Intelligent Urban Parking Recommendation System using Spatiotemporal Graph Neural Network, Large Language Model and Vision Assistant

Muhtada Zuhair Ali [1], Jamshid Bagherzadeh[2] Parviz Rashidi-Khazaee[3*]

[1] Department of Electrical and Computer Engineering, Urmia University, Urmia, Iran
[2] Professor, Department of Electrical and Computer Engineering, Urmia University, Urmia, Iran
[3] Assistant Professor, Department of Information Technology and Computer Engineering, Urmia University of Technology, Urmia, Iran

*** Corresponding author email address**: p.rashidi@uut.ac.ir

Received: 2025-10-01 **Revised:** 2026-02-10 **Accepted:** 2026-02-17 **Initial Publish:** 2026-05-31 **Final Publish:** 2026-09-01

Abstract

The growing imbalance between urban parking demand and available capacity leads to excessive cruising, traffic congestion, and unnecessary emissions, which collectively degrade urban traffic efficiency. To address this challenge, this paper proposes multimodal Spatiotemporal Graph Neural Network, Large Language Models and Vision Algorithm (STGNN-LLMaVA), framework for intelligent urban parking recommendation that jointly models visual perception, semantic context, and dynamic spatiotemporal dependencies. Parking-slot occupancy is inferred from surveillance images using You Only Look Once version 12 (YOLOv12). In parallel, the Large Language model and Vision Assistant (LLMaVA) generate compact semantic and temporal descriptions of the scene, capturing factors such as congestion, visibility, and surrounding activity. These visual and language-derived features are embedded into parking-node representations and processed by a GraphKAN–Temporal Transformer backbone. In this backbone, a Kolmogorov–Arnold Network models nonlinear spatial interactions, while a causal Temporal Transformer captures evolving availability patterns to produce top-k parking recommendations. Experiments conducted on four real-world parking datasets demonstrate that STGNN-LLMaVA consistently outperforms strong Graph Neural Network (GNN)- and Large Language Model (LLM)-based baselines, achieving improvements of up to 24.7% in HitRate@10, 10.5% in NDCG@10, and 9.8% in MRR@10. These results indicate that integrating vision-based occupancy estimation, language-driven contextual reasoning, and spatiotemporal graph learning provides an effective, scalable, and data-efficient solution for sustainable smart parking management.

Keywords: *Smart Parking, Urban Traffic Management, Parking Recommendation, Spatiotemporal Graph Learning, Kolmogorov–Arnold Networks, Temporal Transformer, LLMaVA, YOLOv12*

How to cite this article:

Zuhair Ali, M., Bagherzadeh, J., & Rashidi-Khazaee, P. (2026). Intelligent Urban Parking Recommendation System using Spatiotemporal Graph Neural Network, Large Language Model and Vision Assistant. Management Strategies and Engineering Sciences, 8(5), 1-17.

1. Introduction

Intelligent Transportation Systems (ITS) are crucial for sustainable urban mobility. However, efficient parking management remains a persistent challenge. Rapid urbanization and the increase in private vehicle ownership have intensified the gap between parking demand and supply. This leads to excessive cruising, congestion, higher emissions, and time loss. Studies show that up to 30% of urban traffic results from drivers searching for parking, making parking inefficiency a major contributor to congestion and environmental impact [1]. Consequently, intelligent parking guidance and recommendation systems have become essential in smart-city infrastructures [2, 3].

Early smart-parking solutions relied on IoT sensors, infrastructure monitoring, or rule-based systems to detect slot availability [4, 5]. These approaches are effective in controlled environments but often involve high costs and limited scalability. Vision-based methods emerged as cost-efficient al[6]ternatives, using surveillance cameras and deep learning to detect occupancy [6]. Nevertheless, most vision-only methods are sensitive to illumination, occlusion,

and adverse weather. They also typically treat parking as a static perception task, ignoring temporal and contextual dynamics.

Accurate parking recommendations require more than perception. Practical systems must model spatial relationships among slots, temporal occupancy patterns, and contextual cues such as traffic and surrounding urban activities [7]. Existing research often addresses only a subset of these factors, reducing robustness and adaptability in real-world scenarios.

To address these limitations, this paper proposes STGNN-LLMaVA, a unified multimodal framework for intelligent urban parking recommendation. YOLOv12 infers parking-slot occupancy from surveillance images. LLMaVA generates concise semantic and temporal descriptions of the scene. These features are embedded into a structured parking graph and processed by a GraphKAN–Temporal backbone. Kolmogorov–Arnold decomposition enables nonlinear spatial reasoning, while temporal modeling captures evolving availability patterns. Together, these components produce accurate top-k parking recommendations.

The main contributions of this work are as follows:

- A unified multimodal parking recommendation framework combining vision, language-driven context, and spatiotemporal graph modeling
- Nonlinear spatial reasoning using Kolmogorov–Arnold decomposition for data-efficient modeling of complex interactions
- Context-aware recommendations leveraging LLMaVA-generated semantic and temporal features
- A dual reasoning strategy balancing accuracy, computational efficiency, and deployment flexibility
- Extensive evaluation on four real-world datasets demonstrating improved ranking accuracy, scalability, and robustness

The remainder of this paper is organized as follows. Section II reviews related work, Section III presents theoretical background, Section IV describes the methodology, Section V reports experimental results, and Section VI concludes the paper.

1. RELATED WORK

Spatiotemporal graph neural networks (STGNNs) have become a central approach for modeling urban mobility because they jointly capture spatial correlations and temporal dependencies in traffic systems. Early studies demonstrated that dynamic spatiotemporal graph convolution can adapt to non-stationary traffic patterns by updating node relationships over time, enabling more realistic citywide traffic flow prediction [8]. These findings established dynamic connectivity as a key requirement for learning evolving interactions in large-scale urban road networks and complex systems.

Subsequent research focused on strengthening temporal representation learning within dynamic graph frameworks. Zhang proposed dynamic graph convolutional networks with explicit temporal representation learning to model long-range dependencies in traffic sequences [9]. Building on this direction, Zhang et al. combined spatiotemporal transformers with graph convolutional networks to capture global temporal patterns alongside localized spatial interactions [10]. Jiang and Liu further introduced adaptive spatial–temporal coupling mechanisms to improve robustness under rapidly changing traffic conditions and disturbances [11]

Beyond single-graph formulations, researchers explored multi-view and multi-graph extensions to enhance robustness. Huang et al. proposed a multi-view dynamic graph convolutional network that integrates heterogeneous spatial perspectives to improve traffic flow prediction accuracy [12]. Ye et al. extended this idea by incorporating multiple dynamic graphs that explicitly encode traffic accidents, enabling event-aware modeling under disruptions [13]. These studies demonstrated that complementary graph structures are critical for representing complex urban dynamics in realistic large city scenarios.

Several survey studies synthesized the rapid progress of graph-based traffic modeling. Zheng et al. provided a comprehensive survey of dynamic graph neural networks, categorizing methods by graph construction strategies and temporal modeling techniques [14]. Jiang et al. reviewed advances in graph neural networks for traffic forecasting and highlighted the dominance of structured sensor-based numerical inputs [15]. Both surveys emphasized that perception-driven and semantic-aware modeling remains largely unexplored in current intelligent transportation system research literature today.

In the parking domain, spatiotemporal feature fusion has been used to predict occupancy and availability trends. Zhang et al. fused multifaceted spatiotemporal features to model parking lot dynamics, but relied on handcrafted representations and treated slots independently [16]. Ma et al. reviewed urban parking prediction studies and concluded that most methods remain unimodal and forecasting-oriented rather than recommendation-driven [17].

Consequently, spatial interactions and contextual semantics are insufficiently modeled for real-world adaptive parking decision making tasks.

