2022 European Conference on Computing in Construction
Ixia, Rhodes, Greece
July 24-26, 2022

# GRAPH NEURAL NETWORKS IN BUILDING LIFECYCLE: A REVIEW

Yilong Jia[1], Jun Wang[2], M. Reza Hosseini[1], Wenchi Shou[2]
[1]Deakin University, Geelong, Australia
[2]Western Sydney University, Sydney, Australia

## ABSTRACT

Graph neural networks (GNNs) have attracted much attention in the field of machine learning because of their excellent performance on graph data. Graph data in the architecture, engineering and construction (AEC) sector is very common, such as bubble diagrams for space planning and point clouds for scan-to-BIM models. Some studies in AEC have adopted GNNs to solve practical problems. However, there has been a limited focus on the outcomes of these studies. Therefore, this paper aims to review the applications of GNNs in the building lifecycle. A wide range of existing literature was retrieved. The result shows that the adoption of GNNs is still in its infancy but has been increasing dramatically in recent years. Ten application domains were identified from the planning stage to the operation stage. In addition, the challenges and opportunities of GNNs adoption in AEC were discussed providing directions for future research.

## INTRODUCTION

With the adoption of digital technologies in the field of architecture, engineering and construction (AEC) such as building information modelling (BIM) and geographical information system (GIS), the data across the whole building lifecycle are generated explosively and can be easily archived. These big data are valuable for extracting both explicit and implicit rules of AEC practice. However, the large volume and high complexity hinder the use of traditional data processing tools. Alternatively, deep learning as an important subfield of artificial intelligence and machine learning has been adopted in processing big data without human intervention. Many studies in the AEC sector have successfully applied classical deep learning models, such as artificial neural networks (ANNs), convolutional neural networks (CNNs) and recurrent neural networks (RNNs), to various scenarios across the building lifecycle (Darko et al., 2020). These classical deep learning models have remarkable on well-structured data such as texts, images, and videos, but cannot process graph data.

The graph discussed in this paper is a type of data structure instead of images, pictures, drawings, etc. A graph is intuitively presented by a set of nodes connected by a set of edges in which the order of each node and the size of the graph are variable. The graph-structured data is quite common in daily life such as social networks on Facebook and citation networks in academia. Naturally, graph-structured data is also not rare in the AEC sector, such as bubble diagrams for floor plan design, Gantt charts for project scheduling, and relationships of entities in industry foundation classes (IFC) files.

To process graph-structured data, researchers in the field of computer science have proposed graph neural networks (GNNs) which can perform graph analysis tasks such as node classification, edge prediction and graph classification. GNNs have been widely applied to chemistry, biology, recommendation systems, traffic networks, etc (Zhou et al., 2020). Some studies in the AEC sector also investigated the applications of GNNs such as BIM segmentation enrichment (Wang et al., 2022), point cloud semantic segmentation (Feng et al., 2021), and prediction of building energy consumption (Hu et al., 2022). However, the adoption of GNNs in AEC is still in its infancy. To embrace this new deep learning model, it is necessary to understand the significance of GNNs in the AEC sector. This paper aims to review the existing applications of GNNs in the building lifecycle including planning, design, construction and operation stages.

In the following sections, the concept of graphs and GNNs is firstly introduced. The next section illustrates the research method. On this basis, the current research on the application of GNNs in the AEC sector is reviewed, followed by a discussion on challenges and opportunities. The closing section concludes the findings of this paper.

## BACKGROUND

### Graph

A graph $G$ is a type of non-Euclidean data structure consisting of a set of nodes (or vertices) $V$ and a set of edges $E$. A graph can be denoted as $G = (V, E)$; and an edge can be denoted by its nodes as $e_{ij} = (v_i, v_j) \in E$, where $v_i, v_j \in V$. A graph can be also represented as a $|V| \times |V|$ matrix called adjacency matrix $A$ for computational purposes. If $e_{ij} \in E$, $A_{ij} = 1$; and if $e_{ij} \notin E$, $A_{ij} = 0$. Each node, edge or the whole graph may have attributes that can be represented as feature vectors to contain more information.

A graph can be categorised as follows (Zhou et al., 2020).

