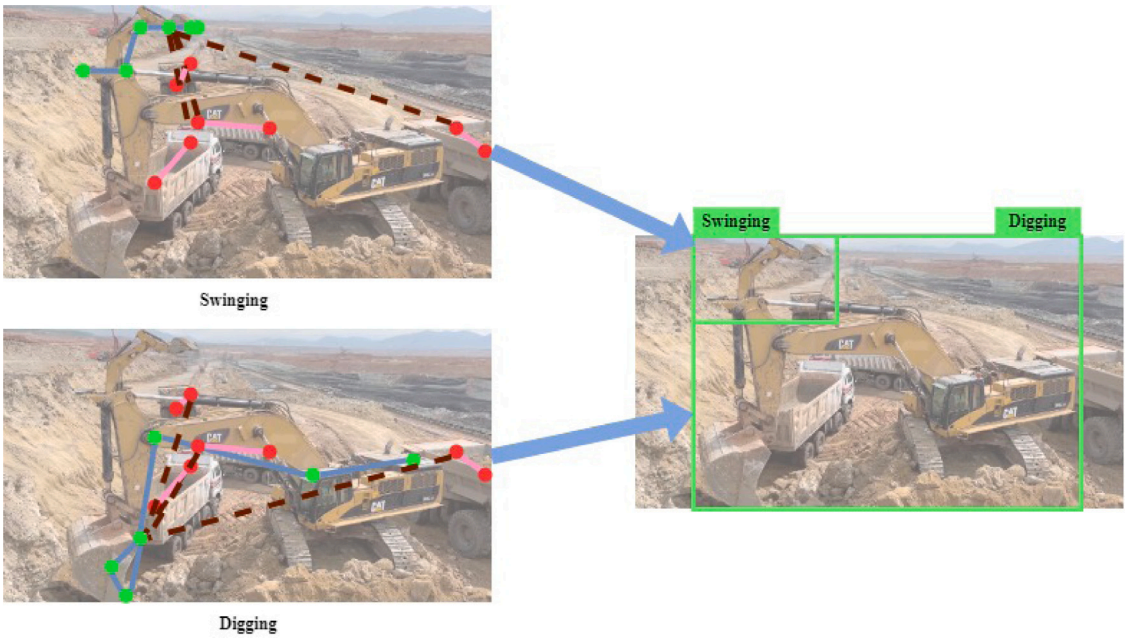


Fig. 10. The visual result of HGCNS classification. The green dots represent the joint on the excavators; the blue lines represent the link between the dots on the same excavator; the red dots represent the keypoints on the dumpers; the pink lines represent the link between the dots on the same dumper; the brown dashed lines describe the connection between the excavator and dumpers.

