CHARACTERIZING RESIDENTIAL BUILDING PATTERNS IN HIGH-DENSITY CITIES USING GRAPH CONVOLUTIONAL NEURAL NETWORKS

MUXIN JIA[1], KAIHENG ZHANG[2] and TARO NARAHARA[3]
[1,2,3]*New Jersey Institute of Technology.*
[1]*mj8@njit.edu, ORCID 0000-0002-2213-6027*
[2]*kz25@njit.edu, ORCID 0009-0006-9671-7269*
[3]*narahara@njit.edu, ORCID 0000-0002-2200-1253*

Abstract. In urban morphology studies, accurately classifying residential building patterns is crucial for informed zoning and urban design guidelines. While machine learning, particularly neural networks, has been widely applied to urban form taxonomy, most studies focus on grid-like data from street-view images or satellite imagery. Our paper provides a novel framework for graph classification by extracting features of clustering buildings at different scales and training a spectral-based GCN model on graph-structured data. Furthermore, from the perspective of urban designers, we put forward corresponding design strategies for different building patterns through data visualization and scenario analysis. The findings indicate that GCN has a good performance and generalization ability in identifying residential building patterns, and this framework can aid urban designers or planners in decision-making for diverse urban environments in Asia.

Keywords. Urban morphology, Machine learning, Building pattern classification, Graph convolutional neural networks.

1. Introduction

Urban morphology is the study of how physical elements like street networks, public spaces, plots, and buildings shape and organize urban areas (Kropf, K., 2018). Residential blocks, a significant component of modern urban fabrics, have always been viewed as a classic research domain by urban designers and planners in terms of their geometric characteristics, spatial layout, and topology structure. Particularly for high-density residential areas in Asian cities, where residents are faced with a more crowded and complex built environment than in Western cities, scholars need to have an in-depth understanding of existing residential building patterns that support environmentally appropriate design guidelines. Building patterns, in essence, refer to recurring and observable layouts or configurations of a collection of buildings, which can be distinguished semantically within a block or architectural context (Zhang, et al., 2013). Previous studies of residential building patterns used qualitative research

ACCELERATED DESIGN, Proceedings of the 29th International Conference of the Association for Computer-Aided Architectural Design Research in Asia (CAADRIA) 2024, Volume 2, 39-48.

methods (such as document analysis, field observations, or visual mapping) to capture the nuances of spatial organization and design principles. However, with the development of computational techniques and the advent of the AI era, machine learning methods, specifically artificial neural networks (ANNs), have become more and more important for urban analytics and, as such, were applied to the urban morphological domain to some extent (Yan, X, et al., 2021).

This research introduced a novel spectral-based graph convolutional neural network (GCN), a type of machine learning architecture in the family of ANNs, to examine if neighboring residential buildings have a similar structure or semantic homogeneities. GCN is specifically developed for processing graph-structured data and, as such is appropriate for building pattern classification, especially when it comes to the spatial topology of residential buildings layout (Yan, X, et al., 2019). The purpose of this study is to inform urban design decisions, planning strategies, and preservation efforts through an in-depth understanding of residential building patterns.

As shown in Figure 1, the framework of the present study includes four sections. Section 1 is made up of two basic arrangements. One is to gather datasets for model training and performance validation, and the other is to define physical and semantic features from both single buildings and groups of buildings. In section 2, we extract the centroid of each building's footprint as a node and connect all the nodes within a residential block to construct a simple undirected graph. Also, the features defined in Section 1 are attached to each node for graph representation. Section 3 is the core content of this experiment, involving model initiation, model training, model validation, and result interpretation. In section 4, we put forward spatial design suggestions for each category of residential building pattern from the urban design and planning perspective, which is a useful application to real-world practice.