- a directed or an undirected graph, depending on if the edges are directed or undirected from a node to another node
- a homogeneous or a heterogeneous graph, depending on if all the nodes or edges represent the same class of information
- a static or a dynamic graph, depending on if the attributes or the topologic structure will change with time

### Graph Neural Networks (GNNs)

The basic idea of GNNs is that the representation of each node in a graph is determined by its own features and the aggregation of the features of its neighbouring nodes through edges. GNNs have better performance in processing relationships among nodes (Hu et al., 2022), while traditional deep learning approaches, such as CNNs and RNNs, assume that nodes are independent of each other and ignore the topological information (Wu et al., 2021b). There are three types of tasks that GNNs can perform:

- Node-level tasks: node classification, node clustering and node regression
- Edge-level tasks: edge classification and link prediction
- Graph-level tasks: graph classification and graph regression

Based on the propagation modules, GNNs can be classified into three categories: recurrent GNNs (Rec-GNNs), convolutional GNNs (Conv-GNNs) and Skip-GNNs (Zhou et al., 2020). The early studies on GNNs stated from Rec-GNNs. Based on recursive neural networks, Gori et al. (2005) formally proposed the concept of GNNs for the first time, and Scarselli et al. (2009) further developed the concept. Inspired by the idea of CNNs, Conv-GNNs were proposed, which generalizes convolution operators from CNNs to GNNs (Wu et al., 2021b). The main difference between Rec-GNNs and Conv-GNNs is the weights among layers: the former uses the same weights across layers during propagation, while the latter uses different weights (Wieder et al., 2020). Based on Rec-GNNs or Conv-GNNs, Skip-GNNs add "skip connection" across layers to avoid over smoothing problems and increased noise in deeper networks (Zhou et al., 2020). Researchers in the discipline of computer science have conducted several surveys or reviews on GNNs (Bronstein et al., 2017; Zhang et al., 2019; Zhou et al., 2020; Wu et al., 2021b; Wang et al., 2022; Zhang et al., 2022). Comprehensive explanations of graphs and GNNs can be found in these studies.

## RESEARCH METHOD

### Data Collection

To have a comprehensive review, an extensive search for relevant research was conducted. Scopus was used as the primary database supplemented with Web of Science and Google Scholar. Keywords related to GNNs and AEC, such as "graph neural networks", "architecture", "construction" and "buildings", were searched in the field of title, abstract and author keywords. Journal articles, conference proceedings, theses and arXiv preprints were included. The publication year was unlimited. After manually screening their titles and abstracts, 33 articles were retained for analysis, including 19 journal articles, 11 conference proceedings, two preprints and one master's thesis. Figure 1 shows the chronological distribution of retrieved articles. The number of 2022 was incomplete as this study was conducted in January 2022. The adoption of GNNs began in 2018 but has been increasing sharply in recent years. The top five sources are *Automation in Construction, ACM Transactions on Graphics, Proceedings of the 2021 European Conference on Computing in Construction, Proceedings of the 38th International Symposium on Automation and Robotics in Construction (ISARC)*, and *arXiv*.

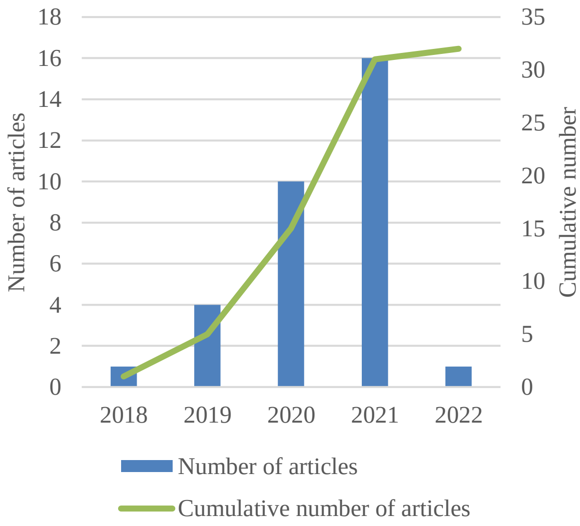


*Figure 1: Chronological distribution of articles*

### Data Analysis

Content analysis was employed in this paper for qualitative data analysis. The application scenarios of each article were initially labelled when screening the title and abstract. the retrieved articles were reviewed in full text to determine the research problems, analyze the mechanism of GNNs application, and categorize them into various stages of the building lifecycle.

