A Graph-Based Computational Tool for Retrieving Architectural Precedents of Building and Ground Relationship (BGR Tool)

Abstract:

The use of neural networks to retrieve relevant images has become mainstream. However, retrieving images that contain specific spatial relationships remains a challenging task. Images alone are not sufficient to fully describe spatial and topological relationships, which are usually better represented as a graph made up of nodes and edges. This paper describes the development of a graph-based computational tool for retrieving architectural precedents that closely match the relationship between a building and its surrounding ground as detected in a designer's project. The tool, titled Building Ground Relationship (BGR), stems from a research project into Graph Machine Learning (GML) that used Deep Graph Convolutional Neural Networks (DGCNNs) to classify building and ground relationships. The neural network was trained using a large synthetic dataset of graphs and optimized through the fine-tuning of its hyperparameters. To verify its performance, a second surrogate model was built using the Deep Graph Library (DGL). The results were nearly identical, thus giving confidence that the model is highly optimized. In the development of the BGR tool, two primary technologies were utilized. In the first instance, the synthetic database was built in Rhino Grasshopper by generating variations of a parametric model. The dual graphs of these models were then automatically generated and exported using the (Blinded for peer review) software library. The second phase involved developing GML models used for predicting the class of the conceptual design, enabling the retrieval of the smaller case study. The results of this research point to the importance of topological representation and machine learning approaches in retrieving and classifying architectural precedents.

Keyword: Deep Graph Machine Learning, Deep Graph Convolutional Neural Networks (DGCNNs), Deep Graph Library (DGL), Computational Tool, Similar Architectural Precedents, Building Ground Relationship (BGR)

1. Introduction

The ground is an integral aspect of a building, and the construction of an object creates a relationship between the two elements (1). This relationship can be in opposition to the context, or it can accept the existence of local conditions. Regardless, the building and ground question remains unavoidable.

It has become evident that the fundamental "shelter" crafted into the earth's crust has evolved into a highly sophisticated building envelope that not one that only separates and protects us from the elements, but also one that allows its inhabitants to interact with and enjoy its surroundings. Carefully considering the relationship between the building and the ground can create synergies between them that enable such interaction and enjoyment. Enhancing the integration between the building and the ground will connect the architecture with the landscape which will reconfigure architecture's position within nature. This study seeks to reveal these building and ground relationships and introduce them to the architect in the early design stages before other architecture, engineering and construction systems are incorporated into the design. Postponing the consideration of the ground and building relationship will negatively impact other architectural systems. For example, the topography and soil makeup of a site can significantly affect several other aspects of the design such as the building's orientation, its access, and the choice and design of its foundational structural system among others. A poorly considered building and ground relationship may result in a building that does not take advantage of what the site offers. Postponing the examination of the building and ground relationship can also be costly and time-consuming as several design decisions and material choices may need to be revised. Obtaining an understanding of the approaches, styles, and similarities between building and ground relationships begins with clustering and classification. Yet, few studies have examined the clustering and classification of architectural forms and their relationship to the ground (2,3). Additionally, no dataset of architectural precedents exists that documents such relationships. Thus, a motivation for this work is to provide a well-structured dataset that will assist the architect and designer when referring to a similar building and ground relationship.

Architects need to have the necessary information about the relationship between buildings and their surroundings to make informed design decisions during the initial stages of design. Doing this manually can be time-consuming, expensive, and prone to errors. However, with the advent of artificial intelligence (AI) and more specifically, graph machine learning (GML), designers can predict the connection between buildings and the ground in the early stages of design through automatic classification. This framework has the potential to introduce comparable precedents into the design process, enabling designers to swiftly determine the effects of their design decisions. However, one drawback of current AI techniques is that they rely on 2D visual representations of building features for their classification, which limits the ability to use 3D information. Encoding a full 3D model for machine learning is challenging and time-consuming. As a result, topological graphs can be used to represent buildings, without compromising the 3D information or incurring the complexity of encoding a complete 3D model. The primary challenge is the ability to derive topological graphs from a conceptual model.

The integration of AI and ML in architecture is essential, and the outcomes of this research will significantly impact architects' education and practice. In fact, numerous architects have expressed their concerns about the necessity for a more effective exploration of the connection between buildings and the ground. These concerns were highlighted during interviews conducted by the researcher, involving more than five architects. The primary objective of these interviews was to delve into the building and ground relationship problem in architectural design. The architects were asked about the challenges they encounter during the early design stage, and their insights were used to evaluate the building and ground relationship taxonomy, content, and visual appearance. Moreover, the researcher aimed to assess the potential benefits and significance of introducing a computational design tool capable of clustering and classifying the building and ground relationship. By conducting semi-structured interviews with the architects, the researcher could delve deeper into the topic and gain valuable insights. The findings from these interviews contribute to the understanding of how architectural practice can be improved through a better understanding of the building and ground relationship. Therefore, by applying the recommended approach resulting from this research, architects can efficiently analyze similar precedents, enhance their designs, and make use of the database.

The subsequent sections of this paper will delve into various aspects related to the paper topic. It will begin by examining the significance of precedents in shaping architectural solutions. Subsequently, the role of generative design and machine learning in aiding the design process will be explored. To provide context, a comprehensive review of relevant prior work in the field will be presented. The methodology employed in this research project will be thoroughly explained to offer insights into the approach taken and the tools utilized. Following that, the paper will present the results obtained from the study. Finally, the discussion will extend to consider potential future work and avenues for further exploration in this domain.

1. Architectural Precedents in Architectural Design

Architectural design can be strengthened by incorporating design precedents and relating them to other aspects of the design. This approach can aid designers in reconsidering their initial design ideas during the early stages. By studying architectural precedents, designers can find solutions to issues that have already been tackled in previous designs . Precedent studies are an essential component of the architectural design process, providing insight into construction methods, material selection, and design concepts. However, there is currently no comprehensive dataset of architectural precedents that places emphasis on the relationship between the building and its surrounding environment.

This study utilizes architectural precedents to provide users with access to case studies that can aid them in reviewing, educating themselves on, and re-evaluating design solutions. Additionally, these precedents can be leveraged for other aspects of design, such as construction or structural design, building materials, building facades, and designing building apertures.

