Occupancy Measurement in Under-Actuated Zones: YOLO-based Deep Learning Approach

Ade Syahputra[1], Yaddarabullah[2], Mohammad Faiz Azhary[3], Aedah Binti Abd Rahman[4], Amna Saad[5]
Department of Informatics, Universitas Trilogi, Jakarta, Indonesia[1, 2, 3]
Schools of Science and Technology, Asia e University, Selangor, Malaysia[4]
Malaysian Institute of Information Technology, Universiti Kuala Lumpur, Kuala Lumpur, Malaysia[5]

***Abstract*—The challenge of accurately detecting and identifying individuals within under-actuated zones presents a relevant research problem in occupant detection. This study aims to address the challenge of occupant detection in under-actuated zones through the utilization of the You Only Look Once version 8 (YOLO v8) object detection model. The research methodology involves a comprehensive evaluation of YOLO v8's performance across three distinct zones, where its precision, accuracy, and recall capabilities in identifying occupants are rigorously assessed. The outcomes of this performance evaluation, expressed through quantitative metrics, provide compelling evidence of the efficacy of the YOLO v8 model in the context of occupant detection in under-actuated zones. Across these three diverse under-actuated zones, YOLO v8 consistently exhibits remarkable mean Average Precision (mAP) scores, achieving 99.2% in Zone 1, 78.3% in Zone 2, and 96.2% in Zone 3. These mAP scores serve as a testament to the model's precision, indicating its proficiency in accurately localizing and identifying occupants within each zone. Furthermore, YOLO v8 demonstrates impressive efficiency in executing occupant detection tasks. The model boasts rapid processing times, with all three zones being analyzed in a matter of milliseconds. Specifically, YOLO v8 achieves execution times of 0.004 seconds in both Zone 1 and Zone 3, while Zone 2, which entails slightly more computational effort, still maintains an efficient execution time of 0.024 seconds. This efficiency constitutes a pivotal advantage of YOLO v8, as it ensures expeditious and effective occupant detection in the context of under-actuated zones.**

***Keywords*—YOLO; HVAC system; occupant's position; occupant calculation; under-actuated zone**

I. INTRODUCTION

In recent years, increasing attention has been paid to improving the energy security of buildings. The focus has shifted to developing innovative concepts and technologies, increasing the energy efficiency of building envelopes and systems, and optimizing renewable energy sources (RES). Approximately 40% of all structures consume residential or commercial primary energy, and residential or commercial structures consume 40% more energy than others, especially in heating, ventilation, and air conditioning (HVAC) [1]. HVAC is an important system that must be considered, as it significantly. The HVAC system has two zones that must be controlled: under-actuated and fully actuated. The first type, "fully actuated," comprises a single room in which HVAC equipment may be controlled separately [2]. This zone is appropriate for areas with a fixed number of inhabitants and activities such as classrooms, offices, and auditoriums. Meanwhile, under-actuated zones in heating, ventilation, and air conditioning (HVAC) systems are areas where ventilation systems cannot effectively regulate air exchange rates. As a result, these areas can experience substandard air quality, adversely affecting the health [3].

Managing under-actuated zones in buildings presents a complex array of challenges, particularly in controlling the air distribution system. A critical factor in this regard is the direct impact of occupancy numbers on the cooling load [3]. Accurate detection of occupants is therefore essential, as variations in occupancy levels can lead to unbalanced cooling loads [4]. This imbalance often results in inadequate climate control, adversely affecting Indoor Air Quality (IAQ) and diminishing the overall energy efficiency of the system [5]. Further complicating the issue is the limited capacity of ventilation systems in these zones, often characterized by inadequate controls. This limitation can significantly hinder the distribution of fresh air throughout the occupied spaces, exacerbating IAQ issues and potentially impacting occupant health and comfort [6].

Addressing the challenges in under-actuated zones underscores the critical need for precisely adjusting airflow and regulating air temperature based on real-time occupancy data. These dynamic adjustments are essential for maintaining optimal environmental conditions and play a pivotal role in reducing unnecessary energy consumption, particularly in heating or cooling areas that are not occupied [7]. Furthermore, ensuring consistent and high-quality indoor air quality is vitally linked to the well-being and productivity of occupants [8]. In under-actuated zones, where occupancy levels vary and control over environmental conditions is limited, there is an increased risk of periods with compromised air quality [9]. The implementation of advanced occupant detection systems is key to enabling effective HVAC controls [10]. This integration facilitates the processing of real-time occupancy data, empowering the HVAC system to perform predictive adjustments and dynamically tailor its operations to align with the actual occupancy needs. Such an adaptive approach is not only crucial for maintaining comfortable environmental conditions but also has a significant impact on energy consumption [11]. By optimizing HVAC operations based on real-time occupancy data, buildings can realize substantial energy savings [12]. This is achieved by reducing the heating or cooling in less occupied areas, while ensuring that comfort is maintained in areas with higher occupancy [13][14].

Current study has developed occupant detection in such area by utilizing video or image processing. In the context of under-actuated zones, the implementation of video-based occupant detection systems faces unique and formidable challenges, primarily due to the unpredictable and complex nature of occupancy patterns in these areas [15]. Occupants in such zones display a diverse range of behaviors, from moving swiftly through the space to remaining stationary for prolonged durations [16][17][18]. This variability significantly challenges video-based detection systems, which must efficiently track fast-moving individuals and simultaneously accurately count and identify stationary occupants, ensuring comprehensive and precise occupancy detection [19][20]. Compounding this challenge is the intricate physical layout of under-actuated zones, often marked by obstructions and blind spots arising from furniture, partitions, and varied architectural elements [21]. These hindrances substantially reduce the efficiency of camera surveillance, leading to zones where occupants might remain unnoticed. Additionally, another challenge stems from inaccuracies in identifying occupants when utilizing standard video input frame rates. For instance, instances may occur where multiple occupants are present within a zone, yet the identification system detects only a single object. Further investigation into optimizing frame rates is warranted to enhance the accuracy of occupant detection. To address these challenges, camera systems require advanced features and optimized frame rates to accurately count and track occupants across varied scenarios, from low-activity environments to areas with high occupancy and dynamic movement patterns [22]. The unpredictability and diversity of occupant dynamics in under-actuated zones further necessitate the deployment of sophisticated algorithms for data processing and analysis [23]. These algorithms must be capable of interpreting complex and varied data to ensure effective tracking and counting of occupants. This requirement is particularly crucial in under-actuated zones, where environmental conditions may not be as controlled or predictable as those in fully-actuated zones, posing additional.