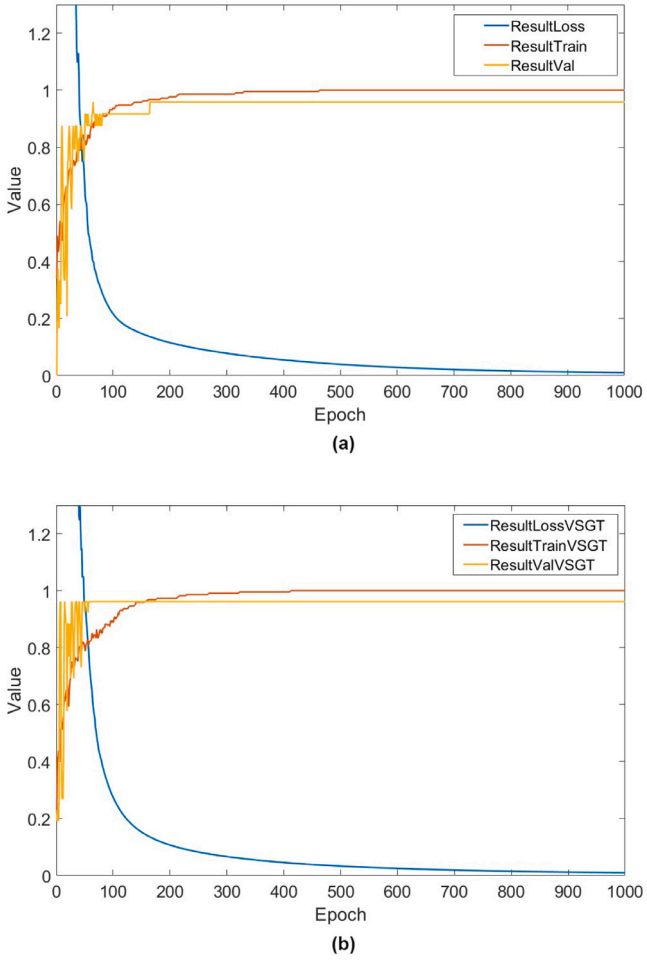


Fig. 11. Training log of the HGCNs model using (a) the output from the machinery keypoint-RCNN as the training dataset, (b) the Ground Truth as the training dataset.

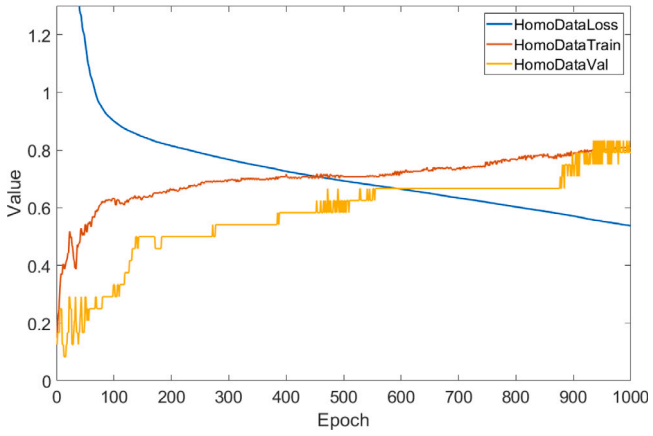


Fig. 12. Training log of applying homogeneous graph data that includes only one type of edge.

Table 3
F-score for each class and weight F-score.

Class	F-score
Digging	97.7%
Dumping	98.0%
Lifting	97.9%
Swinging	96.0%
weight recall	97.0%
weight precision	97.5%
weight F-score	97.2%

each image, the ground truth is determined by the activity of the excavator that occupies the majority of the image. The ground truth of the proposed method allows multiple activities to be assigned to a single image. For all methods, the training epoch was set to 300. Table 4 presents the accuracy and computational cost of the four tested methods. The proposed method achieved the highest accuracy (97.4%), followed by thw two CNN-based methods which perform

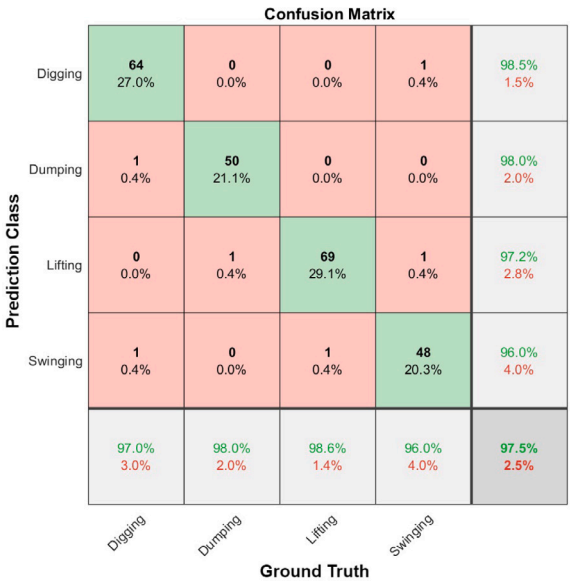


Fig. 13. The Confusion matrix of testing. The right column presents the Precision, and the bottom raw presents Recall. The right bottom box presents the overall accuracy.

Table 4
Classification accuracy and computational complexity of the proposed method in comparison to other state-of-the-art image-based classification models.

Method	Accuracy	Trainable Para	FLOPs
Resnet-34 (He et al., 2015)	95.4%	**21.5M**	**3.6B**
VGG-16 (Simonyan & Zisserman, 2015)	93.5%	138M	15.5B
ViT (Dosovitskiy et al., 2020)	81.8%	85M	17.6B
Proposed	**97.4%**	43M+0.4M	50B+5M

the second-best. In terms of computational cost, ReseNet-34 has the fewest trainable parameters, while the proposed method has the second fewest. Our method has the highest FLOPs while Resenet-34 has the fewest. It should be noted that the time cost of our method comprises two parts: keypoint extraction and activity classification. Keypoint extraction dominates the trainable parameters and FLOPs. To further test the performance in real applications, we applied the proposed model in a state-of-the-art edge computing platform, NVIDIA AGX Orin, and achieved 10 fps (frame per second).

