Proceedings of Engineering and Technology Innovation, vol. 32, 2026, pp. 104-177

Real-Time Table Availability Detection in Dynamic Dining Environments Using YOLOv8 and Geometric Overlap Analysis

Muhammad Risaldi[1], Ayu Safitri[1], Andi Baso Kaswar[1,*], Muhammad Fajar B[1], Dyah Darma Andayani[1], Fhatiah Adiba[1], Firdaus[2], Jumadi M Parenreng[1]

[1]Department of Computer Engineering, State University of Makassar, Makassar, Indonesia

[2]Department of Electrical Engineering, State University of Makassar, Makassar, Indonesia

Received 17 September 2025; received in revised form 17 December 2025; accepted 22 December 2025

DOI: https://doi.org/10.46604/peti.2026.15704

Abstract

This study addresses the challenge of determining table availability in dynamic restaurant environments. Customer mobility, visual obstruction, and irregular table configurations make direct visual classification ineffective. A real-time table availability detection system utilizing YOLOv8 and simple online and real-time tracking (SORT) is proposed to address these difficulties. The primary innovation is a geometric overlap-based inference technique for table status assessment. The method examines the spatial link between customer centroids and table polygons. Centroid area expansion is applied to mitigate bounding box noise. The system is evaluated using an annotated dataset and compared with a direct detection baseline. Experimental findings indicate that the method attains an accuracy of 91.08%, markedly surpassing the baseline accuracy of 35.75%. Real-time performance assessment indicates a processing speed of 14.58 FPS with a latency of 68.57 ms during CPU-only execution. This performance satisfies real-time criteria. The study demonstrates that the method offers a dependable and efficient alternative for automated table availability monitoring.

Keywords: geometric overlap, object detection, real-time monitoring, table availability, YOLOv8

1. Introduction

In the digital era, the food service industry faces increasing challenges in maintaining operational efficiency. One critical aspect influencing both customer satisfaction and business performance is the management of table availability [1]. According to Statistics Indonesia (BPS, 2023), 86.35% of restaurants in Indonesia already hold an operational license, 86.37% possess business certification, and 87.46% are members of associations [2]. Nevertheless, manual practices in managing table availability are often inaccurate and unable to adapt to dynamic dining conditions [3]. This inefficiency can result in long waiting times, decreased customer satisfaction, and financial losses, particularly during peak hours [4].

In this context, customer expectations are rising, with a growing demand for fast and efficient services. Restaurants that fail to adapt to these expectations risk losing their competitiveness in an increasingly saturated market. One of the key challenges lies in managing real-time table availability, particularly when the restaurant is crowded. In many cases, empty tables remain uncleared and unavailable to the next customer due to the absence of effective monitoring by staff [5]. Some restaurants have adopted surveillance cameras (CCTV) to support monitoring. However, these systems remain passive and rely on manual observation, making them vulnerable to oversight and human error [6].

* Corresponding author. E-mail address: andi.baso.kaswar@unm.ac.id

Recent studies in computer vision have shown notable progress in space availability detection, especially in the context of parking lots. For example, Xu developed the VIDAR system using monocular cameras and inertial measurement unit (IMU) sensors, which demonstrated higher accuracy than YOLOv5 in open-area parking detection [7]. Similarly, An et al. proposed EPSDNet combined with EfficientDet-D3 and advanced sorting techniques, achieving 85.2% classification accuracy, 90.1% precision, and a processing speed of 0.5–0.8 seconds per frame [8]. Other works, such as Kurek and Macioszek, explored drivers' perceptions of parking availability with detection systems, yielding prediction reliability of 93.3% and 77.8% for positive and negative assessments, respectively [9]. Asy'ari et al. also utilized YOLOv5 with overlapping cameras, delivering a high mAP@0.5 of 0.953 [10].

Further developments include Mask R-CNN for smart city parking space detection, which achieved 85% accuracy and an F1-score of 86% but required GPU resources for optimal performance [11]. Zhang introduced an improved YOLOv5-based method that achieved 96.6% mAP@0.5, 93.7% accuracy, and real-time speed of 52.3 FPS [12]. Another study enhanced parking availability prediction using MAT-LSTM, achieving up to 48% higher accuracy than the classic LSTM model [13]. Similarly, De Luelmo et al. combined Monodepth2, pyramid scene parsing network (PSPNet), Faster R-CNN, and a rule-based system, obtaining an F1-score of 0.726 and an area under the receiver operating characteristic curve (AUC-ROC) value of 0.864 for automatic parallel parking detection [14].

Alternative methodologies have amalgamated IoT and data mining tools to forecast spatial availability. Ensemble learning outperformed traditional approaches with 93.23% accuracy, whereas the support vector machine (SVM) trailed at 71.72% [15]. Feng et al. introduced a hybrid dConvLSTM-DCN model that integrated spatial and temporal dynamics, enhancing both short- and long-term predictive performance [16]. Chen et al. assessed multiple machine learning methods and determined that Random Forest attains the highest accuracy of 93.96% in IoT-based parking management systems [17].

In addition to parking-related applications, research has progressed to visual place recognition (VPR) employing convolutional neural networks (CNNs) and metric learning methodologies, bolstered by benchmarking frameworks like VPR-Bench [18-19]. Further research has focused on space availability detection in green urban planning [20]. It has also addressed vegetation segmentation employing U-Net style xception models [21]. Additionally, Bakirci (2024) assessed the application of YOLOv8 for traffic surveillance in intelligent transportation systems (ITS). The study demonstrated that this model offered substantial enhancements in accuracy (mAP of 79.7%) and inference speed (8 ms) relative to YOLOv5. This position is particularly pertinent for computer vision-based real-time applications [22].

While computer vision has been utilized in various monitoring applications, previous studies have primarily focused on outdoor environments. These include parking facilities, urban analysis, and vegetation recognition. However, such approaches are not readily applicable to restaurant environments, as customer movement, unpredictable table configurations, and obstructions hinder detection efforts. To the best of current knowledge, only limited research has addressed table availability detection in dynamic restaurant settings. This highlights the need for approaches that can more accurately capture and interpret the spatial interactions between customers and tables.