In this research, we aim to address prevailing gaps by developing a methodology that combines computer vision with deep learning techniques to detect and classify occupants, specifically focusing on quantifying the number of individuals in specific areas within under-actuated zones. Occupant calculation analysis can aid in optimizing indoor air volume distribution in areas with small occupancy HVAC systems. This approach can enhance indoor air quality, minimize energy consumption, and improve occupant comfort and productivity. It is crucial to employ an adept method for analyzing occupant calculation in under-actuated zones. The study centers on implementing the You Only Look Once (YOLO) method, specifically YOLO v8, for detecting occupants in the library rooms of Universitas Trilogi, areas typified as under-actuated zones. A fundamental aspect of this investigation involves analyzing a dataset comprising video input from these under-actuated zones. To facilitate a comprehensive analysis, the dataset was categorized into three types: original, compressed, and slowed down versions. For each frame of video input within these datasets, Roboflow was utilized to annotate the occupants and specific areas of under-actuated zones, thereby creating labeled data essential for training the model. The YOLO v8 model was then employed for each dataset variant, with a focus on investigating the detection confidence threshold to enhance the precision of occupant detection and quantification. A crucial aspect of this study was the comparative analysis of the model's performance, including metrics such as mean average precision, accuracy, and processing time. This performance was benchmarked against state-of-the-art methods like YOLO v5 and Faster R-CNN, providing a comprehensive understanding of YOLO v8's efficacy in occupant detection within under-actuated zone.

II. Related Work

Recent studies have increasingly focused on examining the presence and behavior of occupants in specific zones of HVAC systems, highlighting a keen interest in the correlation between occupancy and system efficiency. Notably, the use of cameras, in tandem with computer vision-based technologies for occupancy detection and recognition, has emerged as a significant area of interest among researchers. This approach is particularly effective as cameras can accurately identify occupants, even those engaged in minimal movement or sedentary activities, a capability crucial for comprehensive monitoring in various scenarios. However, its application in studying and optimizing HVAC systems represents a novel and promising direction in enhancing building energy efficiency.

Tien et al. [8] developed a region-based Faster Convolutional Neural Network (Faster R-CNN) that was capable of detecting and recognizing occupancy patterns and equipment used in an office area. The model was trained and deployed on a regular camera, and field tests were conducted in an office setting. The proposed method was evaluated in the field by recognizing various individuals performing diverse actions in an office environment, such as walking, sitting, and standing. A detection model was created by training a CNN using a transfer learning-based approach to classify occupancy activities. The model was then applied to a camera to enable real-time detection. The model's performance was assessed using a 15-minute experimental detection test, and across all activities, the average detection accuracy was found to be 98.65%.

Wei et al. [24] investigated the potential of using a live occupancy detection approach to help adjust building HVAC system operations to ensure adequate interior thermal conditions and air quality while reducing excessive building energy loads to improve the overall building energy performance. Faster R-CNN models were trained to detect the number of individuals (Model 1) and occupancy activities (Model 2) and deployed to an AI-powered camera to enable live occupancy detection. Model 1 attained an average detection accuracy of around 98.9%, which was higher than Model 2's accuracy of about 88.5%, owing to Model 1's lower complexity. Building energy simulation (BES) model was used to perform scenario-based modeling of the case study building under four ventilation scenarios during the heating and cooling seasons. The results showed that the proposed approach might offer a DCV to improve IAQ and address the under-or overestimation of ventilation demand when utilizing static or fixed profiles. It provides insights into how the proposed approach can adjust HVACs based on occupant dynamic

changes and the potential of this strategy to improve indoor air quality and energy efficiency.

Papakis et al. [25] developed a method that can recognize and classify passengers in a vehicle based on cabin photos. The Second Strategic Highway Research Program (SHRP 2) naturalistic dataset containing blurred cabin photos was used to design and test the system. They proposed a CNN-based approach to detect and locate passengers to recognize and identify individuals and classify them as drivers, front-seat passengers, or rear-seat passengers. After assessing various object detection models, to optimize performance, they used the Faster R-CNN architecture with a ResNet-101 backbone, pre-trained on ImageNet, fine-tuned for person detection using SHRP 2 cabin data, and produced the best results. The two distinct test sets found occupant detection accuracies of 94.5% and 98.1%, respectively.

Taheri [11] developed detection-based techniques using the Kanade–Lucas–Tomasi (KLT) tracker to extract many features from video footage. After proposing a conditioning technique for feature trajectories, they introduced a trajectory-set clustering method for identifying the number of moving objects in a scene. Considering these encouraging results, they propose extending our method to identify a more complex model of the appearance and motion of objects. They also plan to investigate the combination of our approach with static object counting methods. Further improvements will include autocalibration (at least to correct the perspective) and background discrimination from objects to ensure the method works for handheld cameras. The result of the proposed method was conducted on three kinds of datasets (USC, Library, and Cells), where the average error of USC was 0.8, LIBRARY was 2.7, and CELLS was 24. This indicates that the proposed method performs well for the USC dataset.

Chatista [15] proposed a novel algorithm for dense-crowd estimation. The proposed method divides an image into small rectangular patches. Each patch underwent a crowd/non-crowd SURF feature binary SVM classifier. These labels and CNN-based head detections were used to estimate the head size in each patch. The count for patches without head detection was estimated using the weighted average of the neighboring pixel counts. This approach was evaluated using three challenging datasets. The results show that our approach yielded low error rates for high- and medium-density crowd images. Because they used a pre-trained head detector trained on totally different data, they aimed to train our head detector on similar high-density crowd images. This would naturally lead to better detection and, thus, better crowd count estimates. Similarly, a perspective-aware head detector would also boost detection accuracy. In addition, better semantic segmentation of the scene for crowd detection is also under consideration. The overlaying of the rectangular grid on the entire image does not consider the image perspective information; the patch size can be modified as the distance from the camera increases to achieve better results. For better results, the SURF classifier can also be trained on less-dense crowd images, especially compared with no weight, with weight having best performance. As shown from the MSE value, SURF classifier with weight has score 61.4 and with no weight has score 79.8.

Previous studies have predominantly utilized the Faster R-CNN method for occupant counting in the realm of computer vision. This method enhances the original R-CNN framework by accelerating performance through shared computation and employing neural networks for region proposal, rather than relying on a selective search [20]. While Faster R-CNN marks an improvement over R-CNN in terms of speed and accuracy, it still falls short in achieving real-time performance, a significant limitation for practical applications [21]. One of the primary reasons for this shortfall is the extensive number of candidate suggestions it generates, approximately 2000, which makes processing time-intensive. For instance, analyzing an image with the bounding box regressor in Faster R-CNN can take around 50 seconds. Moreover, Faster R-CNN is a resource-intensive approach, necessitating substantial storage for feature maps across all regions [23]. This requirement leads to a considerable storage demand, often in the hundreds of gigabytes, due to the need to cache extracted features from the pre-trained CNN on disk for subsequent SVM training [22]. Additionally, being a multi-stage model with distinct components, Faster R-CNN cannot be trained end-to-end, which adds to its complexity and restricts adaptability. Its reliance on selective search algorithms has been critiqued for rigidity and lack of flexibility in diverse scenarios. Most existing research has been focused on enhancing the detection of occupant quantity and distribution in fully-actuated zones. However, there has been a notable gap in developing effective solutions for under-actuated zones, which pose unique challenges due to their variable occupancy and environmental conditions. The need for advanced methods that can effectively address occupant detection in under-actuated zones remains a significant area for further research and development.