Table 1. COMPREHENSIVE COMPARISON OF RELATED STUDIES

Limitation	Rec.	Multimodal	Semantic	Graph	Temporal	Vision	Year	Ref.
No vision or semantic context	✗	✗	✗	✓	✓	✗	2021	[8]
No multimodal perception	✗	✗	✗	✓	✓	✗	2025	[9]
Limited multimodal fusion	✗	△	✗	✓	✓	✗	2025	[10]
Not designed for parking recommendation	✗	✗	✗	✓	✓	✗	2023	[11]
Traffic-oriented only	✗	✗	✗	✓	✓	✗	2023	[12]
No vision or language semantics	✗	✗	✗	✓	✓	✗	2023	[13]
Survey only; no multimodal framework	✗	✗	✗	✓	✓	✗	2025	[14]
Lacks vision–language integration	✗	✗	✗	✓	✓	✗	2023	[15]
No graph reasoning or semantics	✓	✗	✗	✗	✓	✗	2024	[16]
Review; no unified framework	✗	✗	✗	✗	✓	✗	2024	[17]
No structured spatiotemporal reasoning	✗	✓	✓	✗	✗	✓	2024	[18]
Not applied to ITS or parking	✗	✓	✓	✗	✗	✓	2025	[19]
Not transportation-related	✗	✗	✓	✗	✗	✗	2023	[20]
Not spatiotemporal or parking-related	✗	✗	✗	✗	✗	✗	2024	[21]
–	✓	✓	✓	✓	✓	✓	2025	This work

Recent advances in vision–language models and large language models have enabled high-level semantic understanding from visual data. Xu et al. introduced LVLM-eHub, a comprehensive benchmark for evaluating large vision–language models across perception and reasoning tasks [18]. Han et al. proposed LLMaVA-GM, a lightweight multimodal architecture that improves efficiency while preserving reasoning capability [19]. However, such models lack structured spatiotemporal learning mechanisms for modeling dynamic urban mobility systems and parking processes over time and space jointly.

Related semantic modeling efforts have appeared in other application domains. Okmi et al. reviewed mobile phone data analytics for crime applications, focusing on data taxonomies rather than spatiotemporal structure [20]. Dai et al. proposed a machine learning framework for detecting zero-day attacks without spatial context [21]. In contrast, the proposed STGNN-LLMaVA framework unifies visual perception, vision–language reasoning, and spatiotemporal graph learning to enable robust, interpretable, and context-aware parking recommendations in complex urban environments with structured decision support capability.

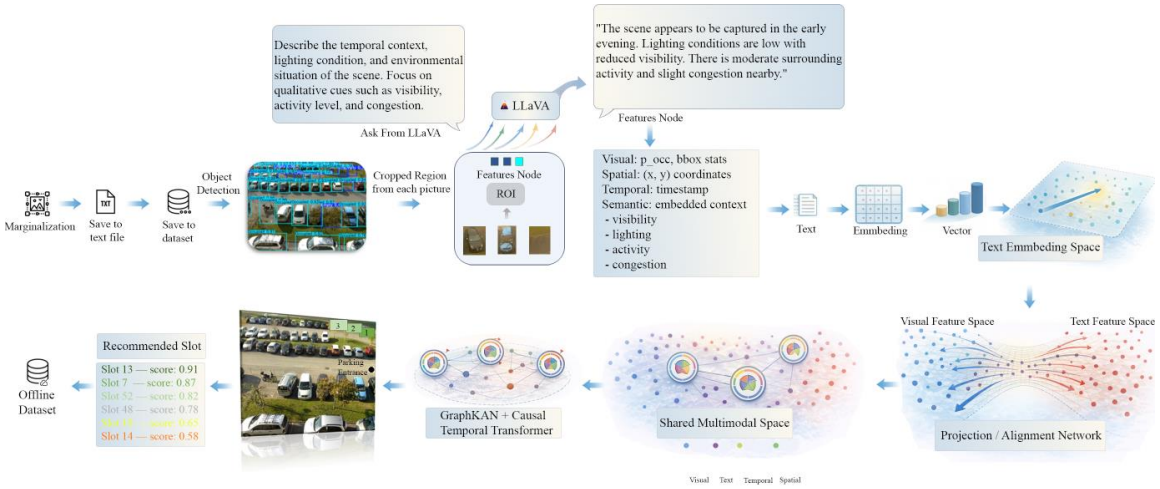


Figure 1. Implementation Pipeline.

2. Background

This section presents the core components of the STGNN-LLMaVA framework. It highlights how multimodal perception, semantic reasoning, and spatiotemporal graph modeling are integrated. These components collectively enable adaptive and interpretable urban parking recommendations.

2.1. Multimodal Learning for Parking Recommendations

Multimodal learning integrates visual, textual, temporal, and spatial information into a unified representation. In smart parking, it captures vehicle appearances and contextual urban patterns, overcoming unimodal limitations [22]. The STGNN-LLMaVA framework combines three complementary modalities. YOLOv12 extracts visual occupancy features from surveillance images. LLMaVA generates semantic and temporal descriptions, including congestion, visibility, and local activity. GraphKAN–Temporal Transformer models nonlinear spatial interactions and evolving temporal dynamics across parking slots. These features are embedded into parking-node representations and fused into a spatiotemporal graph. This fusion enables adaptive, interpretable, and accurate parking recommendations.

$$\widehat{y_i^t} = F_{\substack{GraphKAN-\\Temporal}}(f_{YOLO(I^t)}, f_{LLaVA(I^t)}, [G^{t-k}, \dots, G^t]) \quad (1)$$

2.2. YOLOv12 for Visual Occupancy Detection

YOLOv12 provides precise and robust vehicle localization under diverse lighting and environmental conditions [23]. For a given RGB image $I = \mathbb{R}^{H\times W\times 3}$, the model outputs a set of bounding boxes:

$$YOLO12(I) \rightarrow \{(B_i, S_i, C_i)\}_{i=1}^{N} \quad (2)$$

Here, B_i represents the bounding box coordinates, S_i the confidence score, and C_i the class label. The detected regions are cropped and forwarded to LLMaVA for semantic enrichment. This provides a strong visual foundation for multimodal reasoning.

2.3. LLMaVA for Vision-Language Understanding

LLMaVA serves as a vision–language module that generates concise captions for each detected parking slot [24]. For a cropped image $l_i \in \mathbb{R}^{H'\times W'\times 3}$, the model produces a textual description:

$$T_i = LLaVA(l_i)\, \epsilon\, V^* \quad (3)$$

The captions are then embedded into a fixed-length feature vector using a pretrained language embedding function $\Phi: V^* \rightarrow \mathbb{R}^d$:

$$f_i = \Phi(T_i) = \Phi(LLaVA(l_i)\, \epsilon\, \mathbb{R}^d) \quad (4)$$

These embeddings enrich the graph nodes by linking visual perception to interpretable semantic reasoning.

2.4. GraphKAN–Temporal Transformer for Spatiotemporal Reasoning

The GraphKAN–Temporal Transformer encodes nonlinear spatial interactions among parking slots and captures causal temporal dynamics [25]. The urban layout is an undirected graph $G = (V, E)$, with nodes as parking slots and edges representing spatial or semantic proximity. Node features combine GraphKAN spatial aggregation with temporal evolution captured by the causal Temporal Transformer.

$$h_i^{(l,t)} = \sigma\Big(\sum_{j\in N_s(i)} \phi_s{}^{(l)}\big(h_j^{(l-1,t)}\big) + \sum_{k\in N_t(i)} \phi_t{}^{(l)}\big(h_k^{(l-1,t-1)}\big) + \psi^{(l)}\big(h_j^{(l-1,t)}\big)\Big) \quad (5)$$

3. Methodology

This section presents the formulation of the proposed STGNN-LLMaVA framework. It integrates visual detection, language reasoning, and spatiotemporal graph learning for intelligent parking recommendation. Fig. 1 shows the four main stages of the system: preparation, annotation, semantic-temporal enhancement, and graph-based recommendation.

3.1. Dataset Preparation

The CNRPark+EXT dataset [26] was used for training due to its wide view and detailed labels. For robustness testing, three additional datasets—SPKL [27], Indoor [28], and Parking_ROIs_GoPro [29]—provided diverse lighting, weather, and spatial conditions for evaluating detection stability.

3.2. Data Annotation

After labeling [30] and training, YOLOv12 inferred new frames. Detected slots were mapped to graph nodes, with occupancy encoded as node attributes. Each parking slot p_k was represented as four coordinate points:

$$p_k = \{(x_{ki}, y_{ki})\}_{i=1}^{4}, where\ x_i \in [0, W], y_i \in [0, H] \quad (6)$$

Occupancy was determined using two complementary heuristics:

(1) **IoU-based method:** classifies a slot as occupied when overlap with a detected vehicle box exceeds threshold α:

$$E_1(p_i, b_j) = \begin{cases} 1, if\ IoU(p_i, b_j) \geq \alpha \\ 0,\ otherwise \end{cases} \quad (7)$$

(2) **Euclidean-distance method:** estimates occupancy from centroid distance when overlap is minimal:

$$E_1(p_i, b_j) = \begin{cases} 1, if\ \rho(p_i, b_j) \leq min(p_k) \\ 0,\ otherwise \end{cases} \quad (8)$$

Integrating both methods ensures balanced accuracy under occlusion and varied camera angles [31]. For reference consistency, Table 2 lists the mathematical notations used in this framework.