2. Design Aided by Generative Design and Machine Learning

Generative design allows architects to provide accurate and sophisticated architectural solutions in record time (4). This is something that traditional design methods could never achieve (5). Although generative design is not new to the world of architecture, its application has been limited due to its complexity. The process requires architects to analyze a large amount of data to arrive at the most suitable design. This results in many architects not finding generative design useful beyond iconic buildings or large projects . However, by combining machine learning with generative design tools, we can present them in a more practical and simpler way than ever before (6). Architects can now produce highly accurate and complete architectural designs that can play a valuable role in their daily work. This approach can process different types of data related to building codes specific to each region, building relationships with different ground contexts, designing houses, apartment buildings, commercial centers, zoning, and interiors of residential and commercial buildings. It can also be used in the analysis of spaces, relationships between rooms, privacy, wind direction, light, and heat, among others. AI technology can automate generative design processes efficiently, and designers can produce a large number of designs regardless of their skill level or experience . These designs can be manipulated within a short period of time, thanks to the integration of generative design and machine learning .

3. A Review of Studies on the Classification of Architectural Works Using Graph Topological Machine Learning in Architecture Design

According to the following review of several studies, machine learning techniques have typically classified architectural works based on 2D pixels. There has been a long-standing interest in understanding and classifying architectural forms (7). Several studies have shown the effectiveness of quantitative and statistical methods aided by computational tools in morphological research (8). Despite their limited impact on mainstream practices of architectural design, machine learning technologies have shown signs of revolutionizing the recognition and classification of architectural forms. The adoption of these technologies still faces several challenges. Firstly, large, labelled datasets are required for supervised machine learning. Secondly, most machine learning systems use pixels-based 2D image recognition (9–11). Even though this approach may seem reasonable given the available data, primarily plans and drawings, it encounters major limitations. Due to these limitations, most machine learning systems do not understand the semantics of the image they recognize. The study review also revealed a shortage of distributable 3D data since open-source sets are not uniform in their formats, appropriateness, usability, or licenses (12–14).

Despite the availability of 3D datasets, it has become challenging to recognize and classify them. In a recent study (15,16), researchers investigated whether three-dimensional models could identify features based on representations in two dimensions. However, this approach misses the 3D topology incorporated into the data. In a slightly more sophisticated approach, features from a 3D model are encoded as a vector that can act as input to a neural network (17). This approach extracts only a portion of the data, and it must be transformed into a standard input vector. The topological information that can indicate the type of object is ignored.

A promising approach involves using machine learning on graphs (18–20). Several approaches face the limitation of decomposing graphs into small substructures, such as walks or paths, and calculating similarities between graphs based on a summation of attributes. The Deep Graph Convolutional Neural Network (DGCNN), Deep Graph Library (DGL), and Unsupervised Graph-level Representation Learning bypass such restrictions by offering a machine learning model that categorizes graph-based information. These networks prove beneficial because they accept graphs without changing the data into vectors. Using these machine learning methods produces the following benefits: 1) grouping architectural precedents building and ground relationship approaches into similar classes to increase the availability of the architecture precedent resource; 2) classifying 3D conceptual models based on their topological graphs rather than on their 2D representation.

4. Previous Work

The previous work of this project was started in early 2020 (21). Thispaper presented a new approach to machine learning for urban and architectural classification using topological graphs instead of 2D images. A custom software library and a graph convolutional network were used, achieving very accurate results.

In their work, (blinded) utilized architectural topological models that incorporate graph machine learning to model the interplay between buildings and the ground. The authors concentrated on creating a case study for the relationships between buildings and the ground, as well as outlined guidelines for implementing these connections to form a 3D synthetic architectural database. This dataset could then be used to determine the connection between the building and the ground during the initial stages of the design process (22).

In their paper published in (blinded) explored the use of machine learning classification techniques with 3D topological models in the field of architecture. The authors proposed a novel workflow that utilizes deep graph convolutional neural networks to classify 3D prototypes of architectural precedent models based on a topological graph rather than 2D images. By doing so, the system produced highly accurate results, helping designers choose the most suitable form of building ground interaction. The study highlights the advantage of applying non-manifold topology in achieving accurate results (23).

4.1. Contribution of This Work

This paper addresses the gaps in knowledge in two ways. Firstly, it involves a comparison of the DGCNN model with the DGL model. This comparison provides important insights into the strengths and limitations of each model and can help researchers to determine which model is best suited for their specific needs. Secondly, the paper introduces the BGR tool to practitioners and collects their evaluations of its effectiveness. This information is important for understanding how well the tool performs in real-world settings and can help guide further development and improvements. Overall, this paper provides insights and practical recommendations that can inform future research and practice in this area.

5. Methodology and Machine Learning Model

The methodology described in the paper is divided into two distinct parts. The first part involves the generation of 3D topological building and ground relationship datasets. In the second part, this data is used to train a machine learning model. Once the model has been trained, it is then tested for validation purposes. By following this methodology, the authors are able to demonstrate the effectiveness and accuracy of the machine learning model in predicting building and ground relationships in three-dimensional space.

5.1. Generated 3D Topological Building and Ground Relationship Datasets

The dataset described in (22) pertains to the relationship between the topology of three-dimensional buildings and their corresponding ground surfaces. In summary, this dataset is composed of information that illustrates these relationships in a comprehensive manner as follows (Table 1):

The flat ground has three primary relationship forms: separation, adherence, and interlock. This section generates a total of 240 iterations. Separation creates 180 building and ground separated relationship iterations, where the building columns can be on a plinth or set directly on the ground. Adherence iterations produce 24 iterations, and interlock iterations creates a further 36 iterations.

The sloped ground has three primary relationship forms: separation, adherence, and interlock. This section generates a total of 684 iterations. Separation iterations create 540 building and ground separated relationship iterations, where the building columns can be on a plinth or set directly on the ground. Adherence iterations produce 72 iterations, and interlock iterations create a further 72 iterations.

Level ground (Topographical ground) has three primary relationship forms: separation, adherence, and interlock. This section generates a total of 1,242 iterations. Separation iterations create 810 building and ground separated relationship iterations, where the building columns can be on a plinth or set directly on the ground. Adherence iterations produce108 iterations, and interlock iterations create a further 324 iterations.