III. Materials and Methods

This case study will be conducted at the Universitas Trilogi Library and is shown in Fig. 1. This library consist of five rooms of under-actuated zones includes from number 1 to 5. Each area of under-actuated zones has several distinct areas. The sampling observation and data collection was done in a corner room (room number 4) with three area based ventilation, each of which is 25 m2.

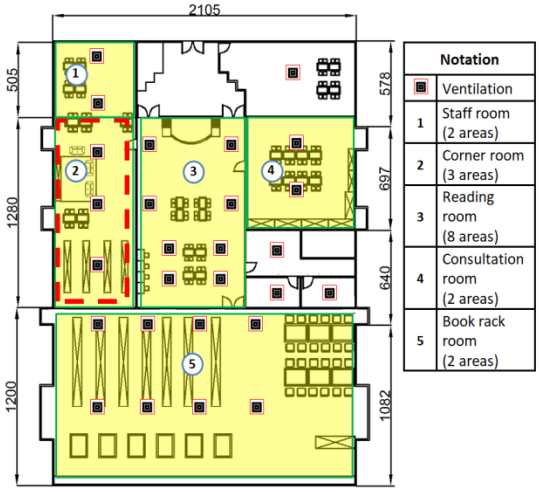

Fig. 1. Layout of Universitas Trilogi Library.

The objective of this research is to determine the arrival patterns of occupants in an under-actuated zone and examine the efficacy of a proposed vision-based, real-time occupancy detection and calculation method. Utilizing YOLO v8, this study aims to accurately identify and count the number of occupants in real-time within such zones. The research methodology is structured into three distinct stages. The first stage, data collection, involves the collection of relevant data variables and their subsequent adjustment to suit the study's needs. The second stage, occupant detection, focuses on identifying and detecting occupants within the under-actuated zone using the YOLO v8 model. The third stage, involves the calculation of the number of occupants present in the zone. The final stage of this research involves a comparative performance analysis between the YOLO v8 model, used for real-time occupant detection in under-actuated zones, and other prevalent models such as YOLO v5 and Fast R-CNN. This comparative analysis aims to evaluate the efficacy, accuracy, and efficiency of YOLO v8 in identifying and counting occupants, in contrast to the performance of YOLO v5 and Fast R-CNN under similar conditions.

A. Data Collection

In our study, we made use of an exclusive dataset that was specifically designed to examine occupancy in under-actuated zones. This dataset was carefully developed through extensive observations that were carried out in three specific areas within the student corner room of Universitas Trilogi library. These areas were identified as under-actuated zones. The data collection process was carried out using three surveillance cameras that were placed in each area in the student corner of the library. The cameras were capable of capturing footage of varying lengths, ranging from 3 to 5 minutes, which resulted in a diverse range of visual data. The footage captured by these cameras provided a detailed and comprehensive view of the occupants and the surrounding environment, such as the tables, chairs, and books. This comprehensive visual data is essential for developing precise 2D object models. The data from each camera offers a unique perspective on the environment, allowing for a multifaceted analysis of occupant behavior and their interaction with the space. The diversity in camera angles and the range of activities captured in the footage ensure a robust dataset.

B. Occupant Detection

This phase involves occupant detection. We utilized YOLO v8 by ultralytics for better throughput with the same number of parameters owing to ultralytics changes, demonstrating hardware-efficient design reforms. All YOLO models were created and used to detect objects. Object detection models were trained to recognize the items in the images. When item classes are discovered, they are surrounded by bounding boxes and are categorized. YOLO is a new algorithm that predicts items and their locations in an image with a single glance. It detects objects in real time using neural networks. This method has evolved over time, beginning with YOLO v1 (or unified), which includes various localization issues and progresses to YOLO v2, YOLO v3, YOLO v4, YOLO v5, YOLO v6, YOLO v7, and YOLO v8(Terven & Cordova-Esparza, 2023).

YOLO divides an image into grids by using a single Convolutional Neural Network (CNN) model. Each grid estimates the bounding boxes and confidence scores. The class of the object in the bounding box is calculated using the predicted confidence score [26]. YOLO v8 variations produce a higher throughput with the same number of parameters, indicating hardware-efficient design reforms. The fact that ultralytics provided YOLO v8 and YOLO v5, with YOLO v5 providing impressive real-time performance, and based on the initial benchmarking results released by ultralytics, it is strongly assumed that YOLO-v8 will focus on constrained edge device deployment at a high inference speed [27].

YOLO v8 is a model that does not rely on anchors. This means that it forecasts the center of an object directly rather than the offset from a known anchor box [27]. Anchor boxes are a very difficult aspect of early YOLO models because they can represent the box distribution of the target benchmark, but not the distribution of the custom dataset. Anchor-free detection minimizes the number of box predictions, which speeds up Non-Maximum Suppression (NMS), a complex post-processing phase that shifts through candidate detection following inference [27]. The first 6 × 6 conv in the stem was replaced with a 3 × 3 conv, the primary building block was modified, and C2f was replaced with C3. The module is depicted below, where "f" represents the number of features, "e" is the expansion rate, and CBS is a block composed of Conv, BatchNorm, and SiLU. C2f concatenates all outputs from the bottleneck (a fancy name for two 3 × 3 convs with residual connections). In C3, only the output of the previous bottleneck was utilized. The bottleneck is the same as that in YOLO v5, but the kernel size of the first convolution increases from 1 × 1 to 3 × 3. Based on this data, we can conclude that YOLO v8 is beginning to regress to the ResNet block described in 2015 [20]. The features were concatenated directly into the neck without forcing the same channel dimensions.

In this study, the YOLO v8 model architecture is utilized for detecting and calculating the number of occupants, a process meticulously illustrated in Fig. 2.

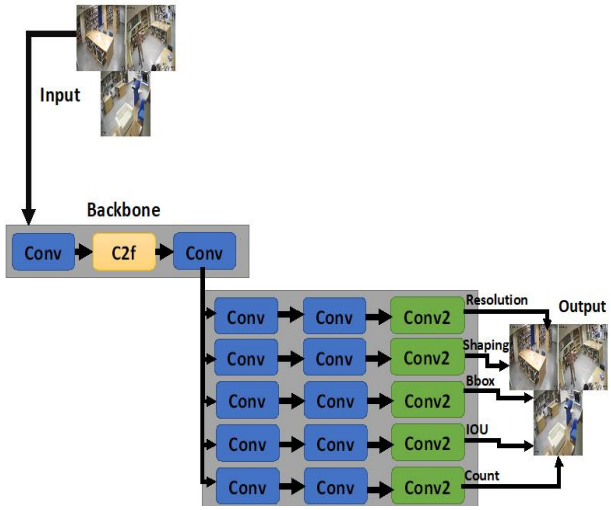

Fig. 2. YOLO v8 model architecture for occupant detection and calculation.