3.3. *Vision-Language Augmented Graph Reasoning*

To integrate spatial reasoning with semantic understanding, STGNN-LLMaVA combines YOLOv12 detections with LLMaVA-generated semantic-temporal captions within a unified graph processed by the GraphKAN–Temporal Transformer. After YOLOv12 localizes parking slots, each cropped slot is passed to LLMaVA. The model generates semantic-temporal attributes, including occupancy, vehicle type, color, timestamp, and contextual scene information such as congestion, visibility, and local activity.

Semantic attributes produced by LLMaVA that are irrelevant or noisy (e.g., lighting or weather cues) are selectively filtered and validated before integration [32]. As shown in Fig. 2, the resulting features constitute enriched node attributes stored as graph annotations.

3.4. *LLM-Augmented Graph Construction:*

Each parking slot is modeled as a graph node enriched with visual, semantic, and temporal descriptors to capture both local appearance and contextual cues. Graph connectivity remains unchanged, preserving spatial relationships among slots. LLMaVA-generated embeddings enhance semantic expressiveness, while irrelevant captions are filtered out. This selective retention enables the model to infer meaningful contextual relations without modifying the underlying graph topology [32].

3.5. *LLM-STGNN Embedding Harmonization:*

The GraphKAN–Temporal Transformer learns spatiotemporal dependencies across multimodal parking

Figure 2. LLM-Augmented Graph Method

Table 2. DESCRIPTION OF MATHEMATICAL SYMBOLS

Symbol / Term	Description	Eq.
I^t	Input RGB image frame at time *t*.	(1), (2)
$f_{YOLO}(I^t)$	Visual features extracted from YOLOv12.	(1)
$f_{LLaVA}(I^t)$	Semantic–temporal features from LLMaVA.	(1)
$\{G^{t-k}, \dots, G^t\}$	Sequence of recent *k* spatiotemporal graphs.	(1)
$F_{GraphKAN-(.)}$ $_{Temporal}$	GraphKAN–Temporal Transformer for multimodal spatiotemporal graphs	(1)
y_i^t	Predicted suitability score for slot *i*.	(1)
B_i	Bounding box of detected object *i*.	(2)
S_i	Confidence score (occupancy probability).	(2)
C_i	Class label (1 = occupied, 0 = empty).	(2)
l_i	Cropped region of slot *i*.	(3)
T_i	LLMaVA-generated caption for slot *i*.	(3)
V, V^*	Vocabulary and textual domain of LLMaVA.	(3)
$\phi(T_i)$	Language embedding function.	(4)
f_i	Node feature vector of slot *i*.	(4)
$h_j^{(l,t)}$	Hidden feature of node *j* at layer *l*, time *t*.	(5)
$(N_s(i), N_t(i))$	Sets of spatial and temporal neighbors of node *i*.	(5)
$(\phi_s^{(l)}, \phi_t^{(l)})$	Spatial and temporal mapping functions.	(5)

$\psi^{(l)}(.)$	Residual nonlinear mapping.	(5)
$\sigma(.)$	Activation function (ReLU, GELU).	(5)
$p_k = \{(x_{ki}, y_{ki})\}_{i=1}^{4}$	Coordinates of *k*-th parking quadrilateral.	(6)
(W, H)	Frame width and height.	(6)
$E_1(p_i, b_j)$	Occupancy evaluation between slot and box.	(7), (8)
$IoU(p_i, b_j)$	Intersection-over-Union ratio.	(7)
α	IoU threshold for occupancy.	(7)
$\rho(p_i, b_j)$	Euclidean distance between centroids.	(8)
$min(p_k)$	Minimum side length for distance threshold.	(8)
R^d	*d*-dimensional embedding space.	(4)
TP/FP	True / False Positive detections	(9)
AP_i	Average Precision of class *i*	(11)
n	Number of evaluated IoU thresholds or classes	(11)
S	Set of evaluation samples	(12),(14)
y^s	Ground-truth preferred items for sample *s*	(12)
$\hat{y}^s$	Predicted top-k ranked items for sample *s*	(12)
$I(.)$	Indicator function (1 if condition holds; 0 otherwise)	(12),(13)
N_k	Normalization factor for *NDCG@k*	(13)
$rank_i$	Rank position of the first relevant item	(14)
k	Cutoff threshold for top-k ranking metrics	(12)-(14)

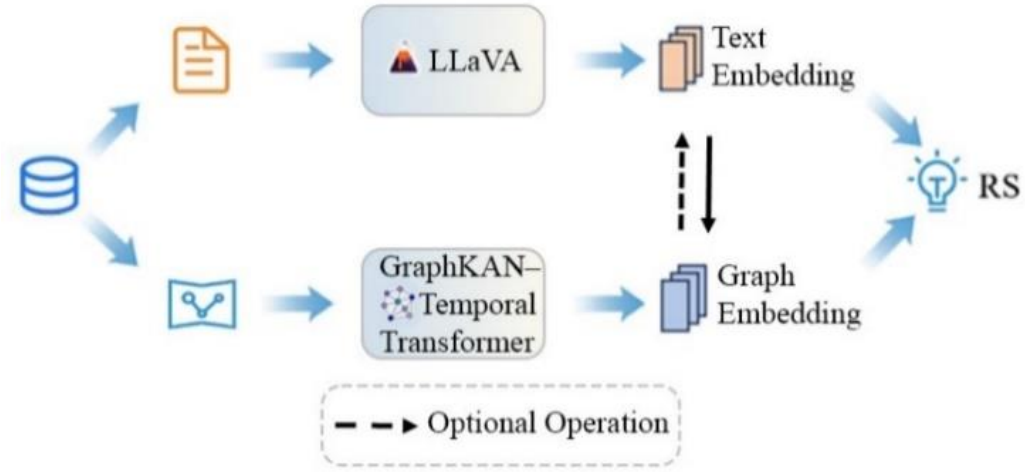


Figure 3. Alignment of Embeddings Between LLMaVA and GraphKAN–Temporal Transformer.

graphs. LLMaVA-generated semantic embeddings are first aligned with visual-spatial node features through a projection network, producing unified multimodal node representations. GraphKAN models nonlinear spatial interactions among parking slots, while a causal Temporal Transformer captures temporal dynamics across graph sequences. Fig. 3 shows the overall spatiotemporal learning architecture.

3.6. LLM-Guided Reasoning Policies:

Two reasoning configurations were explored [32]:

(A) LLM-as-Enhancer:

First, LLMaVA-derived semantic embeddings are harmonized and fused with the visual and temporal node features. They are then processed by the GraphKAN–Temporal Transformer, capturing structured spatiotemporal dependencies and improving both interpretability and predictive accuracy. Fig. 4A shows this multimodal fusion and learning pipeline.

(B) LLM-as-Predictor:

In this lightweight mode, LLMaVA performs recommendations independently by relying on prompt-based inference derived from structured graph descriptors, allowing the model to operate without auxiliary modules while maintaining contextual awareness. This design enables efficient, scalable zero-shot reasoning and fast decision-making across diverse scenarios. Fig. 4B shows the lightweight prompt-based inference mode.

3.7. STGNN-LLMaVA for Intelligent Parking Recommendation:

The final stage unifies the three components into an end-to-end multimodal recommender. During the Semantic Enrichment Phase, YOLOv12 detects occupied and empty slots. LLMaVA then generates descriptive captions, which are used to populate the graph node features.

In the Graph-Based Reasoning Phase, the GraphKAN–Temporal Transformer processes the dynamic spatiotemporal graph $G = (V, E)$. It captures nonlinear spatial interactions and temporal availability patterns.

The system outputs:

- the most suitable available slot,
- its geographic coordinates, and
- the optimal route from the driver's position.

This pipeline ensures robust performance across diverse urban conditions, delivering interpretable, scalable, and adaptive parking recommendations.