## APPLICATIONS OF GNNS

The building lifecycle consists of four stages: planning, design, construction and operation stages (Meng et al., 2020). Based on the four stages, ten application domains of GNNs were identified from the retrieved literature as shown in Figure 2.

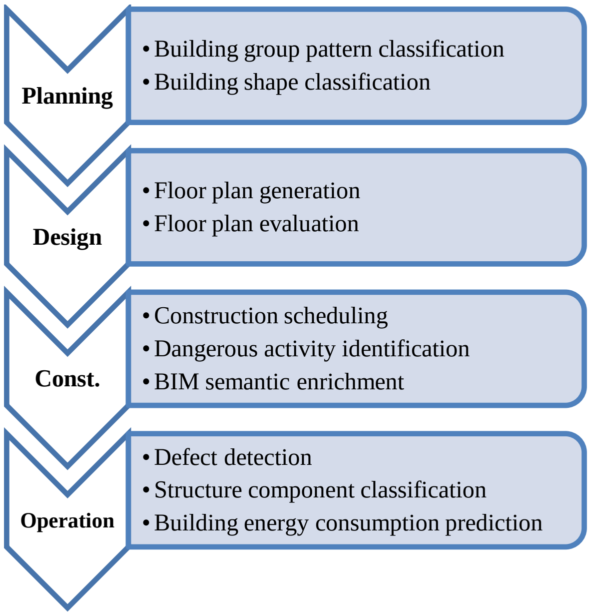


*Figure 2 Applications of GNNs in building lifecycle*

### Planning Stage

The conceptual plan of a construction project forms at this stage. The feasibility, location, time, budget, standards, planning scheme and other constraints should be initially identified. Usually, according to the planning scheme, the spatial cognition of the proposed building should harmonize with its surrounding buildings and environment. Through GNNs combined with GIS, spatial patterns and shapes of buildings can be automatically recognised.

**Classification of building group patterns**. At the level of building groups, each building can be seen as a node, and the lines between the buildings serve as edges, thus forming a graph. The graph convolutional neural network (GCNN) can learn the representation of nodes and edges on the graph to classify the patterns of building groups into regular and irregular patterns (Yan et al., 2019). Further, Zhao et al. (2020) employed graph convolutional networks (GCNs) to classify building groups into specific patterns, such as colinear patterns, rectangle patterns, grid patterns, etc.

**Classification of building shapes**. At the level of single buildings, the graph can be formed by the outline of the building: the node is the vertex of the outline, and the edge is the line between adjacent vertices. Yan et al. (2021) proposed a GCN-based encoder-decoder model to learn the representation of different building shapes in the feature space via unsupervised learning, thus distinguishing different building shapes without manual data labelling. Similarly, Liu et al. (2021) proposed another GCN-based model but with supervised learning to classify buildings into ten alphabet shapes, such as F-shape, E-shape, T-shape, etc. The GNN-enabled automatic classification of building group patterns and building shapes supports urban planning authorities

### Design Stage

At the design stage, GNNs are mainly adopted for floor plan design. At the preliminary phase of the architectural design, the bubble diagram helps architects graphically plan and organize rooms and spaces. A bubble diagram is essentially a graph: nodes are rooms or spaces, and the edges are connections between rooms or spaces such as doors, walls and openings. Attributes, such as the area of the room and the material of the wall, can be attached to nodes or edges.

**Generative design of floor plans**. GNNs are able to learn the feature representation of both explicit and implicit rules of space planning from existing floor plan designs (Hu et al., 2020). The combination of generative adversarial networks (GANs) and GNNs enables the generative design of floor plans. Nauata et al. (2020) proposed a GAN model House-GAN for generating floor plans, in which a novel GCN-based computational module Conv-MPN acts as both generator and discriminator. To extend the generative design from a single floor to the whole building, Chang et al. (2021) proposed another GAN model, in which GNN modules encode bubble diagrams and voxel graphs in generator and discriminator.

**Evaluation of floor plans**. GNNs are also applied to the evaluation of floor plans. As et al. (2019) adopted a GCN-based module Neural FPs that was originally designed for predicting functions of chemical compounds to evaluate the target functional scores of a residential building. This model performs a graph-level regression that identifies high-scored patterns (subgraphs) of room settings and combinations. Interestingly, this model is able to predict subjective scores such as liveability and sleepability, indicating implicit evaluation criteria can be learned by GNNs.