In the sophisticated realm of video-based object detection, the intricately designed model under discussion is specifically engineered to meticulously analyze video input datasets. This analytical journey begins with the critical task of processing raw video footage, a foundational step that determines the efficacy of all subsequent analyses. The core of this model is its head component, which is integral to the complex process of occupant identification within the video stream. The model's head plays a pivotal role in discerning and isolating occupants as distinct entities within the video frames. This task involves a series of intricate steps, beginning with the precise adjustment of the video frames' resolution. This adjustment is not a mere enhancement of visual quality but a strategic decision crucial for balancing clarity with computational efficiency. The model employs advanced algorithms to assess each frame, determining the optimal resolution that ensures clear visibility of occupants while simultaneously minimizing the processing load. This optimization is paramount, as it directly impacts the model's ability to accurately detect and analyze occupants without overburdening the system's computational resources. Furthermore, the model incorporates sophisticated techniques to handle variations in lighting, movement, and background complexity within the video frames. These techniques include dynamic contrast enhancement for low-light conditions, motion stabilization for dynamic scenes, and background subtraction algorithms to isolate occupants from complex backgrounds. Each of these techniques contributes to the model's overall efficiency, ensuring that the occupants are detected accurately regardless of the varying environmental conditions within the video footage. In addition to resolution adjustment and environmental adaptation, the model's head also integrates advanced object recognition algorithms. These algorithms leverage deep learning techniques to discern occupant characteristics, differentiating them from other objects in the frame. The model is trained on extensive datasets, enabling it to recognize a wide range of occupant attributes and behaviors, further enhancing its detection accuracy. The processing of raw video footage, therefore, is a multifaceted and complex endeavor within this model.

The intricate process of object detection in video analysis using the YOLO v8 model consists of several carefully orchestrated stages. The first stage is the preprocessing phase, an essential component of the process. This stage is focused on normalizing video quality and resolution, which lays the foundation for optimal detection performance. During this stage, each video frame is thoroughly analyzed and adjusted to ensure that its quality and resolution are suitable for the detection process. It is crucial to maintain a delicate balance between preserving essential details necessary for accurate identification and optimizing the frames to reduce computational load. The preprocessing phase employs techniques such as dynamic resolution scaling and adaptive bitrate control to maintain the integrity of crucial visual information while ensuring that the frames are not excessively data-heavy. Once the preprocessing phase is complete, the YOLO v8 model moves on to the object detection stage. The model's head, a central component in the architecture, plays a crucial role in this stage. The model's head is designed to efficiently distinguish and identify occupants within the video frames as unique entities. This involves deploying advanced neural networks that have been trained on extensive datasets to recognize human figures and differentiate them from other objects in the frame. Bounding boxes are a critical component in this phase. For each detected occupant, the model meticulously generates a bounding box, carefully encapsulating the occupant. This encapsulation is crucial as it isolates the occupant from the surrounding environment and other non-relevant elements within the frame, ensuring that each detection is distinctly recognized. The positioning and sizing of these bounding boxes are calculated with precision, taking into account the contours and dimensions of each occupant. Once the bounding boxes are established, the YOLO v8 model embarks on a probabilistic assessment to ascertain the likelihood that the objects within these boxes are indeed occupants. This assessment involves calculating confidence levels for each detection, a process that draws upon the model's learning from numerous annotated examples. These confidence levels serve as a measure of the model's certainty in its detections. To enhance the accuracy and reliability of the detection process, the model applies a threshold for these confidence levels. Detections that fall below this threshold are deemed less likely to be accurate and are consequently filtered out. This thresholding is a crucial step in ensuring that the occupant count is not only precise but also reliable, as it effectively eliminates false positives and other erroneous detections. In the final stage of the process, the YOLO v8 model performs the occupant counting task. This involves a comprehensive analysis of the detected occupants, considering factors such as the varying sizes, positions, and even the potential occlusions of the occupants within the frames.

Intersection over Union (IOU) is a metric that is widely regarded for its intuitiveness and effectiveness in the field of object detection, particularly in tasks involving bounding box predictions [28]. The computation of IOU involves a straightforward yet insightful mathematical formula. Essentially, it is calculated by taking the area of overlap between the predicted bounding box and the ground truth bounding box (the actual object's location), and then dividing this overlap area by the union area of these two boxes. The union area is the combined area covered by both the predicted bounding box and the ground truth bounding box, minus the overlap area, following in Eq. (1).

$$IoU = \frac{Area\ of\ Overlap}{Area\ of\ Union} \quad (1)$$

The simplicity of the IOU calculation allows for easy visualization and understanding. One can easily picture the overlapping areas of the two boxes to comprehend how well the predicted bounding box aligns with the actual object's position and size. This visualization aspect makes IOU a particularly accessible metric for evaluating the accuracy of object detection models.

C. Occupant Calculation

In this phase a zone is created by setting the coordinates in the frame. The OpenCV library was used to visualize the zone. A zone was created by setting the coordinates in the zone. Before we can start counting objects in a zone, we must first define the zone in which we want to count objects [29]. The coordinates of the zones are required. We use these later to

determine whether an object is inside or outside the zone. To calculate the coordinates inside a zone, we can use Polygon Zone, an interactive web application that allows to draw polygons on an image and export their coordinates for use with supervision. Once we have added points, a NumPy array will be made available on the page. This array contained the coordinates of the points in the zone [30]. The next step was to identify persons in each frame of the movie using a pretrained YOLO v8 object detection model. The number of objects in the zone is calculated by counting the number of objects with unique IDs. Subsequently, a limit was imposed on this zone. We begin by importing the necessary dependencies and then describe the zone in which to count the items using coordinates [27]. Subsequently, we initialized the objects to be used to process and annotate the video. The zone object tracks the zones in our image, and annotators are used to describe how the predictions in our movie should be annotated [27]. We filter out all classes by specifying that we only want detections with class ID 0. This ID maps to the "person" class. This object recognition and tracking system in a specified zone is useful for counting occupants in a zone in the HVAC system area and for creating several zones to track occupants in an under-actuated zone region.