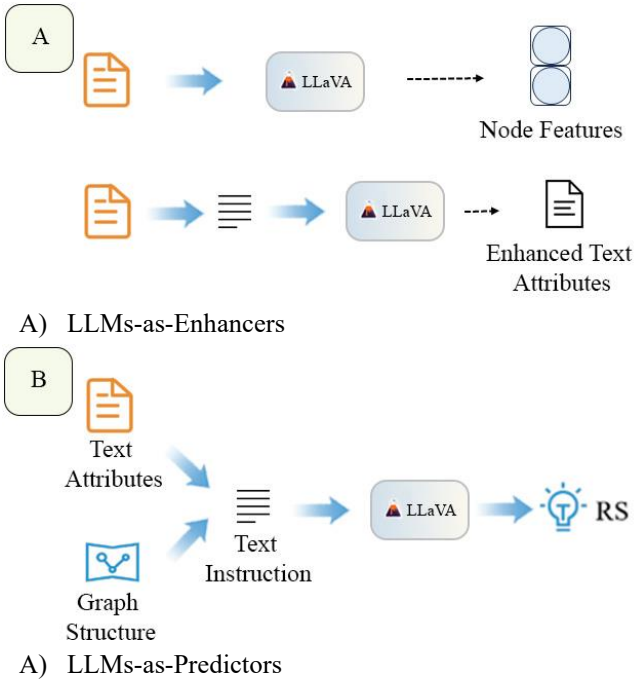


Figure 4. Comparison of (A) LLM-as-Enhancer vs. (B) LLM-as-Predictor

4. Experimental Results and Evaluation

This section evaluates the proposed system through a structured pipeline. It defines the evaluation metrics, analyzes the accuracy and robustness of vision models, assesses the impact of data augmentation, compares trade-offs in LLM integration, and examines cross-dataset performance to demonstrate generalization across different environments.

4.1. Evaluation Metrics

This subsection introduces the evaluation criteria used to systematically assess the performance of the proposed framework. Metrics for the vision-based detection module are first presented to quantify slot-level perception accuracy. Ranking-based metrics for the parking recommendation system are then introduced to evaluate retrieval effectiveness and decision quality.

4.1.1. Metrics for Vision Models:

The performance of the YOLOv12 visual detector was evaluated using four widely accepted metrics [31]:

- Precision: Measures the accuracy of detected occupied parking slots

$$precision = \frac{TP}{TP+FP} \quad (9)$$

- Recall: Measures the detection rate of truly occupied parking slots

$$Recall = \frac{TP}{TP+FP} \quad (10)$$

- mAP@50: Evaluates mean average precision at a fixed IoU threshold of 0.50

$$mAP50 = \frac{1}{n}\sum_{i=1}^{i=n} AP_i \quad (11)$$

- mAP@50–95: Computed as the mean average precision across IoU thresholds ranging from 0.50 to 0.95.

4.1.2. *Metrics for Recommendation System:*

For the recommendation module, ranking-based metrics, including HitRate@k, NDCG@k, and MRR@k (k ∈ {5, 10, 20}) [32], were used to evaluate retrieval accuracy and ranking quality.

- HitRate@k: Evaluates whether the top-k recommendations include at least one true preferred parking slot

$$HitRate@K = \frac{\sum_{s\in S} I(y^S \cap \hat{y}^S \neq \Phi)}{|S|} \quad (12)$$

- NDCG@k: Evaluates the quality of top-k ranking by assigning higher weights to relevant items at higher ranks

$$NDCG@k = \frac{1}{N_k} \sum_{i=1}^{k} \frac{2^{I(\hat{y}^S \epsilon y^S)} - 1}{\lim_{2}(i+1)} \quad (13)$$

- MRR@k: Evaluates how early the first relevant item appears in the top-k ranked list

$$MRR@K = \frac{1}{|S|} \sum_{i=1}^{|S|} \frac{1}{rank_i} \quad (14)$$

These metrics provide a consistent basis for comparing recommendation strategies in the subsequent analyses.

2. *Vision Model Performance Analysis*

The vision models were evaluated for their ability to accurately detect parking slots using standard metrics, including Precision, Recall, and mean Average Precision (mAP) at different IoU thresholds [32].

4.2. *Comparative Model Performance:*

As summarized in Table 3, YOLO-World consistently achieved the best overall detection performance across all metrics. Its high mAP values indicate precise localization and robust detection capabilities.

While RF-DETR showed limited precision and recall, YOLOv11 and YOLOv8 achieved moderate accuracy. YOLOv12 demonstrated strong and stable performance, confirming its potential for real-world applications.

Overall, YOLO-based architectures, particularly YOLO-World and YOLOv12, provided superior performance and stability, making them reliable candidates for smart-parking systems.

Table 3 presents a quantitative comparison of various vision models on the CNRPark+EXT dataset. YOLO-based detectors, especially YOLO-World and YOLOv12, consistently outperformed transformer-based models such as RF-DETR across all evaluation metrics.

Fig. 5 illustrates YOLOv12 predictions for both occupied and vacant parking slots in the CNRPark+EXT dataset. The results are reported as mean values with standard deviation error bars, highlighting performance stability after 100 training epochs.

As shown in Fig. 6, models trained for 25 epochs failed to reliably detect empty spaces, whereas models trained for 50 epochs showed partial improvement. The 100-epoch model accurately recognized both occupied and empty slots. This result confirms a positive correlation between training duration and detection precision.

Table 3. PERFORMANCE COMPARISON OF VISION MODELS (CNRPARK+EXT DATASET, MEAN ± STD OVER 3 RUNS)

Model	Epoch	mAP50-95	mAP50	Recall	Precision
YOLO11	25	0.446 ± 0.011	0.747 ± 0.014	0.681 ± 0.016	0.775 ± 0.012
	50	0.522 ± 0.009	0.841 ± 0.010	0.776 ± 0.013	0.883 ± 0.010
	100	0.543 ± 0.008	0.875 ± 0.009	0.818 ± 0.011	0.891 ± 0.009
YOLO8	25	0.509 ± 0.012	0.816 ± 0.011	0.789 ± 0.015	0.841 ± 0.010
	50	0.531 ± 0.010	0.869 ± 0.008	0.803 ± 0.012	0.891 ± 0.009
	100	0.546 ± 0.009	0.883 ± 0.007	0.843 ± 0.010	0.880 ± 0.008
YOLO-world	25	0.536 ± 0.011	0.857 ± 0.012	0.814 ± 0.013	0.901 ± 0.011
	50	0.552 ± 0.008	0.891 ± 0.009	0.820 ± 0.011	0.936 ± 0.010
	100	0.552 ± 0.007	0.911 ± 0.008	0.882 ± 0.010	0.873 ± 0.009
RT-DETR	25	0.729 ± 0.010	0.893 ± 0.009	0.873 ± 0.012	0.762 ± 0.010
	50	0.548 ± 0.010	0.893 ± 0.009	0.896 ± 0.011	0.871 ± 0.010
	100	0.548 ± 0.009	0.909 ± 0.009	0.885 ± 0.010	0.860 ± 0.009
RF-DETR	25	0.499 ± 0.013	0.827 ± 0.011	0.575 ± 0.016	0.493 ± 0.014
	50	0.526 ± 0.011	0.860 ± 0.010	0.595 ± 0.014	0.519 ± 0.012
	100	0.531 ± 0.010	0.878 ± 0.009	0.593 ± 0.012	0.523 ± 0.011
YOLOv12	25	0.534 ± 0.010	0.875 ± 0.009	0.826 ± 0.013	0.905 ± 0.011
	50	0.529 ± 0.009	0.856 ± 0.008	0.813 ± 0.011	0.873 ± 0.010
	100	0.531 ± 0.008	0.896 ± 0.008	0.844 ± 0.010	0.935 ± 0.009