Ground type	Classes	Categories		No. iteration
Flat Ground	Separation	No plinth	S columns	36
			M columns	36
			L columns	18
		Plinth	S columns	36
			M columns	36
			L columns	18
			Total	180
	Adherence	No plinth		12
		Plinth		12
			Total	24
	Interlock	With the ground		36
			Total	36
Total Flat Ground iteration				**240**

Ground type	Classes	Categories		No. iteration
Sloped Ground	Separation	No plinth	S columns	108
			M columns	108
			L columns	54
		Plinth	S columns	108
			M columns	108
			L columns	54
			Total	540
	Adherence	No plinth		36
		Plinth		36
			Total	72
	Interlock	With the ground		72
			Total	72
Total Sloped Ground iteration				**684**

Ground type	Classes	Categories		No. iteration
Level Ground	Separation	No plinth	S columns	162
			M columns	162
			L columns	54
		Plinth	S columns	162
			M columns	162
			L columns	54
			Total	810
	Adherence	No plinth		54
		Plinth		54
			Total	108
	Interlock	With the ground		324
			Total	324
Total Level Ground iteration				**1242**

Table 1: All the BGR iteration (Flat, Sloped and Level)

5.2. Machine Learning Models

The experiments in this study were conducted on a laptop running the MacOS Catalina 10.15 operating system. The laptop was equipped with an Intel Core i7 Quad-Core CPU running at 2.7GHz and had 16 GB of memory. PyTorch, a deep learning software library for Python, was used to implement DGCNN. Throughout the study, two machine learning models were adopted by the researcher. The first model was an end-to-end deep graph convolutional neural network (DGCNN). For further testing and comparison of results, the second model used was the Deep Graph Library (DGL). DGL is a software library that contains implementations of many models. The purpose of employing DGL was to create a more streamlined workflow and to validate the DGCNN results. By utilizing both DGCNN and DGL, the authors were able to better evaluate the efficacy of each model and determine which approach yielded the most accurate results.

The DGCNN model is structured as follows: The network architecture begins with three graph convolution layers, each comprising 32 neurons. These layers are then concatenated to form a single layer. The SortPooling layer is responsible for sorting the feature descriptors in a consistent order before feeding them into traditional 1D convolutional and dense layers. Furthermore, MaxPooling layers and 1D convolutional layers are added to learn local patterns on the node sequence. Finally, a fully connected layer is included followed by a SoftMax layer to output the final classification. Furthermore, it is worth noting that the DGCNN utilized default parameters in our experiments, with few modifications made to these defaults. To learn more about the specific parameter settings, please refer to (23).

For the Deep Graph Library model (DGL), the researcher used (blinded) in Jupyter Notebook to examine the dataset with different ML models. The DGL model is structured as follows: The network architecture of DGL model starts with three graph convolution layers, each consisting of 32 neurons. These layers are then concatenated to form a unified layer. To ensure consistent ordering, the SortPooling layer is employed to arrange the feature descriptors before passing them to traditional 1D convolutional and dense layers. Additionally, MaxPooling layers and 1D convolutional layers are introduced to capture local patterns within the node sequence. Finally, a fully connected layer is incorporated, followed by a SoftMax layer to produce the final classification output.

5.3. Machine Learning Models Preparation

This experiment utilized a full set of the BGR dataset, which contained a total of 2,136 graphs and 171,232 nodes. The average number of vertices per graph was 80, with the minimum being 20 and the maximum being 258. To prepare for experimentation, the dataset was split into training, validation, and testing sets, with 70% of the graphs used for training and validation and the remaining 30% for testing.

To optimize hyperparameters, the experiment varied several factors, including the number of convolutional layers, the number of neurons in each layer, the number of hidden layers in the final dense layer, as well as the number of epochs, learning rate, and batch size. A subset of the dataset containing 70% or 1,496 graphs was selected for this optimization process. Overall, by adjusting these various hyperparameters, the experiment aimed to achieve the best possible results for the task at hand.

6. Results

6.1. Deep Graph Convolutional Neural Network (DGCNN)

After conducting several experiments using, a combination of incremental and random search, we were able to determine the best model parameters. The optimal configuration for our model included three convolutional layers with 32 neurons each, followed by hidden layers and a final dense layer. We trained the model over 100 epochs using a learning rate of 1-e3 and a batch size of 1. By fine-tuning these parameters, we were able to achieve an accuracy of 99.69% and an average loss of 0.006 (Figure 2). We saved the best-performing model and tested it on the test set to ensure its effectiveness on unseen data. For the unseen data, the researcher used standard accuracy measures and confusion matrices to evaluate the performance of classification models.

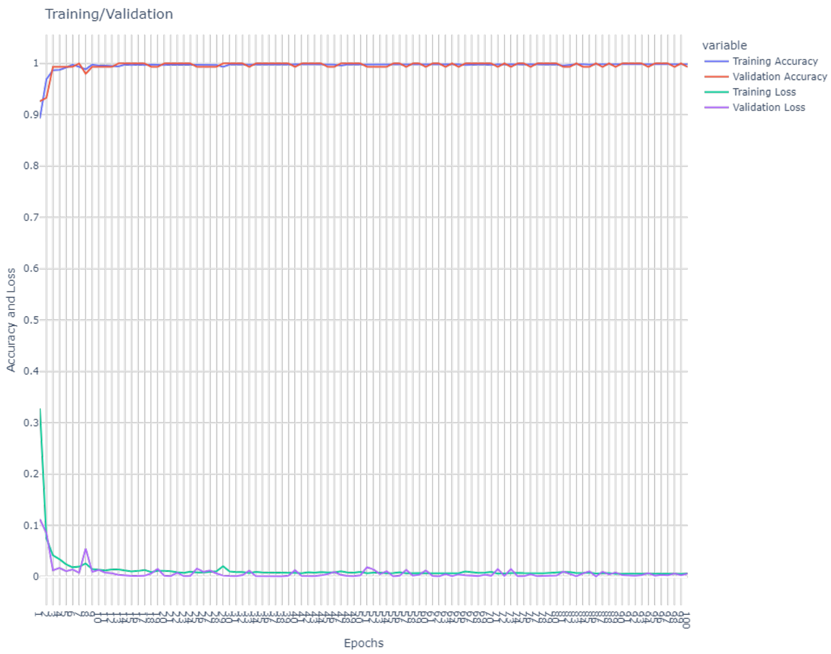


Figure 2: Best DGCNN model performance.