### Construction Stage

Uncertainty of site environment and human errors in the construction phase results in differences between the actual and the planned. The current applications of GNNs try to address the uncertainty of scheduling, construction safety and BIM semantic enrichment.

**Construction scheduling**. Hong et al. (2021) converted construction schedules into graphs in which nodes are construction activities, and edges are logical relationships between activities. A GCN model was trained through semi-supervised learning to classify construction activities so that the standard construction activity groups can be identified, and their construction time can be estimated at the early stage of construction.

**Identification of dangerous activities**. Tang and Golparvar-Fard (2021) constructed a spatio-temporal graph mapping the interaction among construction activities, worker body key points and construction tools. A spatio-temporal GNN model was adopted on the graph to identify workers' gestures and used as inputs for safety evaluation.

**BIM semantic enrichment**. BIM semantic enrichment is a process that infers new semantic information of BIM objects or relationships from a BIM model (Belsky et al., 2016). Semantic enrichment can deal with the loss of semantic information during data exchange between different BIM tools. It is also useful for scan-to-BIM to assign correct attributes to as-built BIM models. BIM object classification is one of the tasks of BIM semantic enrichment (Bloch and Sacks, 2020). Traditional methods of BIM semantic enrichment are rule-based, and recently some studies have investigated GNN-based methods for classifying BIM objects.

Collins et al. (2021) constructed graphs from triangle meshes and point clouds that are converted from the geometry of BIM objects. They proposed a GCN-based approach to encode geometric representations of BIM objects. Through supervised learning, the model is able to classify BIM objects into corresponding IFC entities. On the graphs constructed from bubble diagrams, Wang et al. (2021) used GraphSAGE to classify rooms into corresponding types. However, GraphSAGE does not aggregate the attributes of edges during propagation. As a result, different connection types between rooms cannot be considered in room classification. To tackle this issue, Wang et al. (2022) modified GraphSAGE into SAGE-E which concatenates the features of neighbouring nodes and edges during message passing. The result shows SAGE-E outperforms GraphSAGE.

### Operation Stage

The applications of GNNs at the operation stage mainly focus on defect detection and structural component classification for maintenance tasks. These applications are enabled by point cloud semantic segmentation, that is, each point in a point cloud is classified into specific categories. Point clouds can be easily obtained with scanning devices, such as LiDAR or terrestrial laser scanning (TLS). Each point in the scanned point cloud has attributes such as coordinates (XYZ), colours (RGB), and reflection intensity. The k-nearest neighbours (KNN) algorithm can be applied to the point cloud to form a graph for each point. Based on the KNN graph, a GNN-based model, dynamic graph convolutional network (DGCNN), can learn the local features and globe features of the graph and perform semantic segmentation (Wang et al., 2019).

**Defect detection**. Bahreini and Hammad (2021) adopted DGCNN to identify defects on concrete surfaces from scanned 3D point cloud data. The point cloud data were manually labelled with three types of labels: crack, spalling and non-defect. Each point is represented as a 7-dimensional vector including its XYZ coordinates, GRB channels, and a normalized Y coordinate. However, this supervised learning requires manual labelling of thousands of points, which is time-consuming and error-prone. To reduce this burden, a semi-supervised approach was proposed by Feng et al. (2021). In their research, a GCN-based semi-supervised approach is used to detect pavement cracks from point cloud data. To achieve better performance of semi-supervised learning, features should be well designed to distinguish the features of different categories. Feng et al. (2021) firstly conduct a space mapping to amplify the difference in reflection intensity between cracks and non-crack areas. In addition to XY coordinates and amplified reflection intensity, four local features are designed to enhance the representation of points.

**Structural component classification**. Kim and Kim (2020) adopted DGCNN to classify bridge components into three types of bridges. To achieve higher accuracy in classifying tall components on bridges, Lee et al. (2021) proposed a DGCNN-based model called hierarchical DGCNN (HGCNN) that shortens the architecture of DGCNN and expands the range of the KNN graph. HGCNN shows equivalent levels of overall accuracy (OA) and intersection over union (IoU) to PointNet and DGCNN, while providing better performance for tall components such as electric poles. Yajima et al. (2021) proposed the DGPointNet model, which concatenates local and global features from PointNet and DGCNN, which can classify damaged building components in a disaster environment more accurately.