This paper presents a real-time table availability detection system utilizing YOLOv8 and simple online and real-time tracking (SORT) to meet these requirements. YOLOv8 functions as the primary detection model because of its anchor-free design, superior feature representation, and swift inference abilities, making it ideal for recognizing customers and table areas in busy restaurant settings. This study introduces an inference method that employs geometric overlap between customer centroids and table polygons. The method also expands the centroid region to improve resilience against bounding box displacements and occlusion. A comparative investigation demonstrates that this strategy produces more accurate table status predictions than direct detection. As a result, the proposed system facilitates efficient and real-time monitoring of restaurant environments through improved table status prediction.

2. Research Method

This study develops an object classification model to achieve the research objectives through a systematic methodology. The approach includes data collection, preprocessing, annotation, division into training, validation, and testing sets, and YOLOv8 object recognition. Customer centroids and table intersection determine table availability, whereas SORT is used to monitor objects. To assess the approach efficacy, implementation, and evaluation are conducted. Fig. 1 shows the research technique, with the red area indicating the proposed method's main contributions to table availability identification.

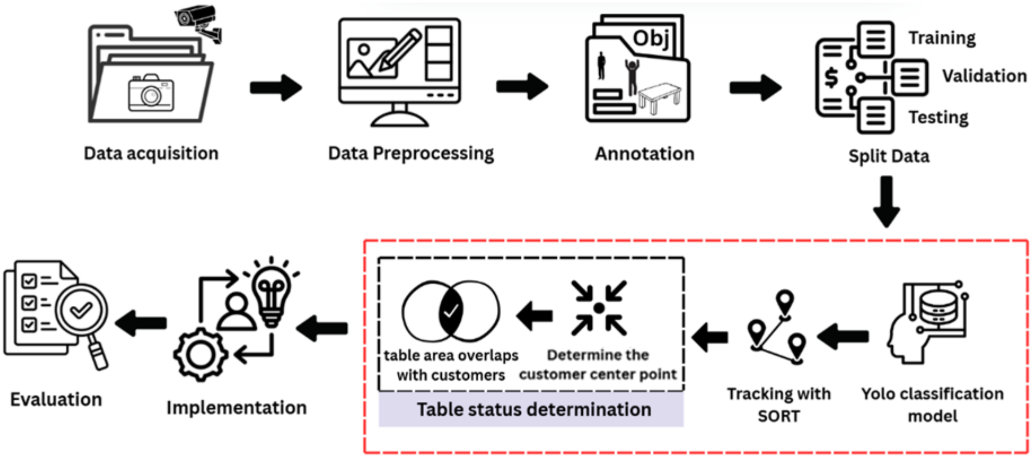

Fig. 1 Stages of the research method

2.1. Data acquisition

Data acquisition involves collecting images and videos to be used as datasets [23]. The data collection is conducted at a local dining establishment, Warung Makan Ayam Penyet Bang Coe, located on Monginsidi Baru Street, Makassar City. The utilization of a single location seeks to illustrate the viability of the suggested methodology under controlled real-world conditions. The site features a static table arrangement and a consistent CCTV camera viewpoint. A total of 200 CCTV videos are recorded in MP4 format, each with a duration of 35 minutes. The CCTV camera used is the CS-C6N-A0-1C2WFR model. It is equipped with a ¼ " complementary metal-oxide-semiconductor (CMOS) image sensor, a 4 mm f/2.4 lens (85° diagonal, 75° horizontal, 45° vertical), 1080p resolution, and a frame rate of 15 FPS. The stationary camera location ensures uniform room arrangement throughout data acquisition. This setup enables the assessment to concentrate on verifying the table availability detection technique, rather than extrapolating to other restaurant designs.

2.2. Data preprocessing

At this stage, the acquired CCTV footage is converted into image frames for model training. A total of 200 raw recordings are collected. However, many are excluded due to excessive blurriness, inadequate lighting, occlusion, or insufficient table visibility. In addition, the CCTV footage is inverted, requiring a 180° rotation during preprocessing. To ensure data quality, 30 optimal videos are selected based on consistent lighting conditions, clearly visible tables, and minimal interference. This dataset size is considered sufficient, as the study focuses on a proof of concept rather than the development of a large-scale, multi-site dataset. From each selected video, 70 frames are extracted, resulting in 2,100 images for YOLO training. All images are rotated by 180° and resized to 640 × 640 pixels. The remaining videos from the original dataset are reserved for model testing and subsequently re-filtered to meet the evaluation criteria.

2.3. Annotation

Object annotation is conducted on the Roboflow platform utilizing a polygon-based labeling method to accurately delineate the shape and boundaries of each object. Two trained annotators annotated a total of 2,100 images. Three object classes are established: customers, non-customers, and tables. Polygons representing tables are constructed to encompass the entire visible surface, whereas polygons for customers and non-customers conform to the body outlines within the frame. Individuals who are seated are classified as customers, whereas those who are standing are classified as non-customers. The annotation process yielded 12,505 customer instances, 1,353 non-customer instances, and 6,843 table instances, as detailed in Table 1.

Table 1 Total object annotation

Class	Number of Annotations
Customer	12.505
Non-Customer	1.353
Table	6.843

Polygon-based annotation facilitates precise representation of object geometry, which is essential for overlap analysis in assessing table status. To ensure labeling consistency throughout the dataset, both annotators adhere to identical annotation guidelines, encompassing criteria for labeling tables and human objects. A selection of annotated samples is examined and revised to rectify boundary inaccuracies or class ambiguities. The application of standard guidelines and a manual verification process contributes to the consistency of annotation quality. This is achieved despite the absence of formal metrics like inter-annotator agreement. Fig. 2 illustrates an example of an annotated object class.

(a) Customer (b) Non-Customer (c) Table

Fig. 2 Example of an object class for annotation

The annotated object shown in Fig. 2(a) represents the customer class, characterized by a seated person. Fig. 2(b) illustrates the non-customer class, represented by standing individuals. Fig. 2(c) depicts the table class. The total number of annotations for each class is summarized in Table 1.