D. Performance Evaluation

Several indicators were utilized to measure the accuracy and efficacy of the suggested method for counting people in a certain region. Typical YOLO performance metrics processing time mean average precision (mAP), and accuracy [31]. mAP is a typical evaluation metric that delivers a single figure as the mean of the Average Precision (AP) values for all classes. This allows for the evaluation of the performance of a model using a single number. As a result, mAP is the most commonly used evaluation metric for object detection algorithms. This is calculated as follows:

$$mAP = \frac{\sum_{q=1}^{Q} AveP(q)}{Q} \quad (2)$$

where, Q is the total number of queries in the set and q is the average precision query. Because our study only has a "person" class, the number of classes will be one. mAP indicates that the confidence threshold (IOU). The Accuracy indicates how close the estimation values of the proposed method are to the true values, and is excellent if it is high. The Accuracy score is calculated by dividing the number of correct predictions by the total prediction number [32]. The accuracy rate formula was calculated as follows. (TP: True Positive, TN: True Negative, FP: False Positive, FN: False Negative):

$$Accuracy = \frac{Tp+Tn}{Tp+TN+Fp+Fn} \quad (3)$$

IV. Result and Discussion

A. Result

The dataset was meticulously compiled through observations in the student corner room at Universitas Trilogi, which is divided into three zones. It originated from three video inputs, each three to five minutes long, capturing detail of occupants within the library's student corner. For each of dataset was expanded into three distinct subsets: the original, compressed, and slowdown. The development environment was established using Google Collabs, a platform for Python programming, integrating several libraries including numpy, ultralytic, and supervisory. Notably, YOLO v8, provided by ultralytic, was selected for its enhanced throughput capabilities, maintaining efficiency with the same parameter count due to ultralytic improvements. The model's third segment was dedicated to occupant's detection and counting.

Tables I, II, and III provide a detailed overview of the datasets used for the training and testing of object detection models across three distinct zones. Each table is specifically allocated to one zone and further categorizes the datasets into three types: Original, Compressed, and Slowdown. These categories are indicative of the different forms of video data utilized in model development. The objective of dividing the three datasets is to evaluate whether the comparison of the number of frames in the dataset has an impact on the result.

The Original Dataset, as represented in these tables, adheres to a standard format. It features a default time duration and maintains a frame rate of 30 frames per second (fps). This dataset serves as a baseline, offering a conventional setting for evaluating model performance. In contrast, the Compressed Dataset focuses on data efficiency. For each zone, the video data is modified by reducing the total frame count to a uniform 500 frames. This approach is designed to test the models in scenarios where full-frame rates are unavailable or computationally burdensome, assessing the models' performance under data-limited conditions. The Slowdown Dataset, on the other hand, is intended to evaluate the models' capabilities in handling more extensive frame sequences. This is achieved by augmenting the total frame count by 30% relative to the original dataset for each zone. Such an increase in frames is aimed at simulating situations where detailed temporal information is crucial for accurate object detection.

TABLE I. Dataset of Zone 1

Dataset	Frame	Training	Testing
Original	7045	6340	705
Compressed	285	256	29
Slowdown	8526	7532	994

TABLE II. Dataset of Zone 2

Dataset	Frame	Training	Testing
Original	2821	2445	376
Compressed	119	106	13
Slowdown	3647	3283	364

TABLE III. Dataset of Zone 3

Dataset	Frame	Training	Testing
Original	2925	2630	291
Compressed	195	175	20
Slowdown	4248	3819	429

For Zone 1, as shown in Table I, the Original dataset comprises 7045 frames, predominantly used for training (6340 frames) with a smaller subset for testing (705 frames). This extensive dataset provides a solid foundation for robust model training. The Compressed dataset, with a total of 285 frames (256 for training and 29 for testing), presents a more condensed form of data, posing potential challenges due to the loss of detail. The slowdown dataset is the most extensive, with 8526 frames, where 7532 are used for training and 994 for testing, offering a vast range of data to assess the model under various temporal conditions. In Zone 2, as per Table II, the dataset sizes are smaller compared to Zone 1. The Original dataset contains 2821 frames, split between 2445 for training and 376 for testing. The Compressed dataset, consisting of 119 frames (106 for training and 13 for testing), is significantly smaller, while the slowdown dataset, the largest in this zone with 3647 frames, is divided into 3283 for training and 364 for testing. This dataset size variation is crucial for evaluating the model's adaptability to different data scales and resolutions. Zone 3, detailed in Table III, mirrors Zone 2 in terms of dataset sizes. The Original dataset has 2925 frames, with 2630 dedicated to training and 291 to testing. The Compressed dataset, comprising 195 frames (175 for training and 20 for testing), and offers a compact data set for model evaluation. The largest dataset in this zone is the slowdown category, with 4248 frames, 3819 for training and 429 for testing, which is instrumental in assessing the model's performance over extended periods. The assessment of the occupant detection model encompassed three distinct areas and four datasets. Tables IV through VI provide essential metrics, including mean Average Precision (mAP), accuracy, recall, and processing time, which are crucial for evaluating the performance of the occupant detection model across four datasets and three zones, utilizing the YOLO v8.

TABLE IV. OCCUPANT DETECTION OF ZONE 1

Dataset	mAP	Accuracy	Recall	Time
Original	99.2	98.4	98.6	0.004
Compressed	92.6	90.6	97	0.004
Slowdown	96.8	96.4	95.5	0.004

In the realm of scientific research, particularly in the evaluation of occupant detection systems within Zone 1 as presented in Table IV, a meticulous comparative analysis between the Original, Compressed, and Slowdown datasets unveils notable differences in their respective performance metrics. This detailed examination is pivotal for assessing the system's accuracy and efficiency under varying data conditions, providing insights into the adaptability and robustness of the detection models. The Original dataset emerges as the benchmark for performance, demonstrating exceptional precision and reliability in occupant detection. It boasts a Mean Average Precision (mAP) of 99.2%, signifying near-perfect accuracy in distinguishing true positives from false positives. Additionally, an accuracy rate of 98.4% and a recall rate of 98.6% underscore the model's effectiveness in correctly identifying true positives and negatives, with minimal instances of false negatives. The rapid execution time of 0.004 seconds further accentuates the model's swift processing capability, a critical factor for real-time applications. In comparison, the compressed dataset, designed to assess performance under data-limited conditions, shows slightly diminished but still robust metrics. It achieves a mAP of 92.6%, indicating strong precision in a compressed frame environment. The accuracy rate stands at 90.6%, and the recall rate at 97%, both of which are commendable given the dataset's reduced frame count. Notably, the model maintains the same execution speed as the Original dataset, evidencing its efficiency in handling fewer data frames without compromising processing speed. The Slowdown dataset, characterized by an increased frame count, displays a competent performance, albeit with slight variations from the Original dataset. It records a mAP of 96.8% and an accuracy of 96.4%, indicating effective detection capabilities, though with a minor decrease in detecting all actual positives, as reflected by a recall rate of 95.5%. Remarkably, the execution time remains consistent at 0.004 seconds, demonstrating that the model's processing efficiency is not adversely affected by the augmented frame count.