4. Conclusion

To address the strong demand for the classification of excavator-involved activities, this paper reports a Heterogeneous Graph Convolutional Networks (HGCNs) based framework to infer the excavator's activity from the machinery keypoints extracted from an image. First, we introduced a machinery keypoint RCNN model to extract the location and type of the machinery keypoint. To treat the information on keypoint relationships located in the same machinery or across machines, we introduced a Heterogeneous Graph Convolutional Networks (HGCNs) model focusing on the processing and aggregating different types of keypoint attributes of action parts and the different relationships between them. With such a model, the relationship between the excavator and the cooperating machinery can be learned and referenced to improve the accuracy of the classification of excavator activities. The main research findings include: (a) the proposed machinery keypoint extraction model can predict all keypoints even when components are obscured; (b) the heterogeneous graph is a more appropriate choice than the homogeneous graph for this application, effectively encoding different types of notes and relationships; (c) the inclusion of notes and relationships with co-operating machinery can improve the classification accuracy; (d) the proposed solution can work effectively with limited training data.

It should be noted that the proposed solution has limitations. Firstly, the computational demand increases with the increment in the number of machinery in an image. Secondly, as a computer vision-based method, the working environment, such as working distance, illumination or weather, poses an inevitable threat to the image quality. A potential solution for future studies is the fusion of data from other types of sensors. Thirdly, the proposed framework is based on images rather than videos, which only considers spatial information while neglecting temporal information. This study considered that processing video data usually requires a large machine learning model and high computational resource, which may not be appropriate for edge-computing devices. However, in terms of the accuracy of activity recognition, video data can provide additional information that images alone cannot provide. In future work, our aim is to explore more intelligent forms of group synthesis graphs to further decrease the graph size and, consequently, reduce the computation time of the classification model. Furthermore, we plan to replace the existing keypoint detection method based on mechanical joints with a more efficient alternative. We will consider implementing a bottom-up approach to diminish the computational resources dedicated to the keypoint extraction stage.

Measuring activities of construction machinery can directly provide information such as productivity (e.g., the amount of materials moved), the utilisation of specfic machinery, and greenhouse gas emissions. This information can be further used for construction fleet management and optimisation, responding to changes in schedules, supply chain or weather conditions. It can be considered as the sensing/monitoring module to create a digital twin for construction sites. Such research represents a promising start towards achieving the digitalisation of the construction industry.

CRediT authorship contribution statement

Shuozhi Wang: Conceptualization, Data curation, Methodology, Formal analysis, Writing – Original draft. **Lichao Yang:** Methodology, Writing – review & editing. **Zichao Zhang:** Investigation, Writing – review & editing. **Yifan Zhao:** Supervision, Writing – review & editing, Project administration.

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.

Data availability

Data will be made available on request.

Acknowledgments

This work was supported by the Royal Academy of Engineering Industrial Fellowship [#grant IF2223B-110].

Appendix A. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.eswa.2023.121525.

References

Akbal, E., Barua, P. D., Dogan, S., Tuncer, T., & Acharya, U. R. (2022). DesPatNet25: Data encryption standard Cipher model for accurate automated construction site monitoring with sound signals. *Expert Systems with Applications*, *193*, Article 116447. http://dx.doi.org/10.1016/j.eswa.2021.116447, URL https://www.sciencedirect.com/science/article/pii/S0957417421017322.

Akhavian, R., & Behzadan, A. H. (2016). Smartphone-based construction workers' activity recognition and classification. *Automation in Construction*, *71*, 198–209. http://dx.doi.org/10.1016/J.AUTCON.2016.08.015.

Anon (2016). MSCOCO keypoint evaluation metric. Available on: https://cocodataset.org/#keypoints-eval.

Bae, J., Kim, K., & Hong, D. (2019). Automatic identification of excavator activities using joystick signals. *International Journal of Precision Engineering and Manufacturing*, *20*, 2101–2107. http://dx.doi.org/10.1007/s12541-019-00219-5.