2.4. Classification model

This classification model is developed using YOLOv8n segmentation to distinguish among customers, non-customers, and table items. A total of 2,100 photos are employed, allocated as follows: 80% for training (1,680 images), 10% for validation (210 images), and 10% for testing (210 images). The model is trained on a laptop running Windows 11 (64-bit). The laptop is equipped with an Intel® Core™ i7-13659HX 13th Generation CPU (2.60 GHz, 20 cores), 16 GB RAM, and an NVIDIA GeForce RTX 4050 GPU (6 GB dedicated, 8 GB shared). The hyperparameters applied during model training are summarized in Table 2.

Table 2 Hyperparameter used during model training

Hyperparameters	Value
Image size	640
Batch size	16
Epoch	100

2.5. Tracking with SORT

In the object tracking stage, this study employs the SORT algorithm, which is designed to track objects in real time within a video or image sequence. SORT identifies detected objects in each frame, assigns a unique ID to each object, and tracks their movements across consecutive frames [24]. In this research, SORT is specifically applied to track customers by assigning a unique ID to each detected individual. To ensure accurate and stable tracking throughout the video, the parameters are set as follows: max_age = 50, min_hits = 5, and iou_threshold = 0.2. These parameter settings contribute to effective and reliable tracking across the video sequence.

To further enhance consistency and accuracy between frames, two supporting algorithms are integrated: the kalman filter and the hungarian algorithm. The kalman filter estimates the position of an object in the next frame based on its previous position and velocity [25]. The hungarian algorithm matches objects detected in consecutive frames to maintain consistent IDs [26]. The combination of these algorithms enables efficient tracking of customer movements and ensures that table status is updated according to each customer's detected position.

2.6. Table Status Determination

In the table status determination stage, this study employed a geometric approach to identify the presence of customers around the table. This approach involves analyzing the geometric shapes of the detected objects. The Shapely Python library is utilized in this process to perform geometric operations. Shapely is not an algorithm or method but rather a library that provides functions for calculating and analyzing geometric shapes [27]. In this research, Shapely is used to measure the degree of overlap between the customer bounding box and the table polygon, while considering a detection margin to account for potential errors caused by inaccuracies in object positioning or recognition. An illustration of the representation of the table and customer areas is provided in Fig. 3.

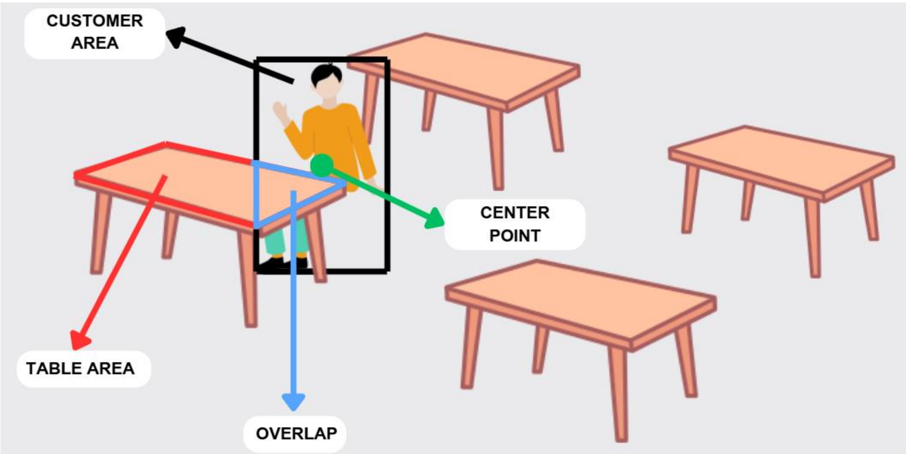

Fig. 3 Illustrative representation of the table and customer area

In Fig. 3, the customer is represented by a bounding box, while the table is defined using a polygon. Seats are unannotated as the system solely distinguishes between customers and non-customers based on seating position. Non-human objects, such as bags, cutlery, trays, or personal belongings, are deliberately disregarded to prevent misleading detections. Moreover, since each table is depicted as a polygon conforming to its physical dimensions. This design accommodates variances in table size, provided they adhere to the same arrangement. The approach depends on a static restaurant configuration and camera perspective. Thus, significant alterations in table or camera placement necessitate re-annotation of the table polygons.

During the table status assessment, visual interference concerns arise, especially when the customer bounding box intersects with neighboring tables, particularly in bustling restaurant settings. The customer box boundary occasionally concealed other tables, resulting in false positives. This study adopts a geometric method, using the customer's centroid as the primary reference to link the consumer to the nearest table. Occasionally, the customer's centroid does not accurately coincide with the polygon of the table at which they are seated due to a slight positional shift in their starting position. The central region is augmented by incorporating margins to its width and height. This margin accommodates slight deviations or annotation errors, enabling the enlarged center region to remain aligned with the accurate table. Let $({}_x, c_y)$ represent the consumer centroid derived from the bounding box. A rectangular margin area with a constant margin value of m = 40 pixels is established around the centroid. The margin region M is delineated by the following formula:

$$M = \left\{ (x, y) \in \mathbb{R}^2 \,\middle|\, c_x - m \le x \le c_x + m ,\ c_y - m \le y \le c_y + m \right\} \tag{1}$$

Each table is represented as a polygonal area T. The table occupancy status is then determined by the following formula:

$$Occupied(T) = \begin{cases} 1 , & if\ \ area\ (M \cap T) > 0, \\ 0 , & otherwise \end{cases} \tag{2}$$

This rule stipulates that a table is deemed occupied if the expanded customer centroid area overlaps with the table polygon. If there is no intersection, the table is deemed empty. Shapely is utilized to assess the intersection between the client center point margin and the table polygon. In the event of an overlap, the table is deemed occupied, and its status is revised. If no clients are identified in the vicinity, the table is deemed vacant. This centroid margin polygon method markedly decreases false positives caused by bounding box interference and enhances system reliability in dynamic and bustling restaurant settings.