TABLE V. OCCUPANT DETECTION OF ZONE 2

Dataset	mAP	Accuracy	Recall	Time
Original	78.3	66.1	84.9	0.024
Compressed	82.9	84.1	80.9	0.004
Slowdown	84.4	80.9	68.2	0.013

In the results section examining occupant detection in Zone 2, as depicted in Table V, an exhaustive analysis of the Original, Compressed, and Slowdown datasets reveals a diverse range of performances in terms of mean Average Precision (mAP), Accuracy, Recall, and execution Time. The Original dataset exhibits a moderate level of detection capability, with an mAP of 78.3%, an Accuracy of 66.1%, and a notably higher Recall of 84.9%. However, its execution time is considerably longer at 0.024 seconds, suggesting a trade-off between accuracy and processing speed. In contrast, the compressed dataset demonstrates enhanced performance with a mAP of 82.9%, a significantly higher Accuracy of 84.1%, and a Recall of 80.9%. Notably, this dataset achieves these metrics while maintaining a much faster execution time of 0.004 seconds, indicating enhanced efficiency in processing compressed data without compromising detection effectiveness. The slowdown dataset presents an interesting profile, registering the highest mAP of 84.4% and an Accuracy of 80.9%, but a lower Recall of 68.2% compared to the other datasets. Its execution time stands at 0.013 seconds, positioning it between the original and compressed datasets in terms of processing speed. Collectively, these results from Zone 2 indicate varying levels of effectiveness in occupant detection across different datasets. While the compressed dataset stands out for its balanced high performance and efficiency, the original dataset, despite its slower processing time, excels in Recall. The slowdown dataset, on the other hand, offers the best mAP but at the cost of a lower Recall rate. This variance in performance across datasets highlights the importance of dataset selection and optimization in occupant detection systems, as each dataset presents its unique strengths and limitations in accurately and efficiently detecting occupants in Zone 2.

TABLE VI. OCCUPANT DETECTION OF ZONE 3

Dataset	mAP	Accuracy	Recall	Time
Original	96.2	90.1	93.7	0.004
Compressed	93.5	91.1	86	0.004
Slowdown	97.4	94.4	97.1	0.004

Table VI displays the results of occupant detection in Zone 3 using three distinct datasets: the original, compressed, and slowdown. The original dataset in Zone 3 sets a high benchmark in terms of performance. It achieves a Mean Average Precision (mAP) of 96.2%, reflecting its high precision in correctly identifying true positive detections. The accuracy rate of 90.1% further illustrates the model's capability in effectively distinguishing between true positives and negatives. Additionally, a recall rate of 93.7% indicates the model's proficiency in identifying the majority of actual positive cases, thus minimizing false negatives. Notably, these metrics are attained with a rapid execution time of 0.004 seconds, underscoring the model's efficiency in processing. In contrast, the compressed dataset, while exhibiting a slightly lower mAP of 93.5%, demonstrates a high accuracy of 91.1%. This suggests that, despite the reduction in data volume, the model retains its effectiveness in accurate detection. However, the recall rate experiences a decline, dropping to 86%. This reduction points to a slight compromise in the model's ability to identify all true positives following data compression. Despite this, the model maintains the same brisk execution time of 0.004 seconds, indicating that the reduction in recall does not significantly impact the overall processing speed of the system. The slowdown dataset, interestingly, outperforms both the original and compressed datasets in Zone 3. It registers the highest mAP of 97.4%, suggesting superior precision in detection. Alongside, it achieves the highest accuracy of 94.4% and the best recall rate of 97.1%, surpassing the other datasets in effectively identifying true positives and minimizing false negatives. Remarkably, these superior metrics are achieved within the same efficient execution timeframe of 0.004 seconds, indicating that the increased frame count in the slowdown dataset enhances performance without compromising on processing speed.

Three zones were measured in pixels using Roboflow polygon zone web tools, which can convert meters to pixels. These tools are adept at converting measurements from meters to pixels, thereby accurately representing the areas of interest in square meters. This precise conversion is essential for the effective application of object detection techniques, where spatial accuracy is paramount. One of the key metrics in object detection is Intersection over Union (IOU), which is critical for evaluating the accuracy of detection models. IOU quantifies the level of overlap between the predicted bounding boxes and the ground truth, essentially measuring the accuracy of the model's predictions. In this context, the IOU threshold is often set at varying levels - 25%, 40%, and 50%. The selection of these thresholds is strategic, as they represent different degrees of alignment between the model's predictions and the actual observed data. Accurate detection is generally considered when at least half of the predicted bounding box aligns with the ground truth, signifying a 50% IOU threshold. This standard is commonly adopted in various object detection tasks, including occupant detection, ensuring that the model's predictions correspond appropriately to real-world instances. The effectiveness of these thresholds and the overall accuracy of the object detection models are comprehensively evaluated across three different zones. Each zone presents a unique scenario with varying occupant numbers: Zone 1 contains 1 occupant, Zone 2 has 13 occupants, and Zone 3 accommodates 10 occupants. Tables VII to IX provide an in-depth comparison of the accuracy of calculating the number of occupants based on the range of IOU thresholds, juxtaposed against actual observations from the three datasets.

In Zone 1, as presented in Table VII, the performance metrics, including mean Average Precision (mAP), Accuracy, Recall, and processing Time, are examined across three datasets: Original, Compressed, and Slowdown. The Original dataset demonstrates exceptional performance, boasting a high mAP of 99.2%, Accuracy of 98.4%, and Recall of 98.6%, all achieved within an impressively rapid execution time of 0.004 seconds. This signifies the model's ability to accurately detect occupants in Zone 1 with both precision and efficiency. The Compressed dataset, while still maintaining good performance, exhibits a slight reduction in mAP (92.6%) and Accuracy (90.6%), although the Recall remains high at 97%. Importantly, the execution time remains consistent at 0.004 seconds, suggesting that data compression does not significantly impact processing speed. The Slowdown dataset stands out in Zone 1, achieving the highest mAP of 96.8%, Accuracy of 96.4%, and Recall of 95.5%, all accomplished within the same efficient execution time of 0.004 seconds. These results underscore the varying efficacies of the occupant detection system across different datasets within Zone 1.