The above research is based on DGCNN. According to different scene requirements, some improvements are made to DGCNN, such as DGPointNet and HGCNN mentioned above. These improvements are achieved by modifying the model architecture. However, the accuracy of DGCNN varies in classifying different building components. Some points of walls at corners were misclassified as columns because DGCNN uses cube blocks in semantic segmentation (Kim and Kim, 2021). Similarly, the accuracy of classifying abutments and piers by DGCNN is relatively lower than of other bridge components (Kim and Kim, 2020). As a result, additional attention should be paid to these areas. For example, if a point is classified as a column but connected to a wall, it should be corrected to a wall.

In addition, Kim and Kim (2020) compared the performance between DGCNN and the other two CNN-based deep learning models, PiontNet and PointCNN, in classifying bridge components. Their research shows that DGCNN has the highest accuracy in terms of OA and IoU.

**Building energy consumption**. In addition to point cloud semantic segmentation, GNNs are also applied to the prediction of building energy consumption. Hu et al. (2022) mapped the relationships of building shadows into a directed dynamic graph, in which buildings are regarded as nodes, and shadows cast from one building to another are regarded as edges. The edge attributes reflect the solar impact, and the edge attributes represent the features of buildings and weather. A spatial-temporal graph convolutional networks (ST-GCNs) model was adopted to learn the time-series change among attributes and thus predict the building energy consumption.

## CHALLENGES AND OPPORTUNITIES

Although GNNs have been adopted across the building lifecycle, they are a new deep learning approach in the AEC sector, which still faces challenges and opportunities in application and research.

### Challenges

**Manual processes of graph construction**. In the studies reviewed in the previous section, most of the graphics construction process is manual, such as BIM models to point clouds, floor plans to bubble diagrams, and construction schedules to graphs. The manual process leads to time-consuming and error-prone preparation of datasets. The reason is twofold. Firstly, the construction of graph data is more complex compared with traditional data structures such as texts, images, and numbers. The topological structure, i.e., the adjacency matrix, should be defined at first; node attributes and edges attributes are then assigned to the graph. However, topological structures and attributes are stored separately in many cases and tend to be lost during the conversion process. For example, when converting a BIM model to a point cloud, the geometry of BIM objects originally stored in a Revit file is transformed through pieces of software, while its semantic information is lost in the process and has to be manually added to the point cloud (Ma et al., 2020). Secondly, raw data may be stored in different data formats. For example, floor plans may be stored in the forms of CAD/BIM models, raster images, or even hand drawings. Due to the diversity of data formats, automatic conversion methods cannot be unified and standardized. Even in the same data format, the requirements for extracting information are different in different usage scenarios, which further hinders the implementation of automatic conversion. For example, given a BIM model, BIM semantic enrichment requires triangular meshes (Collins et al., 2021), while floor plans evaluation requires a bubble diagram (As et al., 2019).

**Lack of AEC-related open datasets**. As stated above, the dataset construction and data pre-processing for training and testing GNN models are cumbersome. Most existing studies need to establish datasets by themselves. The size of self-built datasets is limited, and the quality cannot be guaranteed, which is likely to result in underfitting or low generalization capability of GNN models. The benchmarking between different models also becomes impossible without standard datasets. The AEC-related open datasets are required for adopting machine learning in the field of AEC. In the retrieved literature, only seven studies employed open databases or datasets, and other studies had to establish their own datasets because there was no open dataset available. In addition, the open datasets focus mainly on floor plans, such as RPLAN (Wu et al., 2019), Repository of Unique Buildings (RUB) (Simonsen et al., 2021), and LIFULL HOME'S Dataset (National Institute of Informatics, 2021). However, none of these datasets is formatted in graphs. Additional endeavour is necessary to convert floor plans into graphs.