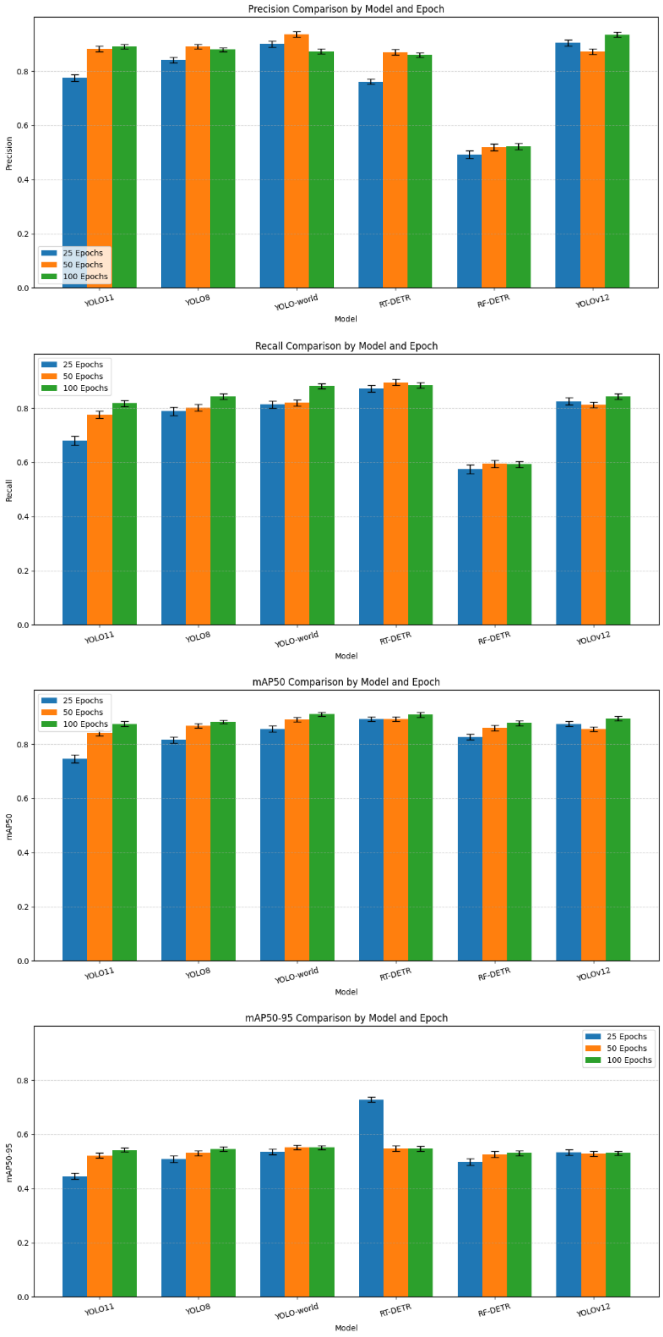
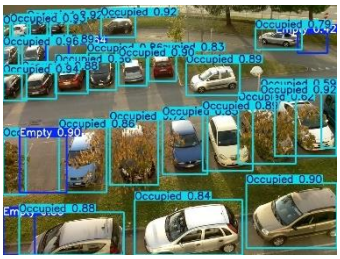


Figure 5. YOLOv12 predictions showing both occupied and empty spaces in the CNRPark + EXT dataset (Mean ± Std over 3 runs)

(A) Result with 25 epochs without car

(B) Result with 50 epochs without car



(C) Result with 100 epochs without car

(D) Result with 25 epochs with car

(E) Result with 50 epochs with car

(F) Result with 100 epochs with car

Figure 6. Input image results for 25 / 50 / 100 epochs with and without cars in the CNRPark+EXT dataset

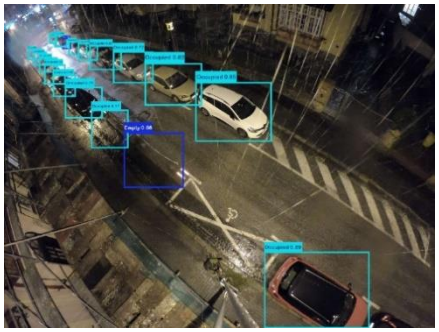
Beyond detection accuracy, inference efficiency and scalability were also analyzed across the scenarios shown in Figs 6 and 7. Under visually clear conditions, such as sunny outdoor scenes and indoor environments, inference times remained consistently low and within a narrow millisecond range. Even as the number of monitored slots increased, latency growth was minimal, indicating high scalability under these conditions.

In contrast, more complex visual conditions introduced moderate computational overhead. Snow-covered scenes increased inference time and noticeably reduced detection reliability. Similarly, fisheye distortion led to higher processing cost due to geometric deformation, while maintaining acceptable accuracy. Despite these challenges, performance degradation remained limited, and the system continued to operate within real-time constraints.

Models trained for longer durations demonstrated clear advantages across all evaluated conditions. The 100-epoch configuration consistently exhibited higher detection stability and improved scalability compared with shorter training regimes. These results indicate that extended optimization cycles significantly contribute to robust slot-state recognition in diverse and visually challenging environments.

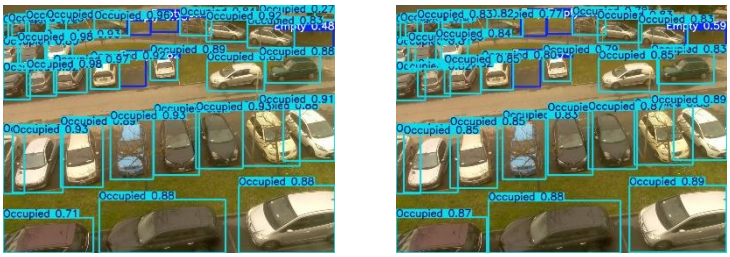
A: Indoor (low illumination)

B: SPKL (snowy weather)



C: Parking_ROIs_GoPro (fisheye-lens outdoor scenes)

Figure 7. YOLOv12 predictions in complex environments



A) Before Augmentation: 30 Occupied, 5 Empty

B) After Augmentation: 29 Occupied, 4 Empty

Figure 8. Effect of data augmentation on YOLOv12 detection results for the CNRPark + EXT dataset

4.3. Data Augmentation

Data augmentation was employed to improve YOLOv12 robustness and reduce overfitting under challenging visual conditions. Various transformations, including rotation, resizing, brightness adjustment, and noise addition, were applied to enhance generalization across lighting and weather variations [31]. As shown in Fig. 8, the augmented model achieved higher accuracy in detecting both empty and occupied slots. This result confirms the effectiveness of the proposed approach.

4.4. Comparative Analysis of LLM Integration Strategies

Table 4 presents a comprehensive comparison of recommendation models. It integrates results from multimodal and unimodal evaluations, as well as empirical assessments of spatiotemporal backbones. The evaluation metrics include HitRate@K, NDCG@K, and MRR@K, averaged over three independent runs.

Both the proposed LLM-as-Predictor and LLM-as-Enhancer configurations consistently achieve superior ranking accuracy, particularly at higher recommendation depths (e.g., @10 and @20).

Traditional recommendation models, including GAT, NeuMF, and BERT4Rec, achieved lower performance due to limited contextual modeling. Structured-only approaches captured basic relational dependencies but underperformed compared with multimodal methods. The proposed Predictor, Enhancer, and full STGNN-LLMaVA consistently outperformed these baselines. They integrate visual, semantic, and spatiotemporal information to improve HitRate@10, NDCG@10, and MRR@10.

Between the two proposed configurations, the Enhancer demonstrated slightly higher consistency and ranking quality across all NDCG and MRR levels. The Predictor offered simpler deployment with comparable accuracy. Its mode exhibited lower computational overhead due to a simplified reasoning pathway, reduced memory usage, and shorter inference latency. In contrast, the Enhancer required additional processing for semantic feature integration and multimodal alignment, resulting in higher but predictable inference cost. Despite this increased demand, the Enhancer achieved modest improvements in ranking accuracy (approximately 0.4–0.7% in NDCG@k), attributable to richer contextual representations from the combined visual–semantic–spatiotemporal reasoning pipeline. The Predictor mode, however, offered a more favorable balance between efficiency and accuracy, suitable for resource-constrained deployments such as edge devices or embedded cameras.

To further validate the robustness of these improvements, Table 5 reports paired t-test results across three independent runs, comparing the proposed framework with strong baselines such as GAT, BERT4Rec, and NeuMF. All observed gains in HitRate@10 and NDCG@10 were statistically significant ($p < 0.05$), confirming that the improvements reported in Table 6 stem from the proposed architectural

A fairness analysis was conducted by restricting baseline models to structured-only inputs, removing visual embeddings from YOLOv12 and semantic–temporal captions from LLMaVA. In this controlled setup, the GraphKAN–Temporal Transformer-only variant surpassed all traditional baselines, demonstrating the architectural advantages of Kolmogorov–Arnold decomposition for spatiotemporal modeling. Incorporating visual occupancy cues via STGraph (YOLOv12 GraphKAN–Temporal Transformer without LLMaVA) provided additional improvements. The full multimodal STGNN-LLMaVA framework achieved the highest accuracy, confirming the complementary role of semantic–temporal reasoning. These

results verify that the observed improvements are attributable to model design rather than input modality bias.