To support this process, the system is designed with an architecture that consists of three main components: data input, detection process, and detection result, each of which plays a critical role in ensuring accurate detection outcomes. The data input component receives the video input and converts it into individual frames. The detection process then analyzes each frame, detects the presence of customers, and determines the table status (occupied or empty) based on geometric analysis. Finally, the detection result component presents the outcomes by displaying the table status with the corresponding visual overlay. The overall system architecture is illustrated in Fig. 4.

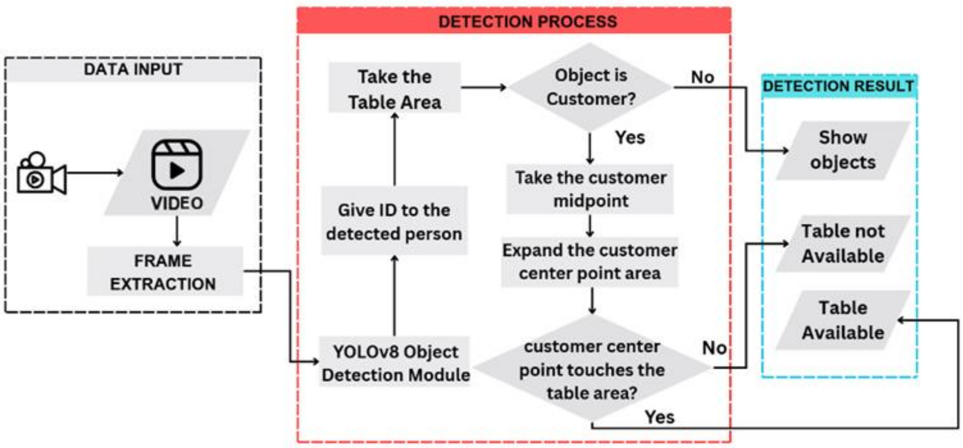

Fig. 4 General architecture of the table availability detection system

2.7. Implementation and Evaluation

At this stage, the developed system is tested to assess the effectiveness of table detection and customer tracking in real-world scenarios. The implementation involves applying the model to the acquired video data. To evaluate model performance, comparison data and model predictions from 100 video frames are used. The evaluation employs a confusion matrix to assess the model's ability to classify objects as customers, non-customers, and tables. From the confusion matrix, recall, precision, and F1-score values are calculated, as presented in Table 3.

Table 3 Confusion matrix

		Predicted	
		Positive	Negative
Actual	Positive	True Positive	True Negative
	Negative	False Negative	False Negative

Based on Table 3, four main components are used to evaluate model performance: true positive (TP), true negative (TN), false positive (FP), and false negative (FN). True positive (TP) represents the number of correct predictions for the positive class, while true negative (TN) indicates the number of correct predictions for the negative class. False positives (FP) refer to the number of incorrect predictions classified as positive, whereas false negatives (FN) represent the number of incorrect predictions classified as negative [23, 28].

These metrics are computed as follows:

$$\mathrm{Re}\,call = \frac{TP}{TP + FN} \tag{3}$$

To calculate the recall value, which represents the model's ability to identify all relevant objects, the formula shown in Eq. (3):

$$\mathrm{Pr}\,ecision = \frac{TP}{TP + FP} \tag{4}$$

Precision measures the accuracy of the model in correctly detecting relevant objects, as expressed in Eq. (4):

$$F1 - Score = 2 \times \frac{\text{Recall} \times \text{Precision}}{\text{Recall + Precision}} \tag{5}$$

Conversely, the F1-Score provides a balance between recall and precision, and is calculated using Eq. (5):

3. Results and Discussion

The table availability detection system is implemented to identify objects within the dining area, including tables, customers, and non-customer individuals. Figure 5 presents the annotation results that support the construction of the detection model.

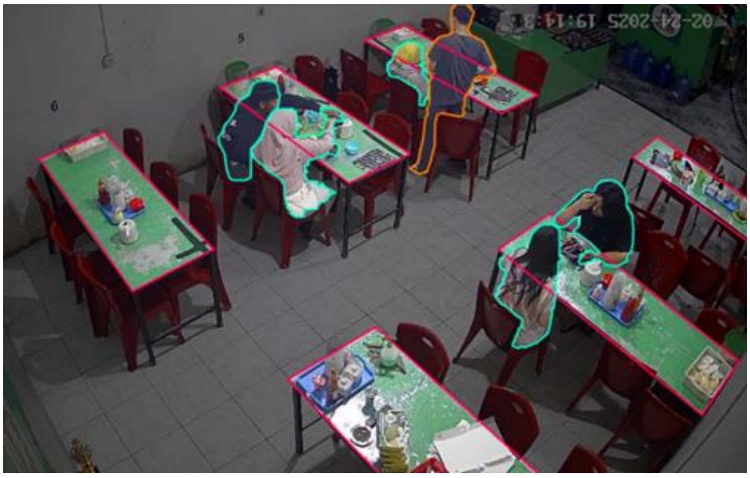

Fig. 5 Result of object annotation for building detection models

Fig. 5 presents the results of object annotation, in which the table class is marked with a red polygon, the customer class with a blue polygon, and the non-customer class with an orange polygon. After annotating the objects according to their respective classes, the best training results are achieved at the 89th epoch. The training performance indicated an mAP50(B) score of 0.96371, demonstrating highly accurate bounding box predictions for class B. Furthermore, the model achieved an mAP50-95(M) score of 0.79287, reflecting strong performance in detecting objects across stricter threshold variations ranging from 50 to 95. The confusion matrix of the selected model at the 89th epoch is shown in Fig. 6.