### Opportunities

**knowledge graph**. The knowledge graph is a very popular concept in interdisciplinary applications. The ontology and the reasoner are two core components in a knowledge graph. In the field of construction, a variety of ontologies have been established for different scenarios, such as concrete bridge rehabilitation (Wu et al., 2021a), road asset management (Lei et al., 2021), and construction defects (Lee et al., 2016). However, most reasoners on these ontologies in construction are rule-based requiring manual intervention (Lei et al., 2021).

From the perspective of graphs, a knowledge graph is a directed and heterogeneous graph consisting of entities (nodes) and relations (edges) (Zhou et al., 2020). This graph structure makes it possible to adopt GNN models in the reasoner of knowledge graphs. Apart from the construction industry, several GNN-based approaches have been used for knowledge graph embedding, relation extraction and graph completion (Ji et al., 2021; Wu et al., 2022). These studies confirm the feasibility of exploring the adoption of GNN-based in the AEC-related ontologies. For example, structured information can be extracted from the BIM model, and unstructured information can be extracted using natural language processing (NLP) from project documents. The extracted information is then applied to the predefined ontology, forming a graph. On the constructed knowledge graph, GNN models can classify duplicated entities and predict missing entities or relations (Wu et al., 2022).

**Generative design**. GNNs can learn not only explicit design rules but also implicit design rules from existing designs, enabling more complex alternative designs (As et al., 2019). Because GNNs are an end-to-end learning model, raw data can be directly fed into the model without manual feature extraction, and the design rules can be learnt by GNNs automatically, reducing the requirement for programming skills. GANs have been integrated with GNNs for the generative design of floor plans (Nauata et al., 2020; Chang et al., 2021). GANs train the generator and discriminator to learn the design rules and evaluation rules. Compared with the classic generative design, a fully trained GNN-based generative design can directly generate suitable designs without iteration and evaluation, thus reducing the computing costs. As stated in the previous section, the current adoption of GNNs in generative design is limited in the generation of floor plans (As et al., 2019; Nauata et al., 2020; Chang et al., 2021). The potential of GNNs in generative design has not been fully explored. More adoption of GNN-based generative design in other application domains, such as structure design, plumbing plan, HVAC ducting and urban design, has not been explored.

**Exploration for graph structures in AEC**. Graph structures can generally be categorized into two scenarios:

structural scenarios in which the relational structure is explicit, and non-structural scenarios in which the relational structure is implicit or absent (Zhou et al., 2020). For structural scenarios, graph structures can be extracted from BIM models, drawings, and other existing forms. For non-structural scenarios, images and texts are the common sources of graphs. All graph structures identified in this paper belong to structural scenarios. Therefore, more graph structures in both structural scenarios and non-structural scenarios should be explored in future research. For example, an HVAC or plumbing system can be presented as a graph where devices are nodes and pipes are edges.

## CONCLUSION

GNNs extend deep learning to graph data and are rapidly being applied to solving practical problems in the AEC sector. This paper reviewed the applications of GNNs in the building lifecycle and discussed the opportunities and challenges in future research and practice. The increasing trend of literature indicates a growing interest in GNNs among AEC researchers. The applications of GNNs have emerged at all stages of the building lifecycle. The development of BIM and point cloud technologies make it easier to record and obtain AEC-related big data, providing a large amount of data for training GNN models.

Although GNNs show exciting potential in the AEC sector, it is still in infancy. The construction of graphs requires manual operation because of the complexity of graphs and the variety of data formats in AEC. In addition, the lack of open datasets related to AEC makes the comparative evaluation of GNN-based research difficult. On the other hand, GNNs may have more potential in the research of the knowledge graph and generative design. It is also recommended to explore more graph-structured data in AEC to make full use of GNNs.

## REFERENCES

As, I., Pal, S. & Basu, P. (2019) Composing frankensteins:Data-driven design assemblies through graph-based deep neural networks. In: the 107th Annual Meeting BLACK BOX: Articulating Architecture's Core in the Post-Digital Era. Pittsburgh, PA, USA, ACSA.

Bahreini, F. & Hammad, A. (2021) Point cloud semantic segmentation of concrete surface defects using dynamic graph cnn. In: 38th International Symposium on Automation and Robotics in Construction (ISARC). Dubai, UAE, International Association for Automation and Robotics in Construction (IAARC).

Belsky, M., Sacks, R. & Brilakis, I. (2016) Semantic enrichment for building information modeling. Computer-Aided Civil and Infrastructure Engineering, 31, 4, pp.261-274.