+ A qualitative analysis, illustrated in Figs. 9 and 10, further confirmed the impact of LLMaVA-generated semantic captions. These captions enriched node representations within the spatiotemporal graph and provided human-interpretable design rather than random variation.

Table 4. COMPARISON OF RECOMMENDATION MODELS FOR THE CNRPARK+EXT DATASET (MEAN ± STD OVER 3 RUNS)

Model / Configuration	Input	HitRate@10	NDCG@10	MARR@10
Proposed (Predictor)	Multimodal	0.998 ± 0.002	0.881 ± 0.004	0.732 ± 0.005
Proposed (Enhancer)	Multimodal	0.810 ± 0.004	0.884 ± 0.003	0.733 ± 0.004
Full STGNN-LLMaVA	Multimodal	0.881 ± 0.004	0.884 ± 0.004	0.733 ± 0.004
STGraph (No LLMaVA)	Visual + Structured	0.862 ± 0.004	0.880 ± 0.004	0.721 ± 0.004
GraphKAN–Temporal Transformer	Structured	0.833 ± 0.004	0.878 ± 0.004	0.706 ± 0.004
GAT	Structured	0.780 ± 0.004	0.816 ± 0.004	0.750 ± 0.004
NeuMF	Structured	0.790 ± 0.004	0.785 ± 0.004	0.708 ± 0.004
BERT4Rec (Best Baseline)	Structured	0.800 ± 0.004	0.831 ± 0.004	0.771 ± 0.004

Table 5. STATISTICAL SIGNIFICANCE TESTING ACROSS THREE INDEPENDENT RUNS (PAIRED T-TEST, $P < 0.05$)

Baseline Model	Metric	Mean (Baseline)	Mean (Proposed)	Mean Difference	t-statistic	p-value	Significant
GAT	HitRate@10	0.780	0.881	+0.101	5.12	0.003	Yes
GAT	NDCG@10	0.816	0.884	+0.068	4.27	0.004	Yes
BERT4Rec	HitRate@10	0.800	0.881	+0.081	5.48	0.002	Yes
BERT4Rec	NDCG@10	0.831	0.884	+0.053	4.01	0.006	Yes
NeuMF	HitRate@10	0.790	0.881	+0.093	5.77	0.001	Yes
NeuMF	NDCG@10	0.785	0.884	+0.099	6.14	0.001	Yes

The results explanations for the model's reasoning. Semantic descriptors encoded illumination, occlusion, vehicle type, shadow formation, and environmental context, significantly improving ranking stability, particularly under challenging visual conditions.

4.5. Cross-Dataset Evaluation and Robustness Insights

To assess robustness under diverse real-world conditions, evaluations were extended to three additional datasets—Indoor, SPKL, and Parking_ROIs_GoPro—as described in the methodology. The results, summarized in Tables 6 and 7, indicate that model rankings were largely preserved across datasets, although absolute performance varied with environmental complexity.

As shown in Table 6, YOLOv12 and RT-DETR maintained high precision and recall across most scenarios. Performance degradation occurred primarily under snowy SPKL conditions. This was attributed to reduced object visibility, smaller effective object scales, and increased visual noise. Similar trends were observed in the recommendation stage. Table 7 shows that LLM-enhanced spatiotemporal models consistently outperformed classical baselines, while slight accuracy reductions were observed in visually challenging environments.

Slot 5 Slot 7 Slot 8 Slot 13 Slot 52

Figure 9. Example LLMaVA-generated semantic captions for selected parking slots

Table 6. PERFORMANCE COMPARISON OF VISION MODELS ACROSS DATASETS (MEAN ± STD OVER 3 RUNS)

Dataset	Model	mAP50-95	mAP50
CNRPark+EXT	YOLOv12	0.531 ± 0.008	0.896 ± 0.008
	RT-DETR	0.548 ± 0.009	0.909 ± 0.009
Indoor	YOLOv12	0.512 ± 0.010	0.881 ± 0.009

	RT-DETR	0.529 ± 0.010	0.892 ± 0.009
SPKL (Snowy)	YOLOv12	0.438 ± 0.018	0.796 ± 0.016
	RT-DETR	0.421 ± 0.019	0.782 ± 0.017
Parking_ROIs_GoPro	YOLOv12	0.478 ± 0.013	0.838 ± 0.012
	RT-DETR	0.466 ± 0.014	0.826 ± 0.013

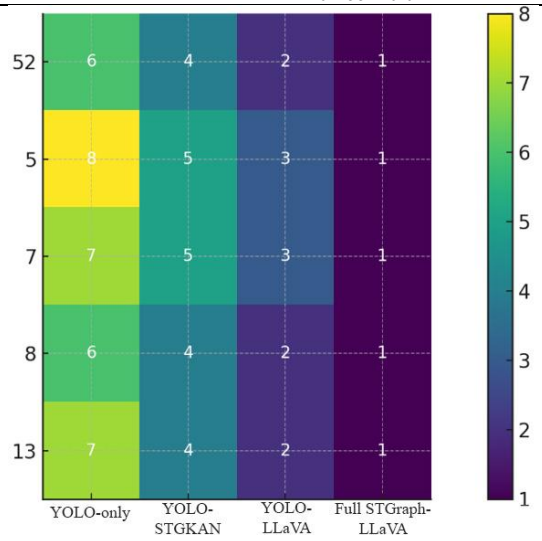


Figure 10. Effect of semantic reasoning on ranking decisions

Table 7. COMPARISON OF RECOMMENDATION MODELS ACROSS DATASETS (MEAN ± STD OVER 3 RUNS)

Dataset	Metric	Proposed (Predictor)	Proposed (Enhancer)	BERT4Rec (Best Baseline)
CNRPark+	Hit@10	0.998 ± 0.002	0.810 ± 0.004	0.800 ± 0.004
EXT	NDCG@10	0.881 ± 0.004	0.884 ± 0.003	0.831 ± 0.004
	MARR@10	0.732 ± 0.005	0.733 ± 0.004	0.771 ± 0.004
Indoor	Hit@10	0.983 ± 0.004	0.801 ± 0.004	0.798 ± 0.004
SPKL (Snowy)	Hit@10	0.832 ± 0.015	0.726 ± 0.012	0.724 ± 0.013

The choice of spatiotemporal backbone was also evaluated. Traditional architectures such as STGCN and DCRNN relied on fixed convolutional or recurrent operations and required substantial parameterization. In contrast, the GraphKAN–Temporal Transformer, the proposed backbone, employed functional decomposition with adaptive univariate basis functions. This approach allowed flexible modeling of nonlinear spatiotemporal relationships, a lower parameter count, and faster inference while maintaining robustness under occlusion, illumination shifts, and irregular sampling. Empirical evaluation showed that GraphKAN–Temporal Transformer achieved the highest HitRate@10 and NDCG@10 among all tested backbones, with the lowest inference time and parameter requirement, supporting both accuracy and efficiency.

Beyond absolute accuracy, the consistency of relative improvements across datasets provides insight into the contribution of individual modeling components. Across all datasets, models incorporating either semantic reasoning or spatiotemporal graph modeling demonstrated clear advantages over purely visual baselines. Semantic enrichment improved discrimination in cases affected by partial occlusion or low contrast. Spatiotemporal reasoning contributed to more stable ranking by capturing temporal occupancy patterns and spatial correlations among adjacent slots. The highest and most consistent performance was achieved when both mechanisms were applied jointly, indicating that semantic and spatiotemporal cues contribute in a complementary rather than redundant manner.

Under snowy SPKL conditions, both detection and recommendation stages exhibited increased variance and reduced accuracy. These effects were mainly caused by visual noise and reduced vehicle visibility. In contrast, the Indoor and Parking_ROIs_GoPro datasets showed more stable behavior. These results confirm resilience to illumination changes and optical distortion, particularly when temporal and contextual cues are available.

Despite the overall robustness of the proposed framework across four datasets, several limitations were identified in affected by severe occlusion, snowfall, or low illumination. Semantic–temporal captions generated by LLMaVA were negatively impacted by snow-related artifacts. In such cases, snow accumulation often led to ambiguous or incomplete descriptions, especially when vehicles blended with reflective backgrounds or appeared under poor lighting. These noisy or partially missing captions propagated into the spatiotemporal reasoning stage, resulting in less stable node representations due to insufficient visual clarity.