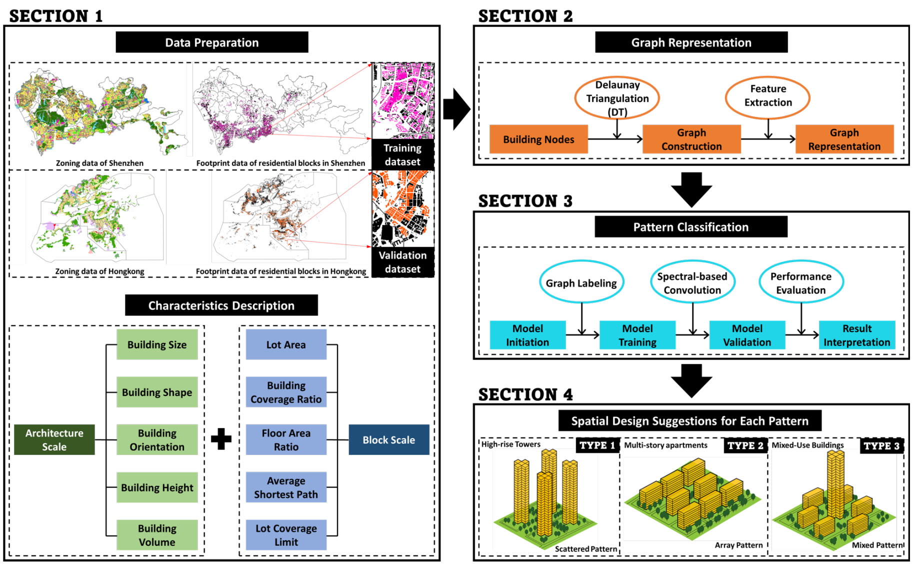

Figure 1. Research framework of classifying residential building patterns with GCN.

2. Methodology

Shenzhen is located to the south of Guangdong Province in China and has a border with Hong Kong. Hong Kong is traditionally recognized as Asia's financial center, whereas Shenzhen has gradually emerged as a significant participant in global manufacturing and trade. Despite the two cities having divergent administrative systems and development trajectories, their close geographic proximity and limited land availability create similarities in the built environment, particularly in high-density residential areas. Therefore, we picked these two cities as study areas by training GCN models on the footprint dataset in Shenzhen and validating model performance on the footprint dataset in Hong Kong.

2.1. DATA PREPARATION

To obtain reliable datasets for the experiment, we downloaded the zoning maps and the latest geographic datasets through their official open-data portal. After selecting all types of residential plots from the zoning maps, along with the intersecting building footprints of the two cities, we got 2196 residential blocks in Shenzhen and 1742 residential blocks in Hong Kong as shown in Figure 2. Note that the 2196 residential blocks along with 97601 individual buildings are input samples for the GCN model training, testing, and relevant analysis. In addition, the geo-dataset is pre-processed on the ArcGIS platform and then exported to a JSON file for further analysis in Python.

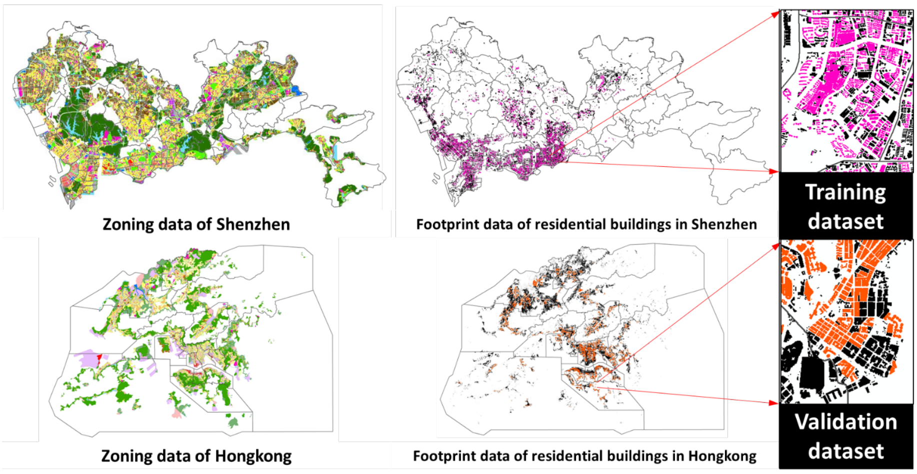

Figure 2. Visualization of zoning maps and residential-building footprints.