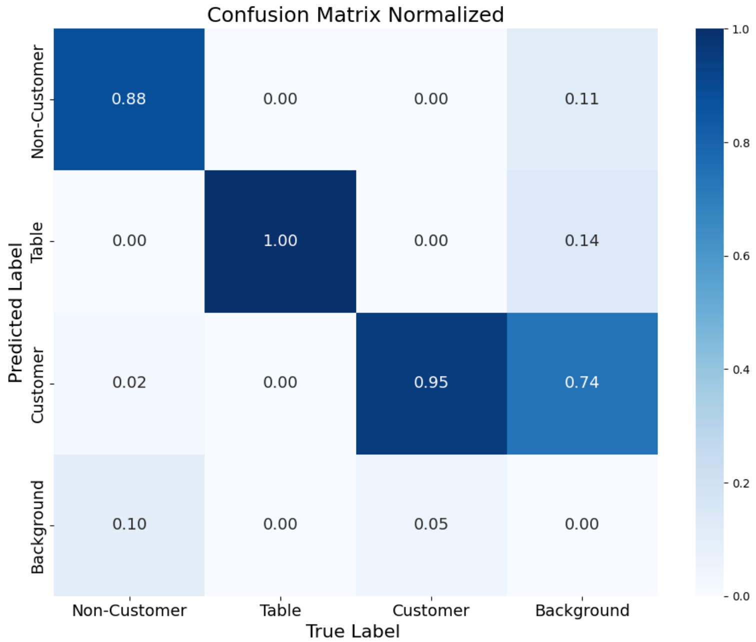

Fig. 6 Confusion matrix of the selected model at the 89th epoch

Based on Fig. 6, the table class achieves the highest TP rate of 100%, followed by the customer class with 95% and the non-customer class with 88%. However, a weakness is observed in predicting background elements that are incorrectly detected as objects. This issue lends to FP, particularly in the customer class, where the FP value is relatively high. This high FP value is caused by the significantly larger number of annotations in the customer class compared to other classes, resulting in a predictive bias toward the customer class. Consequently, when the model faces uncertainty, it tends to classify objects as belonging to the customer class. In addition, FN, which occur when an object that should have been detected is not correctly identified, are attributed to the relatively limited background. Overall, despite some errors in background prediction, the model remains effective in detecting most of the relevant objects, as illustrated by the curve results in Fig. 7.

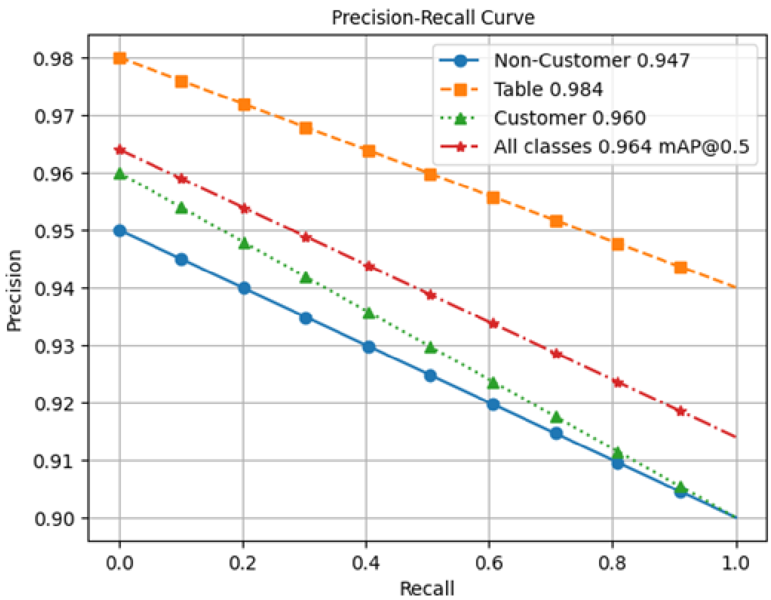

Fig. 7 Precision recall curve results

Based on the curve graph shown in Fig. 7, the mean average precision (mAP) at a threshold of 0.5 for all classes is 0.965, indicating that the model was able to identify and classify objects effectively. The table class demonstrates the highest performance, with a precision-recall value of 0.984. The customer class achieves a precision-recall value of 0.967. Meanwhile, the non-customer class records a precision-recall value of 0.944. Although slightly lower than the customer class, this result indicates that the model is still reliable in detecting non-customer objects. The results of object detection in videos using YOLOv8 are presented in Fig. 8.

Fig. 8 Object detection results using YOLOv8

As shown in Fig. 8, the detection results demonstrate that the model can effectively identify and classify objects, including tables, customers, and non-customers. Each detected object is marked with a bounding box corresponding to its class, providing a clear illustration of the model's capability in real-world scenarios. Prior to implementing the three-class annotation technique, a baseline experiment is conducted. In this experiment, YOLOv8 is trained to categorize table states into two categories: occupied and empty. In this configuration, a table area is designated as occupied if at least one customer is seated nearby, and as unoccupied if no customers are seated. This method seeks to ascertain table availability solely based on the visual characteristics of the table area, without independently identifying customer or table objects. This setup is designated as the direct table status detection model in Table 4. The assessment findings indicate considerable shortcomings of this methodology.

Table 4 Comparison between direct detection and proposed area-overlap models

Direct Table-Status Detection Model					Area Overlap Based Detection Model (Proposed Method)				
TP	Error Detection		Recall (%)	Precision (%)	TP	Error Detection		Recall (%)	Precision (%)
	FN	FP				FN	FP		
173	284	284	37.86	37.86	457	28	28	94.23	94.23
F1-Score (%)			37.86		F1-Score (%)			94.23	
Accuracy (%)			35.75		Accuracy (%)			91.08	

Fig. 9(a) illustrates that the model frequently misclassifies vacant tables as occupied, leading to a significant number of false positives. This inaccuracy arises because the model is required to infer intricate spatial relationships, such as determining if an individual is genuinely seated at a specific table based solely on the visual characteristics of the table area. In the absence of customer and table instance data, the model cannot reliably differentiate occupancy patterns, leading to diminished accuracy. This discovery necessitated a revision of the detection approach. Rather than directly forecasting table state, YOLOv8 is trained on three object classes: customers, non-customers, and tables. This enables explicit detection of each component in the scene, as illustrated in Fig. 9(b). The availability of tables is subsequently assessed by a geometric overlap method involving customer and table polygons. This novel concept underpins the proposed methodology and is juxtaposed with the direct table-status detection model presented in Table 4.