The confusion matrix is a tool used to evaluate the performance of classification models, where (N) represents the number of target classes. It involves comparing the target values with the values predicted by a machine learning model. The resulting matrix allows for a clear comparison between correctly classified examples and those that were misclassified, providing insight into the overall performance of the classifier. With this information, it's possible to identify areas where the model needs improvement and make adjustments accordingly.

The matrix showed 640 predicted data points with high accuracy in all classes (Figure 3). A total of 638 examples were correctly assigned to their respective class, with only two being predicted incorrectly. These two examples were all found within Class 4. The two should have been classified as Class 4 based on their values but were incorrectly labelled as Class 2. It is important to note that the accuracy of Classes 0, 1, and 3 was also very accurate. Despite the limited errors in Class 4, overall, the model performed with high accuracy in all classes (Figure 3).

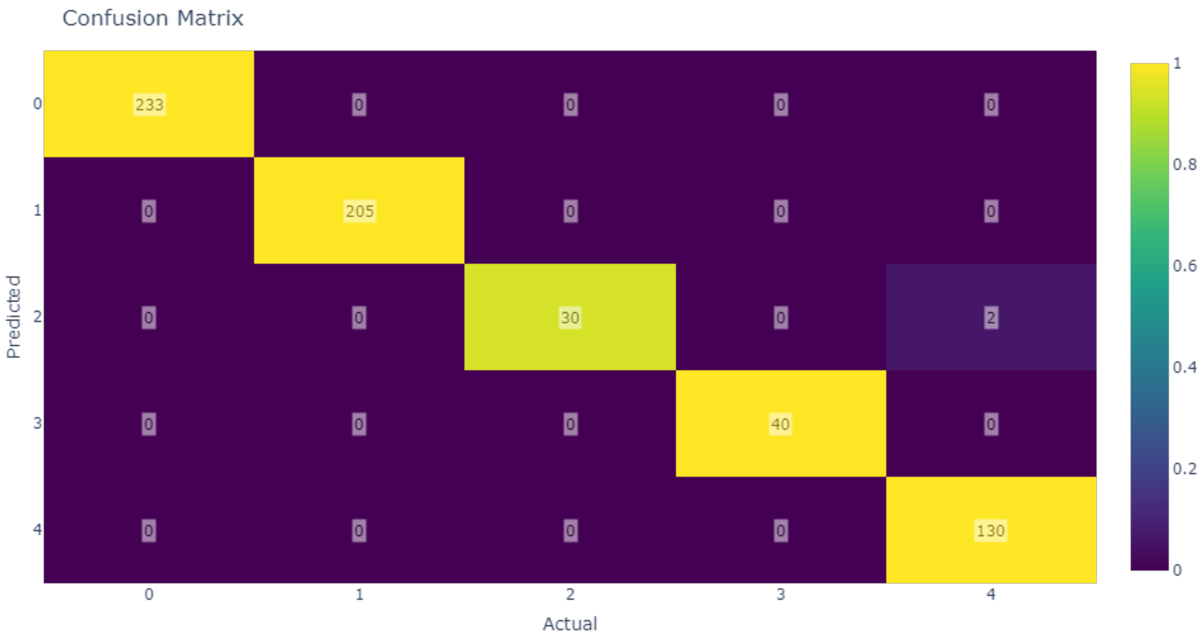


Figure 3: A confusion matrix of 640 unseen datasets (DGCNN) model

6.2. Deep Graph Library (DGL)

This section, the researcher applies a different Deep Graph Neural Network, namely, the Deep Graph Library (DGL). Using different machine learning algorithms, this section evaluates the results of the same dataset. The results of DGL reached 99.8% accuracy performance with 0.14 error loss (Figure 4). The best-performing structure of the DGL model comprised three hidden layers, each with 32 neurons, the Adam optimizer, the SAGEConv convolutional layer type, a train/test split ratio of 80-20%, a final MaxPooling layer, 100 epochs, a batch size of 1, and a learning rate of 0.0001.

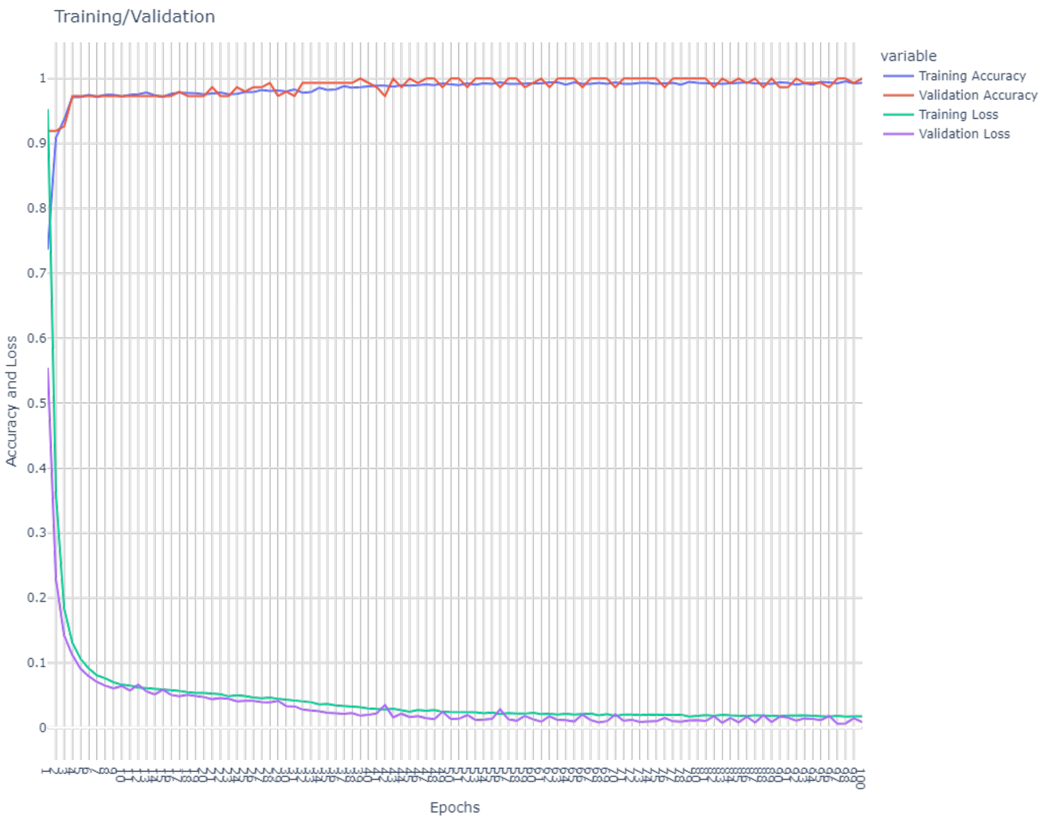


Figure 4: Best DGL model performance.