2.2. CHARACTERISTICS DESCRIPTION

In the field of urban modeling, characteristics description refers to the artificial definition of the input node features, which can be represented by either socio-economic or physical attributes of spatial objects. Prior research has extensively employed geometric homogeneity to depict building shapes or utilized rectangle algebra to portray the adjacent relationship between building locations. However, there is currently no clear standard for which indicators can serve as perfect semantic representations on maps. Therefore, this paper focuses on variable selection at different scales after conducting a thorough literature review. As shown in Table 1, there are 5 variables including building size, building shape, building orientation, building height, and building volume at the architectural scale, similarly, there are 5 variables including lot area, building coverage ratio, floor area ratio, average shortest path, and lot coverage limit at the block scale (except for the position index). Each index is measured with a specific formula to describe the corresponding variable mathematically. Figures 3 and 4 display the geometric signs of all 14 indices at the architectural scale and the block scale, respectively.

Table 1. Input node features for describing semantic characteristics of residential buildings.

Category	Node Features	Index	Description
Intrinsic Attribute	Position	Centroid	Arithmetic means of coordinates of all building footprint vertices, where N is the number of vertices.
Architecture-Scale	Building Size	Area	Coverage area of a single building.
		Perimeter	Perimeter of the outer edge of the building's exterior walls.
	Building Shape	Compactness	A measure that combines height, footprint area, and form to quantify the overall compactness of a building.
		Concavity	Area ratio of the building to its convex hull (CH)
		Overlap Index	The area ratio of the intersection and union of a building with its equal-area circles (EAC)
	Building Orientation	SBRO	Orientation of the smallest bounding rectangle
	Building Height	AGL	Building height above ground level
	Building Volume	Volume	The total three-dimensional space occupied by the building.
		CVI	The ratio of the building volume to the cube of its height.
Block -Scale	Lot Area	LA	The total horizontal area within the lot lines of a lot, exclusive of streets
	Building Coverage Ratio	BCR	The ratio of all building's area to the lot area
	Floor Area Ratio	FAR	The ratio of all building's gross floor area to the lot area
	Average Shortest Path	LUC	The average distance between all pairs of nodes in the graph.
	Lot Coverage Limit	LCL	The maximum percentage of a lot that can be covered by buildings

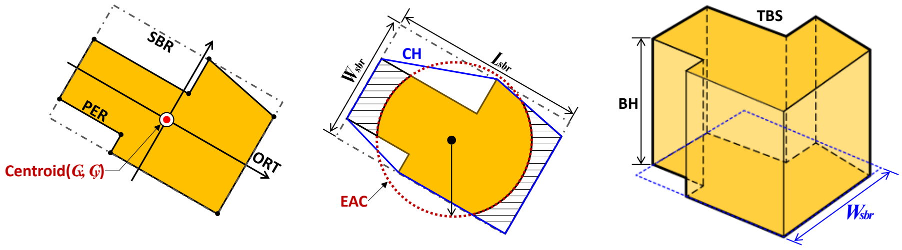

Figure 3. Illustration of indices for the individual building at the architecture scale.

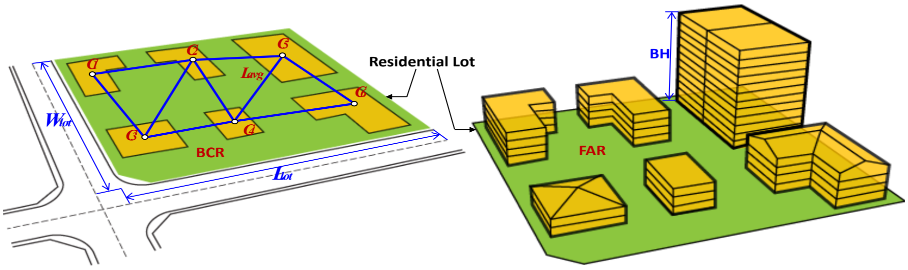

Figure 4. Illustration of indices for the clustering buildings at the block scale.