TABLE VII. OCCUPANT CALCULATION OF ZONE 1

Dataset	IOU	Number of Occupants	Accuracy with actual (%)
Original	0,25	1	100
	0,4	1	100
	0,5	1	100
Compressed	0,25	1	100
	0,4	1	100
	0,5	1	100
Slowdown	0,25	1	100
	0,4	1	100
	0,5	1	100

TABLE VIII. OCCUPANT CALCULATION OF ZONE 2

Dataset	IOU	Number of Occupants	Accuracy with actual (%)
Original	0,25	9	69
	0,4	7	53
	0,5	6	61
Compressed	0,25	10	76
	0,4	7	53
	0,5	6	46
Slowdown	0,25	10	76
	0,4	7	53
	0,5	6	46

Zone 2, as detailed in Table VIII, presents a similar assessment of performance metrics for Original, Compressed, and Slowdown datasets. In the field of object detection, the analysis of performance metrics across different datasets is essential for understanding the effectiveness of detection models. Table VIII offers such an analysis for Zone 2, comparing the performance of the original, compressed, and slowdown datasets. This comparison is crucial in highlighting how different data conditions affect the metrics such as mean Average Precision (mAP), Accuracy, Recall, and execution Time. The Original dataset in Zone 2 demonstrates respectable performance, characterized by a mAP of 78.3%. This figure indicates a decent level of precision in the detection model's ability to correctly identify true positives. The Accuracy of 66.1% suggests the model's general effectiveness in correctly classifying both true positives and negatives, though it also implies room for improvement. A relatively high Recall of 84.9% is observed, indicating the model's proficiency in identifying a large proportion of actual positive cases. However, this dataset shows a slightly longer execution Time of 0.024 seconds, which, while still efficient, is longer compared to other datasets. In the case of the compressed dataset, a notable improvement in mAP is observed, reaching 82.9%. This increase suggests enhanced precision in occupant detection despite the reduced data volume. The Accuracy also sees a significant rise to 84.1%, demonstrating a considerable improvement in the model's overall detection capability. However, the Recall drops to 80.9%, indicating a slight decrease in the model's ability to identify all true positive cases compared to the Original dataset. Despite these variations in mAP, Accuracy, and Recall, the compressed dataset maintains a rapid execution Time of 0.004 seconds, reflecting efficient processing capability. The Slowdown dataset, designed to test the model's performance with an increased frame count, records the highest mAP of 84.4% among the three datasets. This suggests that the augmented frame count contributes to a more precise detection capability. However, this dataset experiences a drop in Accuracy to 80.9% and a more significant decline in Recall to 68.2%, compared to the compressed dataset. These results indicate a trade-off between the increased precision and the model's ability to accurately classify and identify all positive cases. Overall, the analysis of Zone 2's performance metrics across these three datasets illustrates the inherent trade-offs between various performance measures and the characteristics of each dataset. While the Compressed dataset shows improvements in mAP and Accuracy, it slightly compromises on Recall. On the other hand, the Slowdown dataset excels in precision but at the cost of lower Accuracy and Recall.

In the specialized area of object detection within Zone 3, Table IX presents a critical assessment of the model's performance using three distinct datasets: Original, Compressed, and Slowdown. This comprehensive evaluation is integral to understanding how different data conditions affect key performance metrics such as mean Average Precision (mAP), Accuracy, Recall, and execution time. The Original dataset in Zone 3 sets a high benchmark in model performance. It demonstrates exceptional precision with a mAP of 96.2%, indicating its effectiveness in accurately identifying true positive detections. This is complemented by an Accuracy of 90.1%, reflecting the model's overall reliability in distinguishing true positives from false positives and negatives. Additionally, the Recall of 93.7% is noteworthy, as it signifies the model's ability to detect a large majority of actual positive cases, minimizing the instances of missed detections. All these metrics are achieved within an efficient execution time of 0.004 seconds, highlighting the model's rapid processing capabilities. Conversely, the compressed dataset, designed to assess performance under reduced data volume, maintains a commendable mAP of 93.5% and an even higher Accuracy of 91.1% compared to the Original dataset. This suggests that the model retains its effectiveness and precision in a compressed data environment. However, the Recall experiences a slight decrease, dropping to 86%. This reduction indicates a marginal compromise in the model's capacity to identify all true positive cases in the face of data compression. Despite this, the execution time remains impressively swift at 0.004 seconds, suggesting that the reduction in data volume does not significantly affect the overall processing speed of the system. Remarkably, the Slowdown dataset in Zone 3 outshines the other datasets in terms of performance. It achieves the highest mAP of 97.4%, suggesting superior precision in detection. This is further enhanced by the highest Accuracy of 94.4% and the best Recall of 97.1% among the datasets, indicating the model's heightened capability to accurately classify and detect actual positive cases. The attainment of these superior metrics, interestingly, does not affect the execution time, which remains constant at 0.004 seconds. This underscores the model's ability to handle increased frame counts without compromising processing efficiency.

Collectively, these findings illustrate the varying efficacies of the occupant detection system across different datasets in zone 1, 2, and 3. The original dataset provides a balanced combination of precision and efficiency, while the compressed dataset reveals that data compression slightly impacts recall but with minimal effect on processing speed. The slowdown dataset, with its enhanced frame count, demonstrates potential for superior performance.

TABLE IX. OCCUPANT CALCULATION OF ZONE 3

Dataset	IOU	Number of Occupants	Accuracy with actual (%)
Original	0,25	3	100
	0,4	3	100
	0,5	3	100
Compressed	0,25	3	100
	0,4	3	100
	0,5	3	100
Slowdown	0,25	3	100
	0,4	3	100
	0,5	3	100

a. zone 1 b. zone 2 c. zone 3

Fig. 3. Occupant detection and calculation from original video using YOLO v8.

Fig. 3 provides a visual depiction of the object detection and enumeration process across the three clearly defined zones in Zone 3. It effectively illustrates the system's advanced capability in accurately measuring occupants within designated polygonal regions, corresponding to the zones. The use of the optimal Intersection over Union (IOU) threshold for precise detection is evident in the figure, showcasing the system's proficiency in occupant detection. This visual representation, along with the detailed performance metrics, highlights the varying efficacies and robustness of the occupant detection system across different datasets within zone 1, 2, and 3, demonstrating its adaptability and precision in diverse data conditions. The dataset was used is Slowdown.

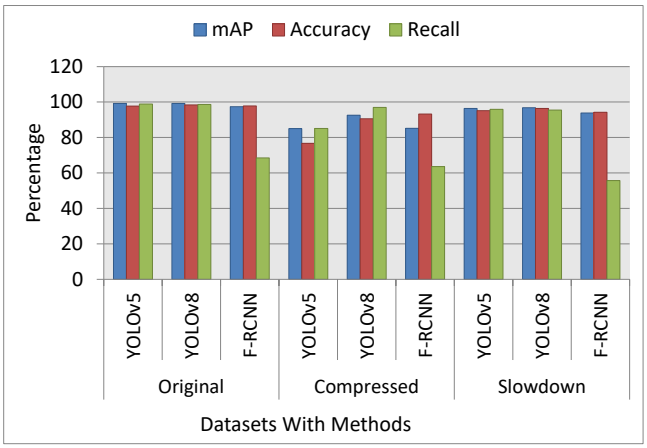

Fig. 4. Performance comparison of zone 1.

Fig. 4 to Fig. 9 offer a comprehensive comparison of the performance metrics for YOLO v8, YOLO v5, and Fast-RCNN across multiple datasets, including the Slowdown Dataset and others utilizing various occupant detection methods. The findings of this research was comparative with Faster-RCNN [8][24][25] and YOLO v5 [21]. The objective is to assess the effectiveness of these models in different detection scenarios. The analysis encompasses several key metrics, such as mean Average Precision (mAP), Accuracy, Recall, and execution Time, which play crucial roles in evaluating the performance of object detection algorithms. These metrics provide nuanced insights into the ability of each model to accurately detect and track occupants across different environments and datasets.