The best model was saved after training the DGL, followed by testing the DGL model on 640 unseen datasets. The accuracy of the model exceeded 99.8%. In the 640 datasets, the model correctly predicts 639 cases in the right category, while only 1 case fell into an incorrect category. This case was classified as Class 4 however, the right classification of this category was expected to be Class 2 (Figure 5).

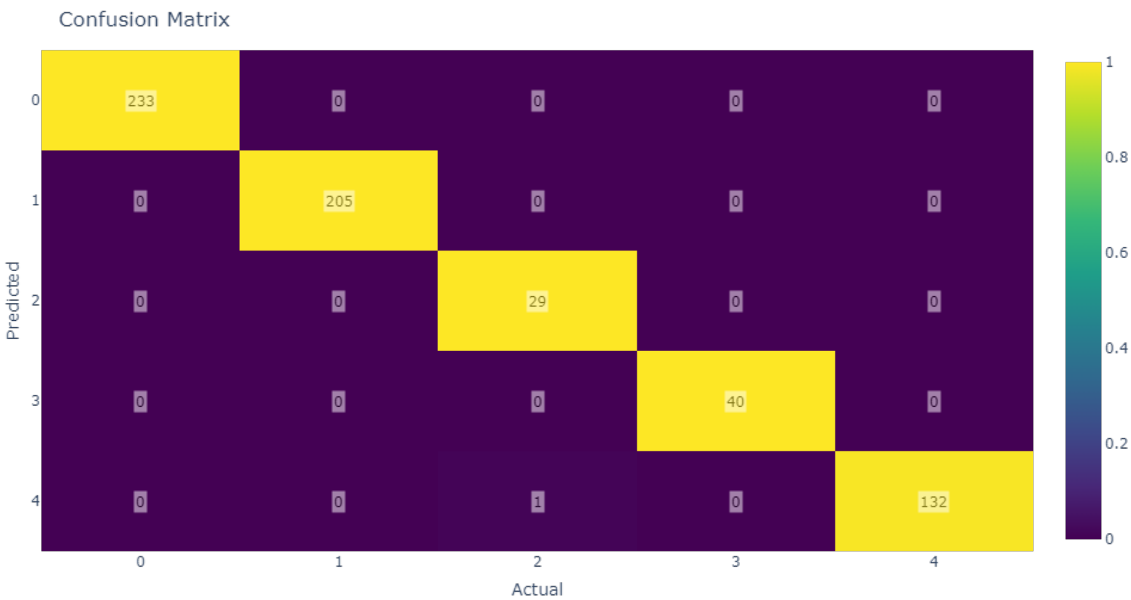


Figure 5: A confusion matrix of 640 unseen datasets (DGL) model

7. Develop Building and Ground Relationship (BGR) Tool

Developing a computational tool with an easy-to-use interface has helped to simplify the workflow process for architects. Early in the design process, the tool assists architects in making informed decisions. A workflow tool has been developed that consists of three major stages: creation, implementation, and retrieval. The building ground relationship object is created, the DGCNN model is implemented to predict the 3D graph of the conceptual design, and similar precedents are retrieved (Figure 6).

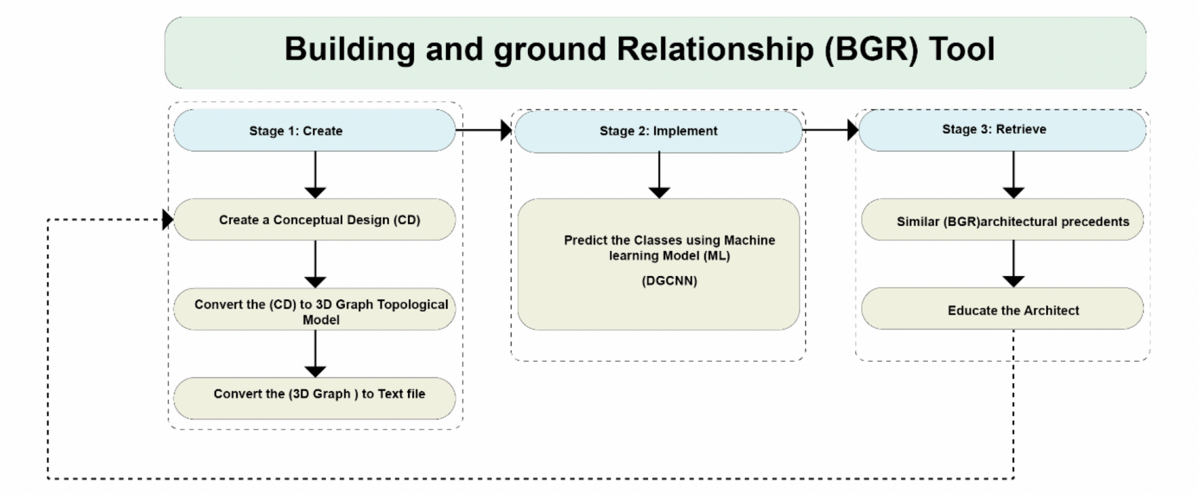


Figure 6: The workflow if the BGR tool

Stage 1: Creation. In this stage, the architects develop a 3D design with simple geometry and then convert it into its dual 3D topological graph. Rhinoceros, Grasshopper, and (blinded) software applications will be utilised to design and produce the graph. Afterwards, the graph is exported to a text file for further processing.