2.3. GRAPH REPRESENTATION

At this stage, we need to represent the relationships between graph nodes (residential buildings) in a 2D space based on their proximity or connectivity and store the graph-based data in computer memory. This process is referred to as graph representation. As the edge orientation is irrelevant, the network connected through the centroids of a set of building footprints can be represented as a simple undirected graph. Mathematically, a simple undirected graph G can be represented as $G = (V, E, W)$, where V is the vertex (node) set, E is the edge set, and W is an adjacency matrix that stores the edge weights between each pair of vertices. Each vertex has 10 variables (measured by 14 indices) denoting the graph signals.

To model a graph structure programmatically, we need to consider how nodes are connected and how statistical distances are measured between buildings. Existing research has shown that there are many methods to create proximity graphs according to graph theory, such as Relative Neighborhood Graph (RNG), Minimum Spanning Tree (MST), K-Nearest Neighbor Graph (K-NN), etc. Different methods have different computational complexities and may be more suitable for specific types of data distributions or applications (Adamczyk, J., 2022). In this paper, we choose Delaunay Triangulation (DT) as the connection method, since it has been widely used in GIS for terrain modeling, network analysis, and spatial analysis (Fig. 5).

Statistical distances, also known as the weights of graph edges, are generally measured through shape homogeneity and spatial proximity. Shape homogeneity quantifies the similarity of building polygons, which has been implicitly represented by the input node features (Yang, Huang, 2023). Hence, only spatial proximity requires additional measurement. Note that we calculated the pairwise Euclidean distance between two centroids of building nodes in Python based on the formula shown in Table 1.

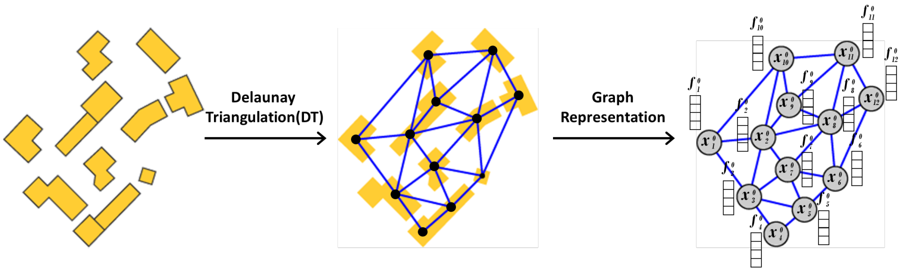

Figure 5. Graph representation of a building group based on Delaunay Triangulation.

Graph labeling is another preparatory work for the following graph classification task. As GCN is a supervised machine learning architecture, it requires assigning labels to the entire graph based on domain knowledge before training. In this paper, we gathered a group of volunteers with expertise in architecture and urban design to manually identify all residential building graphs in Shenzhen as three typical patterns, as shown in Figure 6. Note that because there is no unanimous criterion for the taxonomy of building patterns, we divided the building groups into 3 patterns from the perspective of visual cognition proposed by urban study precursors. Type 1 is the scattered pattern, where only high-rise residential buildings are scattered on the land plot. Type 2 is viewed as the array pattern, with only multi-story residential buildings neatly arranged on the land plot. Type 3 is the mixed pattern, as a mix of high-rise buildings and multi-story buildings is laid out on the land randomly. Finally, a total of