Batty, M. (2018). Digital twins. *Environment and Planning B: Urban Analytics and City Science*, *45*(5), 817–820. http://dx.doi.org/10.1177/2399808318796416, arXiv:10.1177/2399808318796416.

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

Cao, Z., Hidalgo, G., Simon, T., Wei, S.-E., & Sheikh, Y. (2021). OpenPose: Realtime multi-person 2D pose estimation using part affinity fields. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *43*(1), 172–186. http://dx.doi.org/10.1109/TPAMI.2019.2929257.

Chen, C., Zhu, Z., & Hammad, A. (2020). Automated excavators activity recognition and productivity analysis from construction site surveillance videos. *Automation in Construction*, *110*, Article 103045. http://dx.doi.org/10.1016/j.autcon.2019.103045, URL https://linkinghub.elsevier.com/retrieve/pii/S0926580519303905.

Cheng, C. F., Rashidi, A., Davenport, M. A., & Anderson, D. V. (2017). Activity analysis of construction equipment using audio signals and support vector machines. *Automation in Construction*, *81*, 240–253. http://dx.doi.org/10.1016/J.AUTCON.2017.06.005.

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

Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An image is worth 16x16 words: Transformers for image recognition at scale. CoRR bs/2010.11929, arXiv:2010.11929.

Eastman, C. M. (1974). *An outline of the building description system*. Institute of Physical Planning, Carnegie-Mellon University.

Fang, H.-S., Xie, S., Tai, Y.-W., & Lu, C. (2017). RMPE: Regional multi-person pose estimation. In *ICCV*.

Feng, P.-E., Peng, B., Gao, Y., & Qiu, Q.-Y. (2016). Intelligent identification for working-cycle stages of hydraulic excavator. *Zhejiang Daxue Xuebao (Gongxue Ban)/Journal of Zhejiang University (Engineering Science)*, *50*(2), 209–217. http://dx.doi.org/10.3785/j.issn.1008-973X.2016.02.003, Cited by: 4.

Gilmer, J., Schoenholz, S. S., Riley, P. F., Vinyals, O., & Dahl, G. E. (2017). Neural message passing for quantum chemistry. http://dx.doi.org/10.48550/ARXIV.1704.01212, arXiv.

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. http://dx.doi.org/10.1061/(ASCE)CP.1943-5487.0000027.

Grieves, M. (2015). Digital twin: Manufacturing excellence through virtual factory replication-a whitepaper by Dr. Michael grieves. *White Paper*, 1–7.

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

Han, S., & Lee, S. (2013). A vision-based motion capture and recognition framework for behavior-based safety management. *Automation in Construction*, *35*, 131–141. http://dx.doi.org/10.1016/J.AUTCON.2013.05.001.

He, K., Gkioxari, G., Dollar, P., & Girshick, R. (2017). Mask R-CNN. In *Proceedings of the IEEE international conference on computer vision*.

He, X., & Jiang, Y. (2018). Review of hybrid electric systems for construction machinery. *Automation in Construction*, *92*, 286–296. http://dx.doi.org/10.1016/j.autcon.2018.04.005.

He, K., Zhang, X., Ren, S., & Sun, J. (2015). Deep residual learning for image recognition. CoRR abs/1512.03385, arXiv:1512.03385.

Henaff, M., Bruna, J., & LeCun, Y. (2015). Deep convolutional networks on graph-structured data. arXiv:1506.05163.

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

Jiang, F., Ma, L., Broyd, T., Chen, W., & Luo, H. (2022b). Building digital twins of existing highways using map data based on engineering expertise. *Automation in Construction*, *134*, Article 104081. http://dx.doi.org/10.1016/j.autcon.2021.104081.

Kim, H., Ahn, C. R., Engelhaupt, D., & Lee, S. (2018). Application of dynamic time warping to the recognition of mixed equipment activities in cycle time measurement. *Automation in Construction*, *87*, 225–234. http://dx.doi.org/10.1016/j.autcon.2017.12.014.

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. http://dx.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*, Article 104443. http://dx.doi.org/10.1016/J.AUTCON.2022.104443.