Stage 2: Implementation. In this stage, the saved DGCNN or DGL machine learning model is used to predict the 3D graph of the produced conceptual design. The 3D graph is classified into one of 5 possible classes, which are separation, separation with plinth, adherence, adherence with plinth, and interlock.

Stage 3: **Retrieval.** In this stage**,** similar architecture precedents are retrieved. - With the datasets collected and archived, the architects can obtain not only images of architectural precedents but also the name of the architect, building type, building status, building location, and building period.

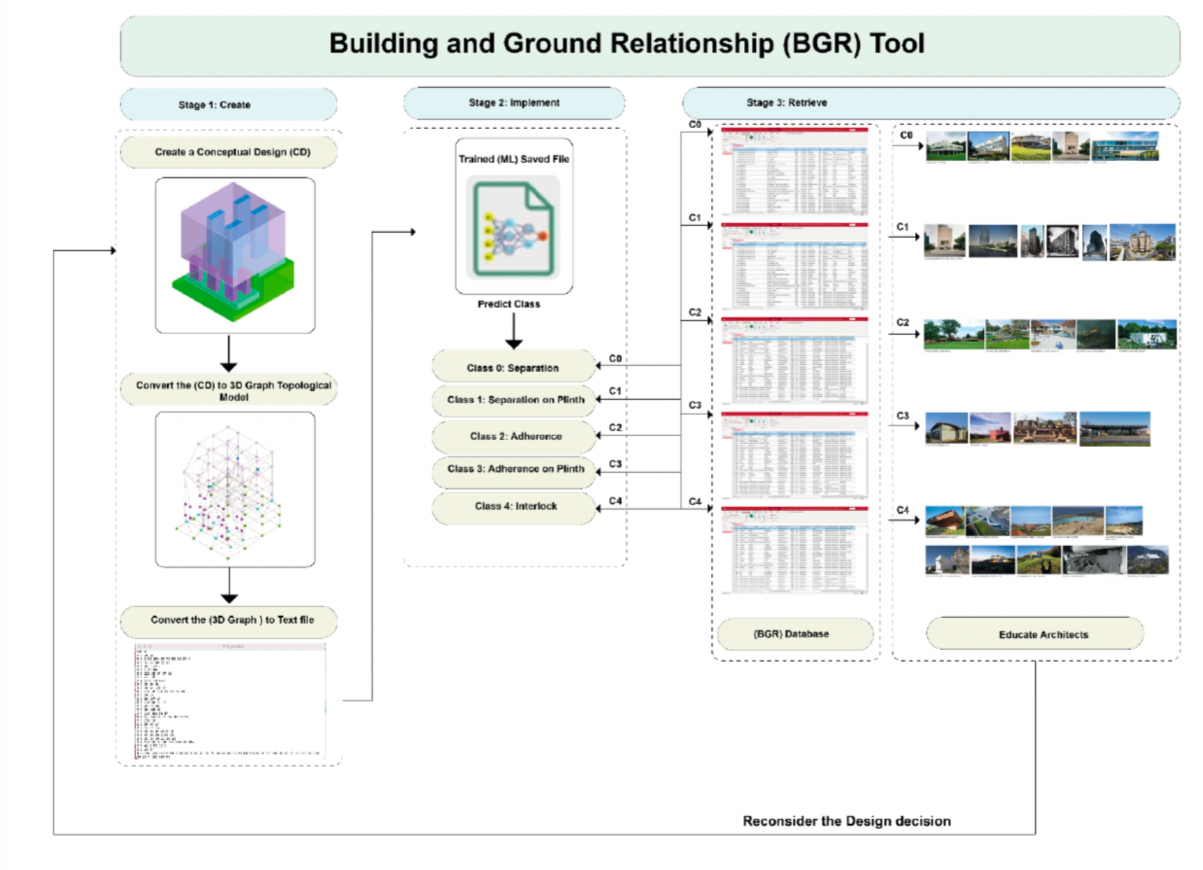


Figure 7: Detailed three stages workflow of the created BGR tool

8. Test the Building and Ground Relationship (BGR) Tool

Twelve participants were recruited to use and evaluate the BGR tool. Tullis and Stetson (2004) suggested that using SUS to analyze a small sample of users (8-14 people) can leave the user reasonably confident that they have accurately evaluated how users perceive a given tool or system (24). (Figure 8) below shows that the level of correct conclusions of the SUS can reach 100% with 12 or more participants. Therefore, the sample size for this SUS questionnaire comprised 12 participants to achieve this level of accuracy.

The experiment aimed to compare the efficacy and performance of a computational design tool between two user groups: professional architects and architecture students. Participants, totalling ten professionals or post-graduate researchers with design experience and two third-year students, were recruited. To ensure consistency, the same conditions were maintained throughout, and a brief tutorial was provided before each session. Participants were briefed on the tool's functionalities and Grasshopper, and they completed an evaluation form post-implementation. This approach allowed for efficient time and budget management while ensuring controlled execution by the researcher.