Bloch, T. & Sacks, R. (2020) Clustering information types for semantic enrichment of building information models to support automated code compliance checking. Journal of Computing in Civil Engineering, 34, 6, p.04020040.

Bronstein, M.M., Bruna, J., Lecun, Y., Szlam, A. & Vandergheynst, P. (2017) Geometric deep learning: Going beyond euclidean data. IEEE Signal Processing Magazine, 34, 4, pp.18-42.

Chang, K.-H., Cheng, C.-Y., Luo, J., Murata, S., Nourbakhsh, M. & Tsuji, Y. (2021) Building-gan: Graph-conditioned architectural volumetric design generation. arXiv:2104.13316.

Collins, F.C., Braun, A., Ringsquandl, M., Hall, D.M. & Borrmann, A. (2021) Assessing ifc classes with means of geometric deep learning on different graph encodings. In: 2021 European Conference on Computing in Construction. Online, University College Dublin.

Darko, A., Chan, A.P.C., Adabre, M.A., Edwards, D.J., Hosseini, M.R. & Ameyaw, E.E. (2020) Artificial intelligence in the aec industry: Scientometric analysis and visualization of research activities. Automation in Construction, 112, p.103081.

Feng, H., Li, W., Luo, Z., Chen, Y., Fatholahi, S.N., Cheng, M., Wang, C., Junior, J.M. & Li, J. (2021) Gcn-based pavement crack detection using mobile lidar point clouds. IEEE Transactions on Intelligent Transportation Systems, pp.1-10.

Gori, M., Monfardini, G. & Scarselli, F. (2005) A new model for learning in graph domains. In: 2005 IEEE International Joint Conference on Neural Networks. Montreal, QC, Canada, IEEE.

Hong, Y., Hovhannisyan, V., Xie, H. & Brilakis, I. (2021) Determining construction method patterns to automate and optimise scheduling – a graph-based approach. In: 2021 European Conference on Computing in Construction. Online, University College Dublin.

Hu, R., Huang, Z., Tang, Y., Kaick, O.V., Zhang, H. & Huang, H. (2020) Graph2plan: Learning floorplan generation from layout graphs. ACM Transactions on Graphics, 39, 4, p.118.

Hu, Y., Cheng, X., Wang, S., Chen, J., Zhao, T. & Dai, E. (2022) Times series forecasting for urban building energy consumption based on graph convolutional network. Applied Energy, 307, p.118231.

Ji, S., Pan, S., Cambria, E., Marttinen, P. & Yu, P.S. (2021) A survey on knowledge graphs: Representation, acquisition, and applications. IEEE Transactions on Neural Networks and Learning Systems, pp.1-21.

Kim, H. & Kim, C. (2020) Deep-learning-based classification of point clouds for bridge inspection. Remote Sensing, 12, 22, p.3757.

Kim, H. & Kim, C. (2021) 3d as-built modeling from incomplete point clouds using connectivity relations. Automation in Construction, 130, p.103855.

Lee, D.-Y., Chi, H.-L., Wang, J., Wang, X. & Park, C.-S. (2016) A linked data system framework for sharing construction defect information using ontologies and BIM environments. Automation in Construction, 68, pp.102-113.

Lee, J.S., Park, J. & Ryu, Y.-M. (2021) Semantic segmentation of bridge components based on hierarchical point cloud model. Automation in Construction, 130, p.103847.

Lei, X., Wu, P., Zhu, J. & Wang, J. (2021) Ontology-based information integration: A state-of-the-art review in road asset management. Archives of Computational Methods in Engineering.

Liu, C., Hu, Y., Li, Z., Xu, J., Han, Z. & Guo, J. (2021) Triangleconv: A deep point convolutional network for recognizing building shapes in map space. ISPRS International Journal of Geo-Information, 10, 10, p.687.

Ma, J.W., Czerniawski, T. & Leite, F. (2020) Semantic segmentation of point clouds of building interiors with deep learning: Augmenting training datasets with synthetic BIM-based point clouds. Automation in Construction, 113, p.103144.

Meng, Q., Zhang, Y., Li, Z., Shi, W., Wang, J., Sun, Y., Xu, L. & Wang, X. (2020) A review of integrated applications of BIM and related technologies in whole building life cycle. Engineering, Construction and Architectural Management, 27, 8, pp.1647-1677.