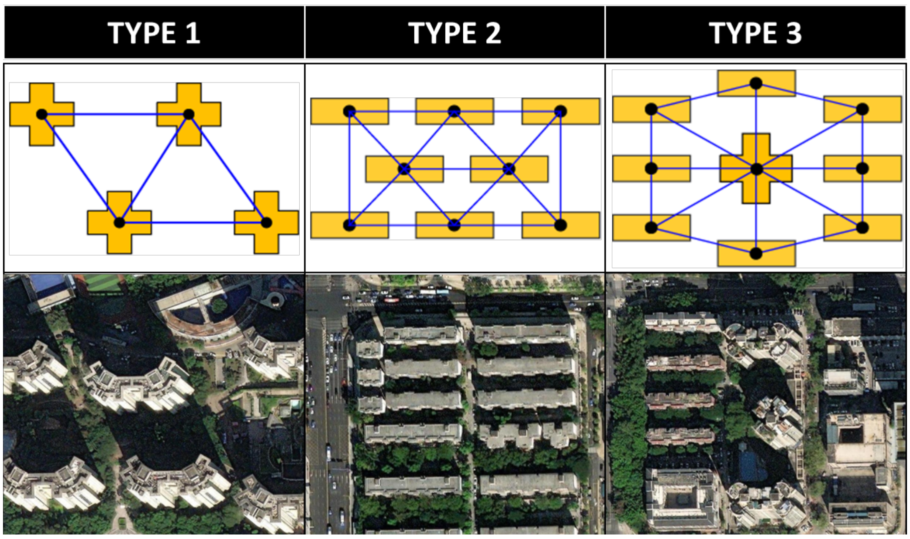

Figure 6. Three typical types of residential building layout in China.

1698 building groups were manually labeled as type 1, 425 as type 2, and 73 as type 3.

3. Graph convolutional neural network

3.1. OVERVIEW OF GCN ARCHITECTURE

Convolutional graph neural networks (ConvGNNs) have become popular in recent years, as this type of machine learning model is efficient in performing convolution operations on graph-structured data, especially in scenarios where relationships matter.

The general architecture of ConvGNNs consists of several key components that work together to exchange information between target nodes and their k-hop neighborhood. The process of message passing is called node embedding, aiming to learn a low-dimensional representation for each node. More importantly, this process will be repeated many times to update the node embedding itself so that the global and structural information of the graph can be captured successfully (Chen, W.,2021). Since 2014, computer scientists have created more than a dozen architectures for ConvGNNs (such as SCNN, Chebnet, GCN, GraphSage, GAT), and most of them adopt the abovementioned logic (Zhao, R., et al, 2020).

In this paper, GCN is used for the building graph classification experiment, due to its high computational efficiency, interpretability, and performance. As shown in Figure 7, the architecture of GCN is composed of an input layer, multiple graph convolutional layers, a pooling layer (readout), two fully connected layers, and an output layer. Note that the input graph consists of an adjacency matrix and a node feature matrix, the activation function we used is ReLu, the graph convolution operation is spectral-based, and the pooling is mean.

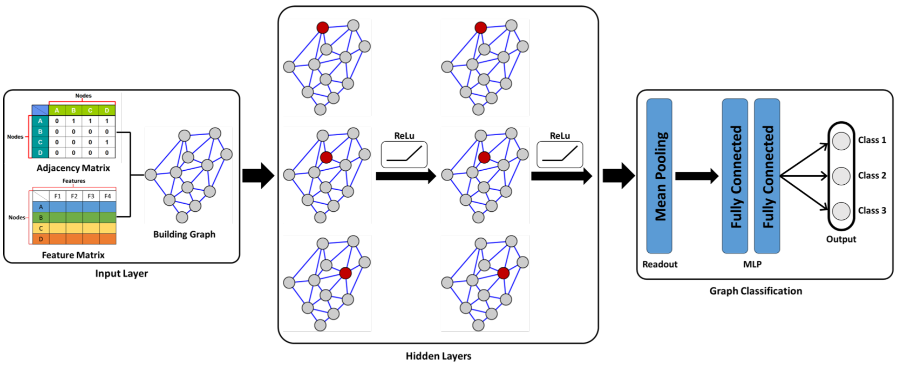

Figure 7. Visualization of GCN architecture for graph classification.

3.2. SPECTRAL-BASED GRAPH CONVOLUTION

As we know, the architecture of graph convolution can be either spectral-based or spatial-based. Unlike spatial approaches, which focus on capturing local information within the immediate neighborhood of each node, spectral approaches can leverage the graph Laplacian's eigenvalues and eigenvectors to capture global information about the graph structure (Wei, Z., et al., 2016). Furthermore, spectral graph convolution has the advantage of being computationally efficient, especially for graphs with a relatively small number of nodes (Du, S., et al., 2018). For these reasons, the spectral method is more suitable for the data we used in the experiment.