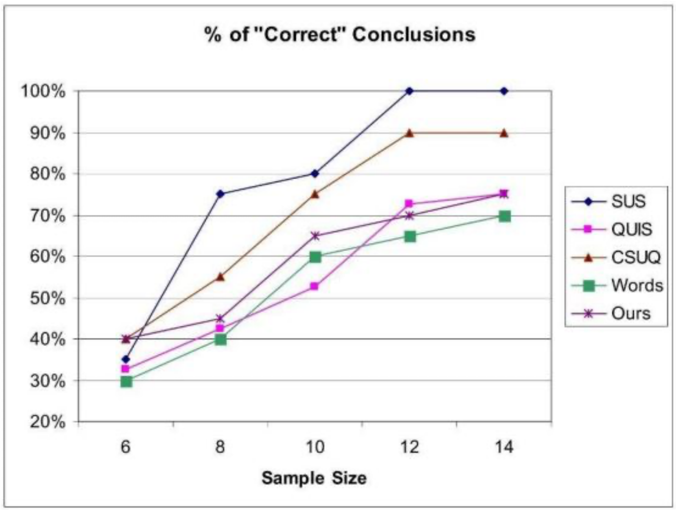


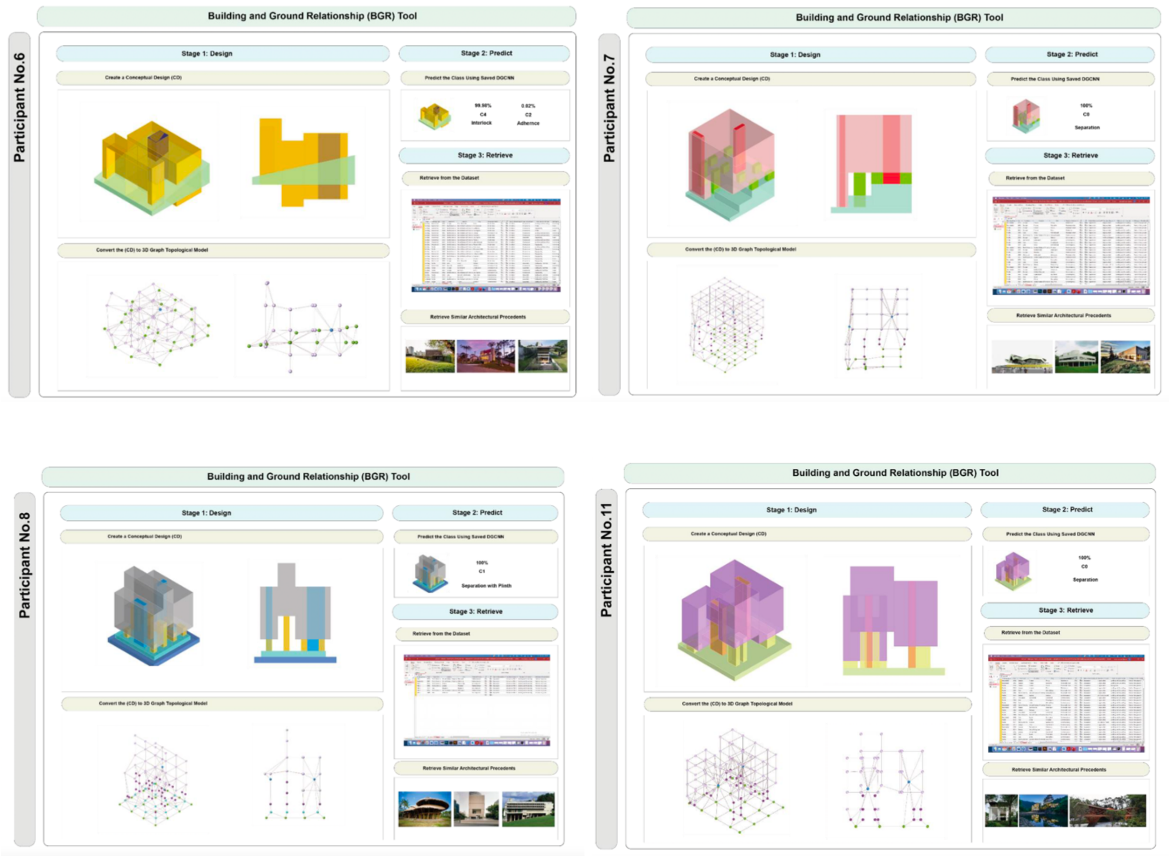
Figure 8: "A Comparison of Questionnaires for Assessing Website Usability" (24)

Initially, the participants were asked to design a building that satisfied the following criteria: 1) Depending on the type of ground, an individual may choose flat ground, sloped ground, or stepped ground, 2) Building objects and cores are not limited in number, 3) There are no restrictions on the height or number of floors, 4) Plinths and columns are required if the participant intends to design the relationship as separation with plinth or adherence with plinth.

The usability and effectiveness of a computational tool are assessed through testing and evaluation following its development.. Since it is a low-cost method of assessing the ease of use of systems, the System Usability Scale (SUS) suits this process (25). The SUS twice evaluated each component of the system separately. The first evaluation assessed the usability of the generative building and ground workflow, while the second evaluation assessed the usability of the ML system. After that, SUS once more assessed the usability of the entire system which both the generative building and ground and ML system.

Based on an evaluation of the tool's usability, it appears that most users found it suitable for designing and retrieving similar architectural precedents when designing their own buildings. The system scored 74.37 out of 100, a rating that indicates that it is a good and acceptable tool for the above purpose. Additionally, the participants indicated their interest in using the BGR tool at an early stage of the design process so that they could examine alternatives that would otherwise go unnoticed during the initial design phase. In comparison to other applications, the BGR tool allows architects and students to complete the design process in a shorter period. The average time taken by all participants to complete the task was 28 minutes.

The results indicate that the model has successfully achieved its intended objective of designing, categorizing, and retrieving similar precedents for the relationship between the building and the ground, as determined by the assessment process and the usability test for the computational tool. Contrary to traditional methods that focus on a single design solution, the BGR tool allows architects to consider multidimensional constraints dynamically and identify similar designs that can reconsider, develop, or modify the first design solution.



9. Conclusion

In this paper, we presented the creation of a computational tool based on graphs. This tool aims to retrieve architectural precedents that accurately reflect the connection between a building and its surrounding ground, as identified in a designer's project. The tool relies on a novel workflow that uses ML on 3D graphs rather than 2D images. We leveraged a sophisticated topology-based 3D modeling environment to derive dual graphs from 3D models and label them automatically. We then exported those graphs to a state-of-the-art deep learning DGCNN and DGL machine learning modes, for comparison and verification. To reach the best accuracy measures, we experimented with different learning rates, the number of epochs, and the number of batches.

Both models achieved high accuracy in classifying the dataset, with the DGCNN model reaching 99.69% accuracy and the DGL model achieving 99.8% accuracy. However, there were slight differences in the specific misclassified examples. The DGCNN model misclassified two examples, both of which fell into class 2 but should have been classified as class 4. In comparison, the DGL model only misclassified one example, which was also classified as class 4 but should have been class 2. Overall, both models performed exceptionally well, with high accuracy rates across all classes. However, the DGL model demonstrated a slightly better accuracy rate and lower error loss compared to the DGCNN model. This suggests that the DGL model may have better generalization ability and improved performance on unseen data. It is worth noting that the findings support the effectiveness of using deep graph neural networks, such as the DGL, in achieving highly accurate classifications. The DGL model's ability to accurately classify the majority of the dataset, including difficult cases like the mis predicted example, showcases its potential in various classification tasks.