National Institute of Informatics (2021). Lifull home's dataset. https://www.nii.ac.jp/dsc/idr/en/lifull/

Nauata, N., Chang, K.-H., Cheng, C.-Y., Mori, G. & Furukawa, Y. (2020) House-gan: Relational generative adversarial networks for graph-constrained house layout generation. arXiv:2003.06988.

Scarselli, F., Gori, M., Tsoi, A.C., Hagenbuchner, M. & Monfardini, G. (2009) The graph neural network model. IEEE Transactions on Neural Networks, 20, 1, pp.61-80.

Simonsen, C.P., Thiesson, F.M., Philipsen, M.P. & Moeslund, T.B. (2021) Generalizing floor plans using graph neural networks. In: 2021 IEEE International Conference on Image Processing (ICIP). Anchorage, Alaska, USA, IEEE.

Tang, S. & Golparvar-Fard, M. (2021) Machine learning-based risk analysis for construction worker safety from ubiquitous site photos and videos. Journal of Computing in Civil Engineering, 35, 6, p.04021020.

Wang, Y., Sun, Y., Liu, Z., Sarma, S.E., Bronstein, M.M. & Solomon, J.M. (2019) Dynamic graph cnn for learning on point clouds. ACM Transactions on Graphics, 38, 5, p.146.

Wang, Z., Sacks, R. & Yeung, T. (2022) Exploring graph neural networks for semantic enrichment: Room type classification. Automation in Construction, 134, p.104039.

Wang, Z., Yeung, T., Sacks, R. & Su, Z. (2021) Room type classification for semantic enrichment of building information modeling using graph neural networks. In: the 38th International Conference of CIB W78. Luxembourg, CIB.

Wieder, O., Kohlbacher, S., Kuenemann, M., Garon, A., Ducrot, P., Seidel, T. & Langer, T. (2020) A compact review of molecular property prediction with graph neural networks. Drug Discovery Today: Technologies, p.37.

Wu, C., Li, X., Guo, Y., Wang, J., Ren, Z., Wang, M. & Yang, Z. (2022) Natural language processing for smart construction: Current status and future directions. Automation in Construction, 134, p.104059.

Wu, C., Wu, P., Wang, J., Jiang, R., Chen, M. & Wang, X. (2021a) Ontological knowledge base for concrete bridge rehabilitation project management. Automation in Construction, 121, p.103428.

Wu, W., Fu, X.-M., Tang, R., Wang, Y., Qi, Y.-H. & Liu, L. (2019) Data-driven interior plan generation for residential buildings. ACM Transactions on Graphic, 38, 6, p.234.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C. & Yu, P.S. (2021b) A comprehensive survey on graph neural networks. IEEE Transactions on Neural Networks and Learning Systems, 32, 1, pp.4-24.

Yajima, Y., Kim, S., Chen, J. & Cho, Y.K. (2021) Fast online incremental segmentation of 3d point clouds from disaster sites. In: the 38th International Symposium on Automation and Robotics in Construction (ISARC). Dubai, UAE, International Association for Automation and Robotics in Construction (IAARC).

Yan, X., Ai, T., Yang, M. & Tong, X. (2021) Graph convolutional autoencoder model for the shape coding and cognition of buildings in maps. International Journal of Geographical Information Science, 35, 3, pp.490-512.

Yan, X., Ai, T., Yang, M. & Yin, H. (2019) A graph convolutional neural network for classification of building patterns using spatial vector data. ISPRS Journal of Photogrammetry and Remote Sensing, 150, pp.259-273.

Zhang, S., Tong, H., Xu, J. & Maciejewski, R. (2019) Graph convolutional networks: A comprehensive review. Computational Social Networks, 6, 1, p.11.

Zhang, Z., Cui, P. & Zhu, W. (2022) Deep learning on graphs: A survey. IEEE Transactions on Knowledge and Data Engineering, 34, 1, pp.249-270.

Zhao, R., Ai, T., Yu, W., He, Y. & Shen, Y. (2020) Recognition of building group patterns using graph convolutional network. Cartography and Geographic Information Science, 47, 5, pp.400-417.

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, pp.57-81.