In essence, spectral approaches stem from graph signal processing, where the convolution operation transforms the graph signals (node features) into the spectral domain using the graph Fourier transform, and then transforms the signals back to the spatial domain using the inverse graph Fourier transform. The graph convolution operation in the spectral domain can be expressed using the graph Fourier transform as follows:

$$\hat{f}^{(l+1)}(\lambda_i) = \sigma(\sum_{j=1}^{N} \hat{f}^{(l)}(\lambda_j) \cdot \hat{g}\theta(\lambda_i, \lambda_j))$$

Where:

- $\hat{f}^{(l)}$ are the graph Fourier coefficients of the input signal at layer l.
- $\hat{g}\theta$ is the filter in the frequency domain parameterized by θ.
- λ_i and λ_j denote the i-th and j-th eigenvalues of the graph Laplacian matrix.
- The graph Laplacian is L=D−A (where D is the degree matrix, A is the adjacency matrix)

Note that the filter function $\hat{g}\theta(\lambda_i, \lambda_j)$ determines how information is aggregated across different eigenvalues during the convolution operation.

4. Experiment and results

4.1. MODEL TRAINING AND EVALUATION

In this experiment, we try to test model performances of GCN architecture by different convolution layers. Before model training, we set the hyper-parameters of learning rate, dropout probability, and mini-batch size of 0.1, 0.5, and 200 separately. Also, we define 0.8 as the threshold to accept correctly predicted classes with an output probability greater than this value. The input data is divided into training sets, test sets, and validation sets according to the ratio of 6:2:2.

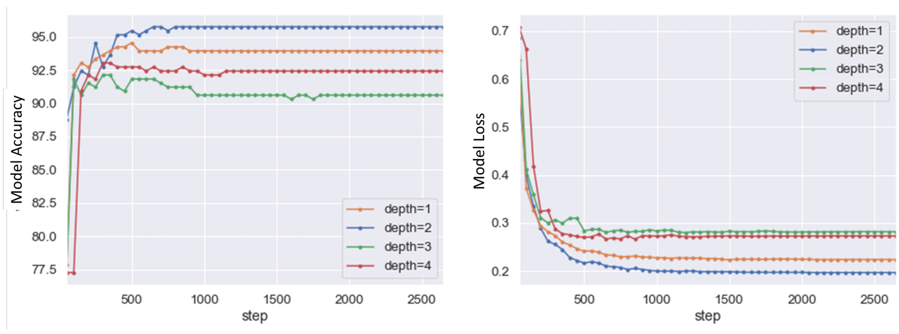

Figure 8. Visualization of GCN architecture for graph classification.

Figure 8 shows the accuracies and loss changes of the validation dataset over training time based on 4 different architectures. Obviously, after 500 steps of training, almost all the models reach stability, and the model architecture with 2 graph convolutional layers has the best performance.

As shown in Table 2, for model 2, the accuracies of training sets and test sets are 97.95% and 94.91%, and the loss values are 0.13 and 0.18, respectively. The close and high accuracies between training and test sets imply that the GCN model has good generalization capacity and can be used for other datasets.

Table 2. Prediction accuracy comparison.

Model Name	Test Accuracy	Train Accuracy	Test Loss	Train Loss	Fit Time (s)
depth=1	90.00	95.16	0.29	0.17	19.12
depth=2	**94.91**	**97.95**	**0.18**	**0.13**	**81.67**
depth=3	90.00	94.53	0.33	0.20	167.08
depth=4	89.09	93.85	0.37	0.17	226.55

4.2. GENERALIZATION ABILITY ANALYSIS

In this section, we ran another experiment on the Hong Kong dataset to verify the generalization capacity of the pre-trained GCN model. After inputting the whole residential footprint for classification, each residential plot was assigned a unique typological label defined previously. To visually display the classification results, we marked the residential areas in different colors on ArcGIS, as shown in Figure 9. It is important to mention that the accuracy has also achieved a high value of 93.77%, indicating that the model possesses a strong ability to generalize.