Kim, J., Chi, S., & Ahn, C. R. (2021). Hybrid kinematic–visual sensing approach for activity recognition of construction equipment. *Journal of Building Engineering*, *44*, Article 102709. http://dx.doi.org/10.1016/j.jobe.2021.102709, URL https://linkinghub.elsevier.com/retrieve/pii/S2352710221005672.

Kim, J., Chi, S., & Seo, J. (2018). Interaction analysis for vision-based activity identification of earthmoving excavators and dump trucks. *Automation in Construction*, *87*, 297–308. http://dx.doi.org/10.1016/j.autcon.2017.12.016.

Kingma, D. P., & Ba, J. (2017). Adam: A method for stochastic optimization. arXiv:1412.6980.

Kipf, T. N., & Welling, M. (2016). Semi-supervised classification with graph convolutional networks. arXiv preprint arXiv:1609.02907.

Koo, B., La, S., Cho, N.-W., & Yu, Y. (2019). Using support vector machines to classify building elements for checking the semantic integrity of building information models. *Automation in Construction*, *98*, 183–194. http://dx.doi.org/10.1016/j.autcon.2018.11.015.

Li, Q., Han, Z., & Wu, X.-M. (2018). Deeper insights into graph convolutional networks for semi-supervised learning. arXiv:1801.07606.

Li, H., Lu, M., Hsu, S.-C., Gray, M., & Huang, T. (2015). Proactive behavior-based safety management for construction safety improvement. *Safety Science*, *75*, 107–117. http://dx.doi.org/10.1016/j.ssci.2015.01.013.

Liang, C.-J., Lundeen, K. M., McGee, W., Menassa, C. C., Lee, S., & Kamat, V. R. (2019). A vision-based marker-less pose estimation system for articulated construction robots. *Automation in Construction*, *104*, 80–94. http://dx.doi.org/10.1016/j.autcon.2019.04.004.

Lin, T.-Y., Dollár, P., Girshick, R., He, K., Hariharan, B., & Belongie, S. (2016). Feature pyramid networks for object detection. http://dx.doi.org/10.48550/arxiv.1612.03144.

Lu, Q., Xie, X., Heaton, J., Parlikad, A. K., & Schooling, J. (2020). From BIM towards digital twin: Strategy and future development for smart asset management. In T. Borangiu, D. Trentesaux, P. L. ao, A. G. Boggino, & V. Botti (Eds.), *Service oriented, holonic and multi-agent manufacturing systems for industry of the future* (pp. 392–404). Cham: Springer International Publishing.

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. http://dx.doi.org/10.1016/j.autcon.2016.02.003.

Luo, H., Wang, M., Wong, P. K.-Y., & Cheng, J. C. (2020). Full body pose estimation of construction equipment using computer vision and deep learning techniques. *Automation in Construction*, *110*, Article 103016.

Martin, J. (2021). Productivity in the construction industry, UK: 2021.

Opoku, D.-G. J., Perera, S., Osei-Kyei, R., & Rashidi, M. (2021). Digital twin application in the construction industry: A literature review. *Journal of Building Engineering*, *40*, Article 102726. http://dx.doi.org/10.1016/j.jobe.2021.102726, URL https://www.sciencedirect.com/science/article/pii/S2352710221005842.

Parente, M., Cortez, P., & Gomes Correia, A. (2015). An evolutionary multi-objective optimization system for earthworks. *Expert Systems with Applications*, *42*(19), 6674–6685. http://dx.doi.org/10.1016/j.eswa.2015.04.051, URL https://www.sciencedirect.com/science/article/pii/S0957417415002936.

Park, J. W., Yang, X., Cho, Y. K., & Seo, J. (2017). Improving dynamic proximity sensing and processing for smart work-zone safety. *Automation in Construction*, *84*, 111–120. http://dx.doi.org/10.1016/J.AUTCON.2017.08.025.

Pregnolato, M., Gunner, S., Voyagaki, E., Risi, R. D., 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*, Article 104421. http://dx.doi.org/10.1016/J.AUTCON.2022.104421.

Rashid, K. M., & Louis, J. (2019). Times-series data augmentation and deep learning for construction equipment activity recognition. *Advanced Engineering Informatics*, *42*, http://dx.doi.org/10.1016/j.aei.2019.100944.