By creating a user-friendly computational tool, architects have been able to streamline their workflow process. This tool can be particularly useful during the early stages of design, as it assists architects in making well-informed decisions. The development of a workflow tool involved three key stages: creation, implementation, and retrieval. Initially, architects create a building ground relationship object. Subsequently, they implement the DGCNN model to accurately predict the 3D graph of the conceptual design. Finally, the tool allows architects to retrieve and study similar precedents to further enhance their design process.

Although this study has provided encouraging results, some limitations nevertheless require attention. Despite the tool helping the user achieve the proposed objective, switching between the different software platforms presented some challenges. The generated syntactical 3D graph topological dataset was limited to 2136 elements. The size of the initial set may limit the capabilities of ML models.

Rather than representing a final solution, this research study aims to contribute to the body of knowledge in an ongoing manner. Therefore, future research will focus on adding more semantically rich information to the building and ground data set, such as windows (aperture) environmental analysis, more building context and building setback. This will result in a more comprehensive description of the building and its surroundings. Moreover, applying these approaches to a real 3D BIM dataset, covering diverse research orientations, can provide a more comprehensive understanding of the building and its surroundings. It is intended that future work will focus on applying this approach to different types of data to generalise it. After creating and evaluating the BGR tool, it is essential to further examine how this tool impacts architectural processes and the cognitive process of acquiring knowledge through sensory experiences. This examination will help gain a better understanding of the BGR tool's influence on these processes and provide valuable insights for future work.

References

1. Manferdini E. Revealed Authenticities #1 - angelicalorenzi [Internet]. China Town, Los Angeles; 2016 [cited 2019 Sep 4]. Available from: https://angelicalorenzi.com/Revealed-Authenticities-1
2. Berlanda T. Architectural Topographies: a graphic lexicon of how buildings touch the ground [Internet]. 2014 [cited 2018 Oct 17]. 171 p. Available from: http://search.lib.virginia.edu/catalog/u6268613
3. Porter Z. Assorted-Grounds [Internet]. 2018. Available from: https://www.zacharytateporter.com/assorted-grounds
4. Smorzhenkov N, Ignatova E. The use of generative design for the architectural solutions synthesis in the typical construction of residential buildings. E3S Web Conf. 2021;281:1–9.
5. Antal MR, Dénes L, Vas ZA, Polgár A. Comparative Study of Conventional and Zero-joint Edgebanding. Acta Silv Lignaria Hungarica. 2023;17(1):21–35.
6. Vaidhyanathan V. Using Generative Design and Machine Learning for Faster Analysis Feedback. 2021;(November 2020):0–15.
7. Steadman P, Bruhns HR, Holtier S, Gakovic B, Rickaby PA, Brown FE. A classification of built forms. Environ Plan B Plan Des. 2000;
8. Gil J, Beirão JN, Montenegro N, Duarte JP. On the discovery of urban typologies: Data mining the many dimensions of urban form. Urban Morphol. 2012;16(1):27–40.
9. Shalunts G, Haxhimusa Y, Sablatnig R. Classification of gothic and baroque architectural elements. 2012 19th Int Conf Syst Signals Image Process IWSSIP 2012. 2012;(April):316–9.
10. Yoshimura Y, Cai B, Wang Z, Ratti C. Deep learning architect: Classification for architectural design through the eye of artificial intelligence. Lect Notes Geoinf Cartogr. 2019;249–65.
11. Shalunts, Gayane, Yll Haxhimusa RS. Architectural Style Classification of Building Facade Windows. Lect Notes Comput Sci (including Subser Lect Notes Artif Intell Lect Notes Bioinformatics). 2011;9474:285–94.
12. Collins FC, Braun A, Ringsquandl M, Hall DM, Borrmann A. Assessing IFC classes with means of geometric deep learning on different graph encodings. Proc 2021 Eur Conf Comput Constr. 2021;2:332–41.
13. Gröger G, Plümer L. CityGML - Interoperable semantic 3D city models. ISPRS J Photogramm

Remote Sens. 2012;71:12–33.

14. Ma L, Sacks R, Kattel U, Bloch T. 3D Object Classification Using Geometric Features and Pairwise Relationships. Comput Civ Infrastruct Eng. 2018;33(2):152–64.
15. Sarkar K, Varanasi K, Stricker D. Trained 3d models for CNN based object recognition. VISIGRAPP 2017 - Proc 12th Int Jt Conf Comput Vision, Imaging Comput Graph Theory Appl. 2017;5:130–7.
16. Kasaei H. OrthographicNet: A Deep Learning Approach for 3D Object Recognition in Open-Ended Domains. 2019; Available from: http://arxiv.org/abs/1902.03057
17. Qin F, Li L, Gao S, Yang X, Chen X. A deep learning approach to the classification of. 2014;15(2):91–106.
18. Vishwanathan SVN, Schraudolph NN, Kondor R, Borgwardt KM. Graph kernels. J Mach Learn Res. 2010;11:1201–42.
19. Kriege N, Mutzel P. Subgraph matching kernels for attributed graphs. Proc 29th Int Conf Mach Learn ICML 2012. 2012;2:1015–22.
20. Orsini F, Frasconi P, De Raedt L. Graph invariant kernels. IJCAI Int Jt Conf Artif Intell. 2015;2015-Janua(Ijcai):3756–62.
21. Jabi W, Alymani A. Graph Machine Learning Using 3D Topological Models. SimAUD. 2020;(May):427–34.
22. Alymani A, Jabi W, Corcoran P. Modelling the Relationships between Ground and Buildings Using Architectural Topological Models Utilizing Graph Machine Learning. 6th Int Symp Form Methods Archit. 2022;
23. Alymani A, Jabi W, Corcoran P. Graph machine learning classification using architectural 3D topological models. Simul Trans ofthe Soc Model Simul Int. 2022;
24. Tullis TS, Stetson JN. A Comparison of Questionnaires for Assessing Website Usability ABSTRACT : Introduction. Usability Prof Assoc Conf [Internet]. 2004;1–12. Available from: http://home.comcast.net/~tomtullis/publications/UPA2004TullisStetson.pdf
25. Brooke J. SUS : A Retrospective. J usability Stud. 2013;8(January 2013):29–40.