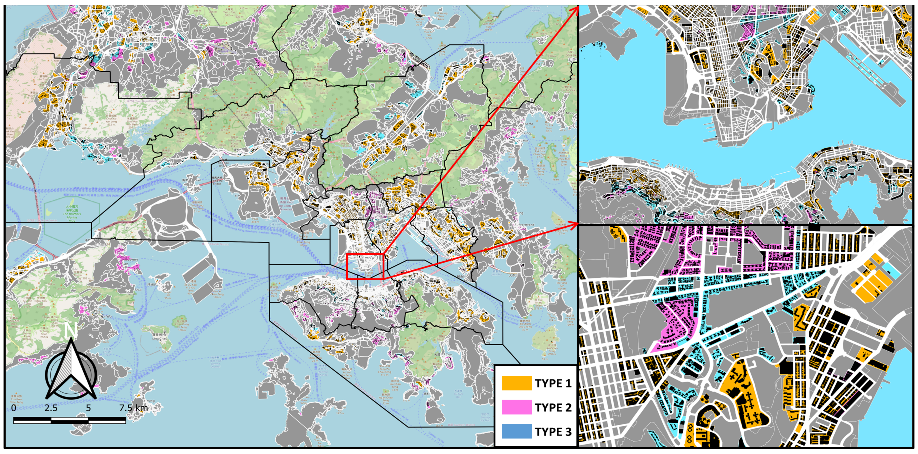

Figure 9. Visualization of prediction result of Hong Kong dataset.

5. Urban design suggestions for different types of building layouts

It is crucial to apply the results of machine learning to practical tasks in the field of urban design and planning. Therefore, this paper uses GCN to automatically classify urban residential spaces and proposes design and planning suggestions for three different building patterns.

For type 1, careful attention should be paid to sunlight and walkability, since the proximity of tall buildings on both sides of a street can create a visually enclosed or constrained space, i.e., the canyon effect. For type 2, residential blocks with only low houses lack the diversity of land uses, housing types, and amenities, leading to a homogenous and boring environment. Hence, we recommend increasing recreational facilities and featured landscapes. For type 3, we suggest residents should focus more on community safety due to the complex built environment and limited open spaces.

6. Conclusion

Classification of perceptual patterns of residential building groups through supervised learning facilitates the interpretation of urban morphology with spatial semantic structures, thereby contributing to informing real-world planning decisions and improving the overall quality of urban life. The present study has demonstrated the potential of the GCN classifier to automatically identify the semantic relationship of group buildings through experiments on large urban datasets. More importantly, the findings make us confirm that the application of machine learning technologies brings data-driven insights, automation, and predictive capabilities to urban design and planning processes.

References

Kropf, K. (2018). The handbook of urban morphology. John Wiley & Sons.

Zhang, X., Ai, T., Stoter, J., Kraak, M. J., & Molenaar, M. (2013). Building pattern recognition in topographic data: examples on collinear and curvilinear alignments. Geoinformatica, 17, 1-33.

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), 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, 259-273.

Adamczyk, J. (2022). Application of Graph Neural Networks and graph descriptors for graph classification. arXiv preprint arXiv:2211.03666.Schölkopf, B., Smola, A. J., Williamson, R. C., & Bartlett, P. L. (2000). New support vector algorithms. Neural computation, 12(5), 1207-1245.

Chen, W., Wu, A. N., & Biljecki, F. (2021). Classification of urban morphology with deep learning: Application on urban vitality. Computers, Environment and Urban Systems, 90, 101706.

Du, S., Shu, M., & Feng, C. C. (2016). Representation and discovery of building patterns: A three-level relational approach. International Journal of Geographical Information Science, 30(6), 1161-1186.

Wei, Z., Guo, Q., Wang, L., & Yan, F. (2018). On the spatial distribution of buildings for map generalization. Cartography and Geographic Information Science, 45(6), 539-555.

Yang, L., & Huang, W. (2023). Representation and assessment of spatial design using a hierarchical graph neural network. Automation in Construction, 147, 104727.

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), 400-417.