(a) False Detection (b) True Detection

Fig. 9 Detection results before and after class change

This modification demonstrates the enhanced capability of the model to recognize and differentiate object classes. Furthermore, the SORT model is optimized to assign unique IDs to detected person objects, which are then matched to the customer and non-customer classes. An example of object class ID assignment is presented in Fig. 10.

Fig. 10 Example of assigning ID to object classes

Fig. 10 shows an example of ID assignment for each detected object, namely, customers and non-customers. Each object is assigned a unique ID to facilitate tracking and movement analysis. For instance, the customer class is assigned ID 24, while two objects in the non-customer class were assigned IDs 21 and 20. To determine whether tables are occupied or empty, the system analyzes the proximity and overlap between the customer bounding box and table polygons. Using this approach, the system successfully identifies occupied tables, as illustrated in Fig. 11.

(a) False Detection (b) Correct Detection

Fig. 11 Table state detection results

Fig. 11 illustrates that the system effectively designates Table 5 as occupied due to the presence of multiple customers within the table's vicinity. A mistake occurs with Table 3, as shown in Fig. 11(a), where no customers are present, yet it is erroneously labeled as occupied. The misclassification occurs due to one of the client bounding boxes in Table 5 encroaching upon the area of Table 3, leading the system to erroneously designate Table 3 as occupied.

To address this issue, the center point of the customer's bounding box is used as a reference to calculate the overlap with the table area. Using this approach, each customer is associated only with the nearest table. The distance between points is calculated to achieve a more accurate overlap analysis between tables and customers. To further improve detection accuracy, a margin of 40 pixels is added around the customer's center point. With this adjustment, the customer's center point successfully overlaps only with the intended table, resulting in a corrected classification. As demonstrated in Fig. 11(b), Table 3 is accurately identified as vacant, despite one of the customer bounding boxes from Table 4 visually encroaching upon the area of Table 3.

Additionally, to evaluate the model's ability to determine whether a table is occupied or vacant, a classification evaluation is conducted for each table instance within individual video frames. Each frame contains multiple tables, and the occupancy status of each table is compared against ground truth annotations derived from manual labeling. A total of 628 table-status instances are collected from 100 test frames, forming the basis for quantitative performance analysis. This evaluation framework is designed to assess table occupancy determination under realistic and dynamic visual conditions.

The first configuration, referred to as the direct table-status detection model, serves as a baseline in which YOLOv8 classifies table regions into occupied and empty categories based solely on visual cues. This approach exhibits notable limitations, particularly when human presence is ambiguous or partially occluded. The second configuration represents the proposed method, in which YOLOv8 detects customers, non-customers, and tables as distinct object classes, and table occupancy is determined using a geometry-based overlap mechanism. A centroid-based margin region is generated for each detected customer and checked for intersection with the table polygon; overlapping tables are classified as occupied, while others are labeled as vacant. A comparison of both approaches is presented in Table 4.

Table 4 illustrates a notable disparity in accuracy. The proposed technique achieved an accuracy of 91.08%, a precision of 94.24%, a recall of 94.24%, and an F1-score of 94.23%, indicating substantial dependability in ascertaining table availability status. In contrast, the direct table-status detection model achieves an accuracy of only 35.75%, indicating that direct categorization based on the visual characteristics of the table area is insufficient for accurate table status assessment. The geometric overlap-based method markedly enhances system reliability by leveraging the spatial interaction between consumers and tables during the inference phase. A visualization of the comparison between the direct table-status detection model and the area overlap-based detection model (proposed method) is presented in Fig. 12.

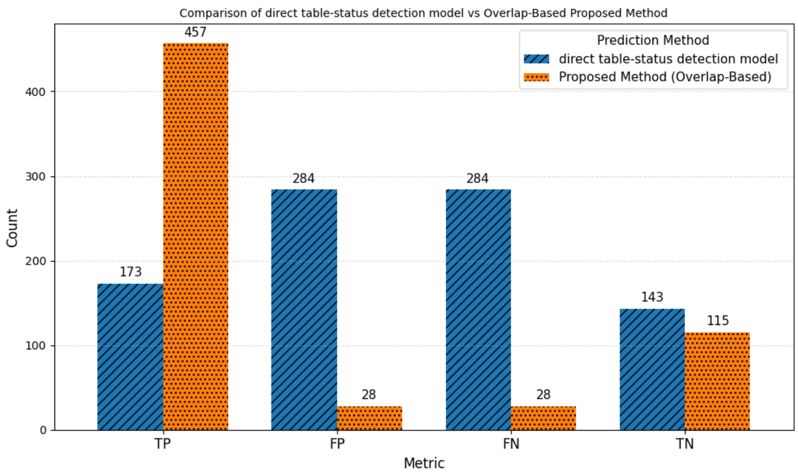

Fig. 12 Comparison between direct and proposed area overlap-based models

According to Fig. 12, the overlap-based detection model attains a True Positive (TP) count of 457, which is substantially higher than the 173 TPs obtains by the direct detection model. This illustrates that the overlap method is superior in precisely detecting vacant tables. Furthermore, the False Positive (FP) and False Negative (FN) values in the overlap method are much lower (28 each) compared to the direct table-status detection model, which produces 284 FP and 284 FN. This reduction in FP and FN indicates that the direct detection model frequently misclassified empty tables as occupied and vice versa.

In terms of true negatives (TN), the overlap-based detection model achieves a slightly lower value of 115 compared to 143 in the direct detection model. However, this difference remains within the acceptable range, considering the substantial increase in TP detection and the significant reduction in errors (FP and FN).