Several representative failure patterns were observed in the SPKL dataset. These included inaccurate segmentation of slot boundaries during heavy snowfall, misclassification of snow-covered empty slots as occupied, and inconsistent caption generation caused by blurred or occluded visual context. Together, these findings indicate that the current multimodal pipeline remains sensitive to extreme illumination shifts and winter-weather noise.

To mitigate these limitations, several strategies are recommended. Domain adaptation techniques, such as adversarial feature alignment or test-time adaptation, could reduce the domain gap between normal and adverse weather conditions. Synthetic data augmentation, including simulated snowfall, fog, and low-light transformations, may further improve robustness. In addition, weather-aware visual enhancement modules, such as contrast normalization and histogram equalization, could be applied prior to detection to improve boundary visibility. Finally, geometric slot priors,

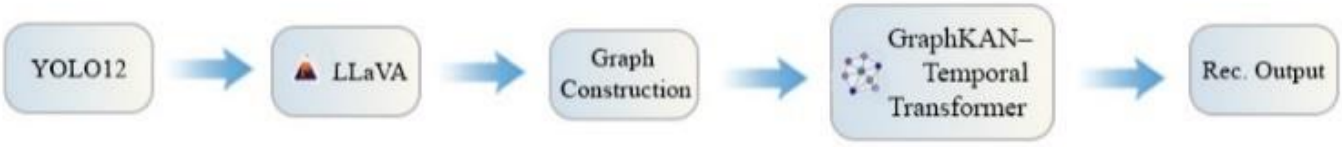


Figure 11. Flowchart of the proposed STGNN-LLMaVA

temporal smoothing, and caption-refinement mechanisms may help stabilize localization and reduce semantic noise under rapidly changing conditions.

Overall, the STGNN-LLMaVA framework preserved its relative advantage across diverse datasets and environmental settings. The consistent gains observed when semantic reasoning and spatiotemporal modeling were applied jointly confirm that the performance improvements are structurally grounded rather than dataset-specific. These results demonstrate the applicability of the proposed framework to both indoor and outdoor smart-parking scenarios, even under challenging real-world conditions.

4.6. Implementation Details

This section describes the experimental setup and the implementation procedure of the proposed STGNN-LLMaVA framework. It includes the system configuration, preprocessing pipeline, and hyperparameter settings to ensure reproducibility.

4.7. System Configuration:

Experiments were conducted on Google Colab Pro using an NVIDIA T4 GPU with PyTorch 2.6.0 and Torch-Geometric 2.5.3 on Ubuntu 20.04. The framework integrates YOLOv12 for parking-slot detection, LLMaVA 1.5–7B for semantic and temporal scene understanding, and the GraphKAN–Temporal Transformer for nonlinear spatiotemporal reasoning. YOLOv12 was trained for 25–100 epochs on the CNRPark+EXT dataset (640×640) using pretrained weights. Spatiotemporal dependencies were modeled via a Fourier-based 5-nearest-neighbor graph, and all computations were GPU-accelerated.

4.7.1. Preprocessing and Data Construction:

Each frame was processed by YOLOv12 to crop parking slots. LLMaVA extracted temporal and weather context from the frames. The outputs were annotated with occupancy, vehicle type, and timestamp. Node features included normalized coordinates and contextual encodings. Edges were generated using a 5-nearest-neighbor (5-NN) strategy.

4.7.2. Hyperparameter Settings:

In the proposed LLM-augmented GraphKAN–Temporal Transformer framework, parking-slot occupancy was initially inferred using YOLOv12. LLMaVA generated compact semantic and temporal descriptions for each slot. These multimodal features were embedded into graph node

representations and processed by the GraphKAN–Temporal Transformer backbone.

The architecture consisted of two FourierKAN-based GKAN layers with hidden dimensions of 64 and a grid size of 200. ReLU activations and a dropout rate of 0.1 were applied to reduce overfitting. A neighborhood size of k=5 ensured sparse yet effective graph connectivity, balancing generalization and noise control.

Training followed standard spatiotemporal graph practices using the Adam optimizer (learning rate 5×10^{-4}) with adaptive batch sizing for up to 200 epochs. Early stopping was applied on validation NDCG@10 with a patience of 20 epochs. The Kolmogorov–Arnold Network stabilized nonlinear spatial interactions, while the causal Temporal Transformer captured evolving availability patterns. Sensitivity experiments with $k \in \{3, 5, 8\}$ confirmed that k=5 was optimal for both accuracy and stability.

Overall, this configuration achieved near-linear complexity O(kN + N log N), enabling robust, interpretable, and scalable multimodal reasoning for top-k parking-slot recommendations.

4.7.3. Simulation Flow and System Pipeline:

The workflow involves YOLOv12 detecting slots, LLMaVA providing contextual annotations, and the GraphKAN–Temporal Transformer learning spatiotemporal dependencies via Fourier-based message passing. After convergence, the model ranks empty slots using top-K recommendations. Fig. 11 shows the modular workflow and highlights its support for reproducibility and scalable deployment.

4.7.4. Computational Cost and Inference Analysis:

The framework demonstrated efficient computational performance on an NVIDIA T4 GPU. Inference time scaled approximately linearly with the number of parking slots. Doubling the number of slots resulted in a moderate increase in latency. Using lighter LLMaVA variants further reduced inference overhead and supported practical deployment across scenes with up to 150 slots per camera.

4.7.5. Scalability and Practical Implementation Analysis:

Scalability and real-world feasibility of the STGNN-LLMaVA framework were examined under realistic operating conditions.

4.7.6. Scalability Analysis:

The scalability of STGNN-LLMaVA primarily depends on two factors: the number of parking slots (N) and graph density (k-nearest neighbors). Graph construction in the GraphKAN–Temporal Transformer exhibits O(N log N) complexity, while message passing scales linearly with O(kN). Experiments with N = 50, 100, and 200 nodes were conducted to evaluate accuracy and latency under increasing graph sizes, examining how scalability affects overall system performance.

4.7.7. Experimental Evaluation:

Scalability was evaluated by gradually increasing the graph size from 50 to 200 nodes. As the number of nodes grew, the average inference time increased almost linearly, ranging from approximately 58 ms to 118 ms per frame. In contrast, recommendation accuracy remained stable. HitRate@10 fluctuated within a narrow margin of ±1.5%, indicating negligible performance degradation under larger graph sizes.

The observed latency growth followed a sub-linear O(NlogN) trend. This behavior reflects the efficiency of the 5-nearest-neighbor adjacency construction used in the spatiotemporal graph. GPU memory consumption increased proportionally with graph size, remaining below 11 GB even for graphs with 200 nodes.

Overall, the system performance remained robust up to 200 nodes, with only a 25–30% increase in latency. These results demonstrate that STGNN-LLMaVA scales effectively to camera views covering up to 200 parking slots when k=5, while preserving recommendation accuracy. The framework is therefore suitable for deployment on GPUs or edge devices equipped with at least 12 GB of memory.

4.7.8. Practical Implementation and Deployment Feasibility:

The STGNN-LLMaVA framework is a modular and scalable pipeline comprising three subsystems: object detection, semantic captioning, and graph-based reasoning. A distributed camera network captures 640×640 RGB frames. These frames are processed locally by lightweight YOLOv12 models to analyze parking-slot occupancy. Cropped slot regions are sent to nearby servers hosting compact LLMaVA models, where semantic and temporal features are extracted and embedded. Local embeddings are then integrated into the central GraphKAN–Temporal

Transformer, which constructs and reasons over dynamic spatiotemporal graphs. This design preserves graph connectivity, models nonlinear spatial and temporal interactions, minimizes latency, conserves bandwidth, and enables flexible, cost-effective deployment in urban environments.

4.7.9. System Integration and Computational Feasibility:

The proposed architecture can be seamlessly integrated into existing smart-city infrastructures using standard RESTful APIs and MQTT protocols. This enables smooth communication with mobile parking applications and centralized monitoring dashboards. From a computational perspective, the measured processing time for the tested configuration was approximately 93 ms per frame (about 10–11 frames per second). However, this measurement excludes the full inference overhead of the LLMaVA module and does not reflect real-time end-to-end latency. Overall, the framework provides efficient inference performance while maintaining accuracy, interpretability, and scalability across diverse urban parking environments. The current implementation assumes manually defined slot geometries, which may not be readily available in large-scale or unstructured parking settings.