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster R-CNN: Towards real-time object detection with region proposal networks. In C. Cortes, N. Lawrence, D. Lee, M. Sugiyama, & R. Garnett (Eds.), *Advances in neural information processing systems, Vol. 28*. Curran Associates, Inc..

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*, Article 102811. http://dx.doi.org/10.1016/j.autcon.2019.04.006.

Sandler, M., Howard, A. G., Zhu, M., Zhmoginov, A., & Chen, L. (2018). Inverted residuals and linear bottlenecks: Mobile networks for classification, detection and segmentation. CoRR abs/1801.04381, arXiv:1801.04381.

Sato, M., Tsunano, Y., Sano, K., Warisawa, S., Aizawa, M., Nishimura, K., & Fukui, R. (2022). Evaluation of excavation motion sequence for hydraulic excavators based on extraction of excavation style and phase. *Journal of Field Robotics*, *39*(7), 1112–1122. http://dx.doi.org/10.1002/rob.22090, arXiv:https://onlinelibrary.wiley.com/doi/pdf/10.1002/rob.22090.

Scarpiniti, M., Colasante, F., Di Tanna, S., Ciancia, M., Lee, Y.-C., & Uncini, A. (2021). Deep belief network based audio classification for construction sites monitoring. *Expert Systems with Applications*, *177*, Article 114839. http://dx.doi.org/10.1016/j.eswa.2021.114839, URL https://www.sciencedirect.com/science/article/pii/S0957417421002803.

Schlichtkrull, M., Kipf, T. N., Bloem, P., Berg, R. v. d., Titov, I., & Welling, M. (2017). Modeling relational data with graph convolutional networks. http://dx.doi.org/10.48550/ARXIV.1703.06103, arXiv.

Shi, Y., Xia, Y., Luo, L., Xiong, Z., Wang, C., & Lin, L. (2021). Working stage identification of excavators based on control signals of operating handles. *Automation in Construction*, *130*, Article 103873. http://dx.doi.org/10.1016/j.autcon.2021.103873.

Shi, Y., Xia, Y., Zhang, Y., & Yao, Z. (2020). Intelligent identification for working-cycle stages of excavator based on main pump pressure. *Automation in Construction*, *109*, Article 102991. http://dx.doi.org/10.1016/j.autcon.2019.102991.

Shi, D., Zhong, Y., Cao, Q., Ma, L., Li, J., & Tao, D. (2023). TriDet: Temporal action detection with relative boundary modeling. arXiv:2303.07347.

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. arXiv:1409.1556.

Soltani, M. M., Zhu, Z., & Hammad, A. (2017). Skeleton estimation of excavator by detecting its parts. *Automation in Construction*, *82*, 1–15. http://dx.doi.org/10.1016/J.AUTCON.2017.06.023.

Tang, J., Wang, M., Luo, H., Wong, P. K.-Y., Zhang, X., Chen, W., & Cheng, J. C. (2023). Full-body pose estimation for excavators based on data fusion of multiple onboard sensors. *Automation in Construction*, *147*, Article 104694. http://dx.doi.org/10.1016/j.autcon.2022.104694, URL https://www.sciencedirect.com/science/article/pii/S0926580522005647.

Tompson, J., Goroshin, R., Jain, A., LeCun, Y., & Bregler, C. (2015). Efficient object localization using convolutional networks. In *2015 IEEE conference on computer vision and pattern recognition* (pp. 648–656).

Wei, S.-E., Ramakrishna, V., Kanade, T., & Sheikh, Y. (2016). Convolutional pose machines. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4724–4732).

Zhang, C., Wu, J., & Li, Y. (2022). ActionFormer: Localizing moments of actions with transformers. arXiv:2202.07925.

Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., & Sun, M. (2020). Graph neural networks: A review of methods and applications. *AI Open*, *1*, 57–81. http://dx.doi.org/10.1016/j.aiopen.2021.01.001.

2023-09-22

Keypoints-based heterogeneous graph convolutional networks for construction

Wang, Shuozhi

Elsevier

Wang S, Yang L, Zhang Z, Zhao Y. (2024) Keypoints-based heterogeneous graph convolutional networks for construction, Expert Systems with Applications, Volume 327, Part C, March 2024, Article Number 121525

https://doi.org/10.1016/j.eswa.2023.121525