To illustrate real-time capabilities, testing is performed on a 19-second video comprising 580 frames on a CPU device. The YOLO model achieves an inference speed of 21.98 FPS, but the complete pipeline, encompassing centroid calculation, margin creation, polygon overlap assessment, SORT tracking, and visualization, operates at 14.58 FPS. The system's mean latency is 68.57 ms per frame, much beneath the 100 ms benchmark commonly regarded as the norm for real-time performance. These findings indicate that the proposed model enhances the system’s effectiveness in determining table availability in dining environments. The detection results of the model are presented in Fig. 13.

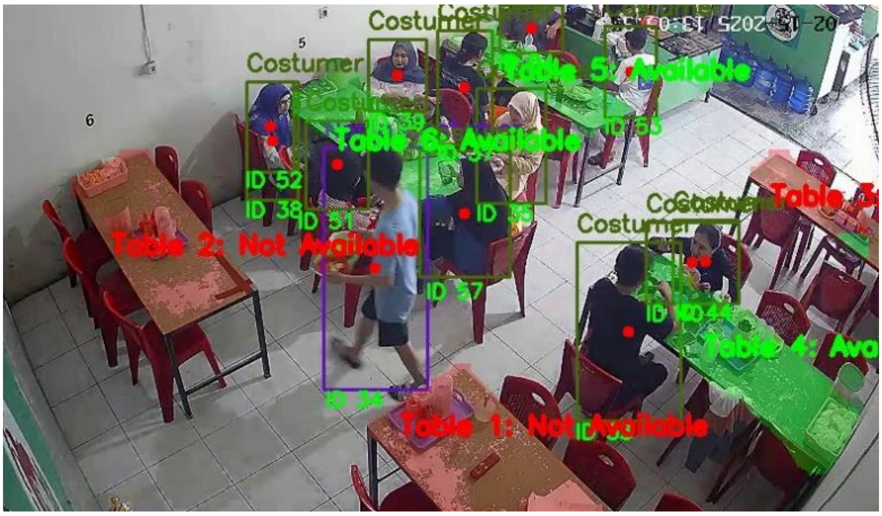

Fig. 13 Results of overlapping application detection between customers and tables

Fig. 13 illustrates three occupied tables (tables 4, 5, and 6), where several customers are detected within the seating areas. In contrast, Tables 1-3 are identified as empty. The proposed strategy demonstrates efficacy in a controlled context; however, its generalizability is constrained. The system is trained and evaluated using data from a single restaurant, including a static table arrangement and predetermined CCTV camera angles. Consequently, the system's precision is significantly reliant on a uniform room configuration. Should there be a substantial alteration in the table arrangement or camera positioning, the table polygon area must be re-annotated.

Moreover, the system employs solely two human-related categories: customer and non-customer, to delineate the table state. Non-human objects, such as bags, trays, or personal goods, are disregarded to prevent erroneous "occupied" detections. This methodology enhances the system's resilience in high-traffic restaurant settings but constrains its capacity to comprehend more intricate real-world situations.

The polygon-based table representation enables this strategy to accommodate different table sizes and shapes within the same environment; however, this approach has not been proven across several restaurants with differing geometries or seating configurations. Incorporating more object classes may enable the system to differentiate between human and non-human objects on tables, hence enhancing the robustness of table availability detection in dynamic real-world scenarios.

4. Conclusion

This study established a real-time table availability detection system integrating YOLOv8, SORT, and geometric overlap analysis to determine table status in a dynamic restaurant environment. Based on the experimental results, the following conclusions can be summarized:

(1) Geometric overlap-based inference improves table status detection: The method analyzed spatial relationships between customer centroids and table polygons, overcoming limitations of direct prediction in crowded restaurant environments.

(2) The proposed method achieved high accuracy: It attains 91.08% table state inference accuracy, which significantly outperformed the direct detection baseline of 35.75%.

(3) The system supported real-time performance on CPU: The system operated at 14.58 FPS with 68.57 ms average latency per frame, confirming feasibility for real-world restaurant deployment.

(4) Integration of YOLOv8 and SORT ensured robust tracking: The approach provided stable multi-object tracking and consistent table state inference, despite occlusions, customer movement, and bounding box disturbances.

Future research could focus on expanding the dataset with variations in restaurant layouts and camera angles. It could also aim to improve robustness to ambiguous objects, such as bags or cutlery placed on the table, and explore end-to-end architectures that integrate spatial reasoning directly into the table state inference process.

Acknowledgment

The authors would like to thank all parties who provided support during the completion of this research. This study did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of Interest

The authors declare no conflict of interest.

References

[1] R. A. Egigogo, M. T. Naniya, A. A. Abubakar, and A. Mansir, “Design and Implementation of Computerized Restaurant Table Booking System,” Ceddi Journal of Information System and Technology (JST), vol. 3, no. 1, pp. 47-55, 2024.

[2] S. Hasibuan, R. Basuki, and W. S. Jati, Statistik Penyedia Makan Minum 2022, Jakarta, Indonesia: BPS-Statistic Indonesia, 2023.

[3] M. Lv, J. Huang, H. Chen, and T. Zhang, “An Excimer Lamp to Provide Far-Ultraviolet C Irradiation for Dining-Table Disinfection,” Scientific Reports, vol. 13, no. 1, pp. 1-13, 2023.

[4] H. Mokayed, T. Z. Quan, L. Alkhaled, and V. Sivakumar, “Real-Time Human Detection and Counting System Using Deep Learning Computer Vision Techniques,” Artificial Intelligence and Applications, vol. 1, no. 4, pp. 221-229, 2022.

[5] F. M. Daradkeh, T. H. Hassan, T. Palei, M. Y. Helal, S. Mabrouk, M. I. Saleh, et al., “Enhancing Digital Presence for Maximizing Customer Value in Fast-Food Restaurants,” Sustainability, vol. 15, no. 7, article no. 5690, 2023.

[6] V. Singh and K. Karthik, “Optimizing Food Quality and Customer Service in Restaurants Through AI-Powered Monitoring Systems,” Technological Innovations in the Food Service Industry. Hershey, PA, USA: IGI Global, 2025, pp. 49-51.

[7] C. Huang, S. Yang, Y. Luo, Y. Wang, and Z. Liu, “Visual Detection and Image Processing of Parking Space Based on Deep Learning,” Sensors, vol. 22, no. 17, article no. 6672, 2022.