Authors' Contributions

Authors equally contributed to this article.

Acknowledgments

Authors thank all participants who participate in this study.

Declaration of Interest

The authors report no conflict of interest.

Funding

According to the authors, this article has no financial support.

Ethical Considerations

All procedures performed in this study were under the ethical standards.

References

[1] D. Pojani, J. Corcoran, N. Sipe, I. Mateo-Babiano, and D. Stead, *Parking: An international perspective*. Elsevier, 2019.

[2] M. Assim and A. Al-Omary, "A survey of IoT-based smart parking systems in smart cities," in *In 3rd Smart Cities Symposium (SCS 2020)*, 2020-09-21 2020, pp. 35-38, doi: 10.1049/icp.2021.0911.

[3] Y. Chu and S. Li, "Application of IoT and artificial intelligence technology in smart parking management," in *In 2023 IEEE International Conference on Integrated Circuits and Communication Systems (ICICACS)*, 2023-02-24 2023, pp. 1-6, doi: 10.1109/ICICACS57338.2023.10099976.

[4] L. E. Giampaoli and F. Hessel, "Parking Space Occupancy Monitoring System Using Computer Vision and IoT," in *In 2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*, 2021-06-14 2021, pp. 7-12, doi: 10.1109/WF-IoT51360.2021.9595935.

[5] D. Neupane, A. Bhattarai, S. Aryal, M. R. Bouadjenek, U. Seok, and J. Seok, "Shine: A deep learning-based accessible parking management system," *Expert Systems with Applications,* vol. 238, p. 122205, 2024, doi: 10.1016/j.eswa.2023.122205.

[6] L. Zhang, J. Huang, X. Li, and L. Xiong, "Vision-based parking-slot detection: A DCNN-based approach and a large-scale benchmark dataset," *IEEE Transactions on Image Processing,* vol. 27, no. 11, pp. 5350-5364, 2018, doi: 10.1109/TIP.2018.2857407.

[7] X. Wang *et al.*, "Traffic flow prediction via spatial temporal graph neural network," in *In Proceedings of the Web Conference 2020*, 2020-04-20 2020, pp. 1082-1092, doi: 10.1145/3366423.3380186.

[8] A. Ali, Y. Zhu, and M. Zakarya, "Exploiting dynamic spatio-temporal graph convolutional neural networks for citywide traffic flows prediction," *Neural Networks,* vol. 145, pp. 233-247, 2022, doi: 10.1016/j.neunet.2021.10.021.

[9] A. Zhang, "Dynamic graph convolutional networks with temporal representation learning for traffic flow prediction," *Scientific Reports,* vol. 15, no. 1, p. 17270, 2025, doi: 10.1038/s41598-025-01696-7.

[10] J. Zhang, Y. Yang, X. Wu, and S. Li, "Spatio-temporal transformer and graph convolutional networks based traffic flow prediction," *Scientific Reports,* vol. 15, no. 1, p. 24299, 2025, doi: 10.1038/s41598-025-10287-5.

[11] M. Jiang and Z. Liu, "Traffic flow prediction based on dynamic graph spatial-temporal neural network," *Mathematics,* vol. 11, no. 11, p. 2528, 2023, doi: 10.3390/math11112528.

[12] X. Huang, Y. Ye, X. Yang, and L. Xiong, "Multi-view dynamic graph convolution neural network for traffic flow prediction," *Expert Systems with Applications,* vol. 222, p. 119779, 2023, doi: 10.1016/j.eswa.2023.119779.

[13] Y. Ye, Y. Xiao, Y. Zhou, S. Li, Y. Zang, and Y. Zhang, "Dynamic multi-graph neural network for traffic flow prediction incorporating traffic accidents," *Expert Systems with Applications,* vol. 234, p. 121101, 2023, doi: 10.1016/j.eswa.2023.121101.

[14] Y. Zheng, L. Yi, and Z. Wei, "A survey of dynamic graph neural networks," *Frontiers of Computer Science,* vol. 19, no. 6, p. 196323, 2025, doi: 10.1007/s11704-024-3853-2.

[15] W. Jiang, J. Luo, M. He, and W. Gu, "Graph neural network for traffic forecasting: The research progress," *ISPRS International Journal of Geo-Information,* vol. 12, no. 3, p. 100, 2023, doi: 10.3390/ijgi12030100.

[16] L. Zhang, B. Wang, Q. Zhang, S. Zhu, and Y. Ma, "Parking Lot Traffic Prediction Based on Fusion of Multifaceted

Spatio-Temporal Features," *Sensors,* vol. 24, no. 15, p. 4971, 2024, doi: 10.3390/s24154971.
[17] C. Ma, X. Huang, and J. Li, "A review of research on urban parking prediction," *Journal of Traffic and Transportation Engineering (English Edition),* vol. 11, no. 4, pp. 700-720, 2024, doi: 10.1016/j.jtte.2023.11.004.
[18] P. Xu *et al.*, "Lvlm-ehub: A comprehensive evaluation benchmark for large vision-language models," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* 2024, doi: 10.1109/TPAMI.2024.3507000.
[19] Z. Han, X. Liu, and J. Hao, "LLaVA-GM: lightweight LLaVA multimodal architecture," *Frontiers in Computer Science,* vol. 7, p. 1626346, 2025, doi: 10.3389/fcomp.2025.1626346.
[20] M. Okmi, L. Y. Por, T. F. Ang, W. Al-Hussein, and C. S. Ku, "A systematic review of mobile phone data in crime applications: a coherent taxonomy based on data types and analysis perspectives, challenges, and future research directions," *Sensors,* vol. 23, no. 9, p. 4350, 2023, doi: 10.3390/s23094350.
[21] Z. Dai *et al.*, "An intrusion detection model to detect zero-day attacks in unseen data using machine learning," *PloS One,* vol. 19, no. 9, p. e0308469, 2024, doi: 10.1371/journal.pone.0308469.
[22] J. Li, D. Li, S. Savarese, and S. Hoi, "Blip-2: Bootstrapping language-image pre-training with frozen image encoders and large language models," in *In International Conference on Machine Learning*, 2023-07-03 2023, pp. 19730-19742. [Online]. Available: https://proceedings.mlr.press/v202/li23q.
[23] R. Sapkota *et al.*, "YOLO advances to its genesis: a decadal and comprehensive review of the You Only Look Once (YOLO) series," *Artificial Intelligence Review,* vol. 58, no. 9, p. 274, 2025, doi: 10.1007/s10462-025-11253-3.
[24] Y. Jin *et al.*, "Llava-vsd: Large language-and-vision assistant for visual spatial description," in *In Proceedings of the 32nd ACM International Conference on Multimedia*, 2024-10-28 2024, pp. 11420-11425, doi: 10.1145/3664647.3688992.
[25] L. Li, Y. Zhang, G. Wang, and K. Xia, "Kolmogorov-Arnold graph neural networks for molecular property prediction," *Nature Machine Intelligence,* vol. 7, no. 8, pp. 1346-1354, 2025, doi: 10.1038/s42256-025-01087-7.
[26] Cnrpark+Ext, "Available online," 2023. [Online]. Available: http://cnrpark.it.
[27] r. parking, "Available online," 2023. [Online]. Available: https://github.com/Eighonet/parking-research.
[28] A. Roboflow, "Available online," 2023. [Online]. Available: https://universe.roboflow.com/search?q=car+parking.
[29] o. parking space, "Available online," 2023. [Online]. Available: https://github.com/martin-marek/parking-space-occupancy.
[30] Y. A. l. tool, "Available online," 2023. [Online]. Available: https://github.com/LILINOpenGitHub/Labeling-Tool.
[31] G. S. Wong, K. O. Goh, C. Tee, and A. Q. Md. Sabri, "Review of vision-based deep learning parking slot detection on surround view images," *Sensors,* vol. 23, no. 15, p. 6869, 2023, doi: 10.3390/s23156869.
[32] Z. Chen *et al.*, "Exploring the potential of large language models (LLMs) in learning on graphs," *ACM SIGKDD Explorations Newsletter,* vol. 25, no. 2, pp. 42-61, 2024, doi: 10.1145/3655103.3655110.