[8] Q. An, H. Wang, and X. Chen, “EPSDNet: Efficient Campus Parking Space Detection via Convolutional Neural Networks and Vehicle Image Recognition for Intelligent Human–Computer Interactions,” Sensors, vol. 22, no. 24, article no. 9835, 2022.

[9] A. Kurek and E. Macioszek, “Drivers’ Subjective Assessment of the Ease of Finding a Vacant Parking Space in an Area Equipped with Vehicle Detection Devices,” Sensors, vol. 22, no. 18, article no. 6734, 2022.

[10] M. F. Asy’ari, C. Fatichah, and N. Suciati, “Parking Space Availability Detections from Two Overlapping Cameras Using

YOLOv5 and Image Stitching Methods," International Journal of Intelligent Engineering System, vol. 16, no. 4, pp. 276-288, 2023.

[11] H. M. Al-Barhamtoshy, K. K. A. Abdullah, M. K. Dauda, and T. F. Himdi, "Detecting Available Parking Spaces in Smart Cities," Proceedings of the 1st International Conference in Advanced Innovation in Smart Cities (ICAISC), IEEE Press, pp. 1-5. 2003.

[12] X. Zhang, W. Zhao, and Y. Jiang, "Study on Parking Space Recognition Based on Improved Image Equalization and YOLOv5," Electronics, vol. 12, no. 15, article no. 3374, 2023.

[13] F. Zhang, K. Shang, L. Yan, H. Nan, and Z. Miao, "Prediction of Parking Space Availability Using Improved MAT-LSTM Network," ISPRS International Journal of Geo-Information, vol. 13, no. 5, article no. 151, 2024.

[14] S. P. De Luelmo, F. J. Garcia-Espinosa, A. S. Montemayor, and J. J. Pantrigo, "Combining Deep Learning Methods and Rule-Based Systems for Automatic Parking Space Detection," Integrated Computer Aided Engineering, vol. 32, no. 1, pp. 97-108, 2025.

[15] A. Dahiya, P. Mittal, Y. K. Sharma, U. K. Lilhore, S. Simaiya, E. Ghith, et al., "Machine Learning-Based Prediction of Parking Space Availability in IoT-Enabled Smart Parking Management Systems," Journal of Advanced Transportation, vol. 2024, no. 1, pp. 1-16, 2024.

[16] Y. Feng, Y. Xu, Q. Hu, S. Krishnamoorthy, and Z. Tang, "Predicting vacant parking space availability zone-wisely: a hybrid deep learning approach," Complex & Intelligent System, vol. 8, pp. 4145-4161, 2022.

[17] Z. Chen, X. Wang, W. Zhang, G. Yao, D. Li, and L. Zeng, "Autonomous Parking Space Detection for Electric Vehicles Based on Improved YOLOV5-OBB Algorithm," World Electric Vehicle Journal, vol. 14, no. 10, article no. 276, 2023.

[18] P. Wozniak and D. Ozog, "Cross-Domain Indoor Visual Place Recognition for Mobile Robot via Generalization Using Style Augmentation," Sensors, vol. 23, no. 13, article no. 6134, 2023.

[19] I. T. Papapetros, I. Kansizoglou, L. Bampis, and A. Gasteratos, "Visual Place Recognition in Changing Environments with Sequence Representations on the Distance-Space Domain," Machines, vol. 11, no. 5, pp. 1-11, 2023.

[20] I. Rakhmatullina, Z. Rakhmatullin, G. Zaitsev, A. Davydychev, G. Gilmanova, and M. Komissarov, "The Green Space Availability in Ufa City Metropolis," Forests, vol. 14, no. 7, article no. 1297, 2023.

[21] A. Duarte-Rangel, C. Camacho-Bello, E. Cornejo-Velazquez, and M. Clavel-Maqueda, "Semantic Segmentation in Large-Size Orthomosaics to Detect the Vegetation Area in Opuntia spp. Crop," Journal of Imaging, vol. 10, no. 8, article no. 187, 2024.

[22] M. Bakirci, "Utilizing YOLOv8 for Enhanced Traffic Monitoring in Intelligent Transportation Systems (ITS) Applications," Digital Signal Processing, vol. 152, article no. 104594, 2024

[23] H. Makmur, Wulandari, M. F. B., A. B. Kaswar, D. D. Andayani, F. Adiba, et al., "Motorcycle Parking Violation Detection System Using YOLOv7 with Region of Interest Mapping and Object Area Calculation," Advances in Technology Innovation, vol. 10, no. 1, pp. 29-43, 2025.

[24] M. M. Morsali, Z. Sharifi, F. Fallah, S. Hashembeiki, H. Mohammadzade, and S. B. Shouraki, "SFSORT: Scene Features-Based Simple Online Real-Time Tracker," unpublished.

[25] R. Yatigul, T. C. Jie, P. Gamolped, and E. Hayashi, "Kalman-YOLO Improving YOLO Tracking Performance through the Integration of a Kalman Filter for a Beach Cleaning Robot," Proceedings of the International Conference on Artificial Life and Robotics (ICAROB2025), pp. 326-329, 2025.

[26] C. A. Holz, C. Bader, M. Enzweiler, and M. Drüppel, "Data-Driven Object Tracking: Integrating Modular Neural Networks into a Kalman Framework," unpublished

[27] V. T. Nguyen, C. D. Do, T. D. Hoang, D. T. Nguyen, and N. T. Le, "Person Detection for Monitoring Individuals Accessing the Robot Working Zones Using YOLOv8," Proceedings of Computational Intelligence Methods for Green Technology and Sustainable Developement, Cham, Switzerland: Springer, pp. 50-59, 2024.

[28] I. P. Malashin, I. S. Masich, V. S. Tynchenko, A. P. Gantimurov, V. A. Nelyub, and A. S. Borodulin, "Anomaly Detection in Restaurant Receipts Data," IEEE Access, vol. 12, pp. 145590-145607, 2024.