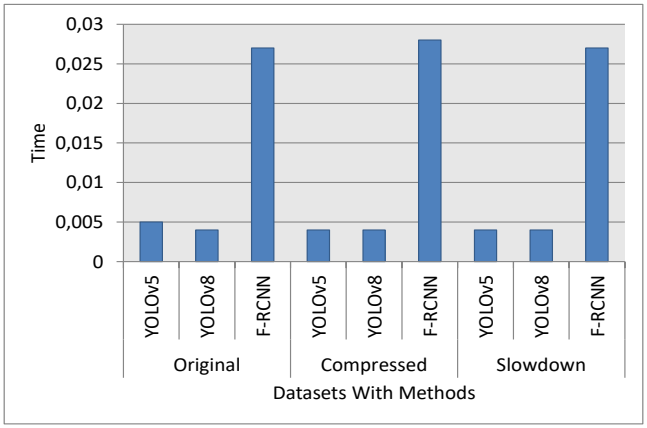

Fig. 5. Time process comparison of zone 1.

In Fig. 4 and Fig. 5, the emphasis is on assessing occupant detection in Zone 1, utilizing three distinct object detection methods. In the original dataset, YOLO v5 and YOLO v8 perform exceptionally well, both achieving a mAP of 99.2%, indicative of highly accurate detection capabilities. YOLO v8 slightly surpasses YOLO v5 in Accuracy, scoring 98.4% against 97.7%, and also demonstrates a marginal edge in processing efficiency (0.004 seconds compared to YOLO v5's 0.005 seconds). However, F-RCNN, despite a decent mAP of 97.4% and Accuracy of 97.8%, shows a significant deficiency in Recall (68.5%), suggesting it misses more true positive detections than its YOLO counterparts. Additionally, F-RCNN's longer processing time (0.027 seconds) may hinder its application in real-time scenarios. The compressed dataset reveals YOLO v8's adaptability, maintaining a high mAP of 92.6% and an Accuracy of 90.6%, with a Recall of 97%. In contrast, YOLO v5 exhibits a drop in performance, with lower

mAP (85%) and Accuracy (76.7%), although it still maintains a relatively high Recall of 85.1%. F-RCNN shows some improvement in Accuracy (93.2%), but its mAP (85.2%) and particularly low Recall (63.6%) underscore persistent limitations in comprehensive occupant detection. In the slowdown dataset, both YOLO v5 and YOLO v8 continue to demonstrate strong performance, with mAP values of 96.4% and 96.8% respectively, and Accuracy rates above 95%. Their processing times remain impressively low, underscoring their efficiency in various data conditions. F-RCNN, while showing an improved mAP of 93.8% and Accuracy of 94.2%, continues to struggle with a low Recall rate (55.7%). YOLO v5 and YOLO v8 consistently outperform F-RCNN across different datasets in Zone 1, exhibiting superior mAP, Accuracy, and Recall, coupled with faster processing times.

Fig. 6 and Fig. 7 addresses occupant detection in Zone 2, comparing the same object detection methods across distinct datasets. YOLO v8 achieves the highest mAP of 78.3%, coupled with an Accuracy of 66.1% and Recall of 84.9% in the Original dataset. Compressed data still yields high mAP (84.1%) and Accuracy (82.9%), although Recall is slightly lower at 80.9%. YOLO v5 and F-RCNN exhibit varying performance metrics across datasets, emphasizing the dataset-dependent nature of these methods. In the Slowdown dataset, YOLO v8 maintains its superior mAP and Recall, highlighting its consistent performance.

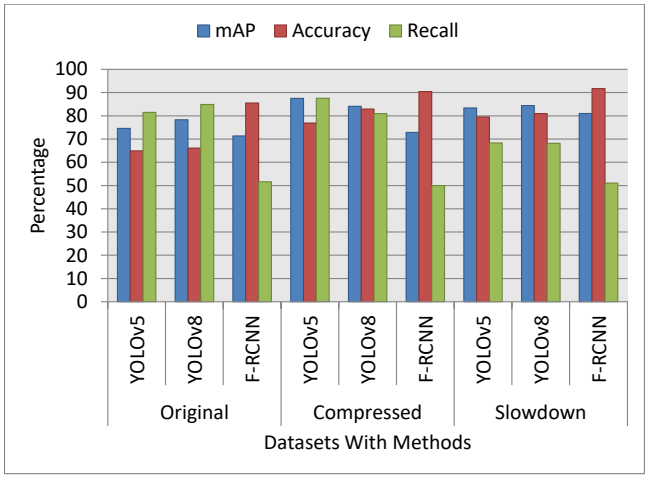

Fig. 6. Performance comparison of zone 2.

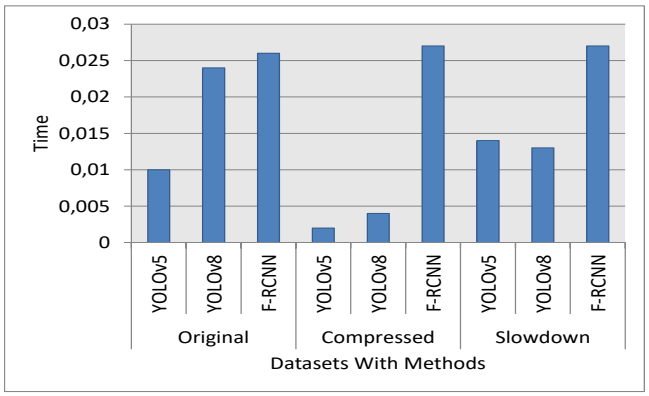

Fig. 7. Time process comparison of zone 2.

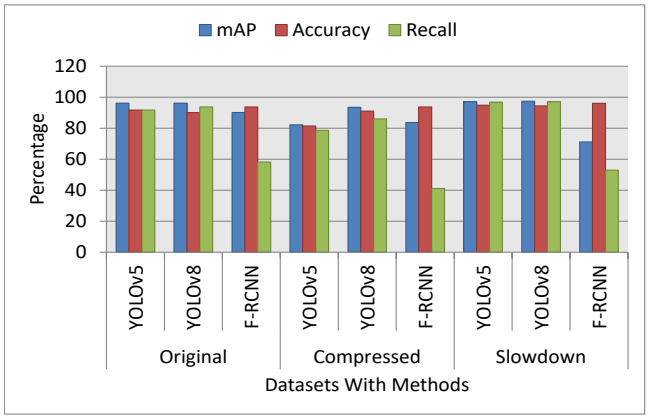

Fig. 8. Performance comparison of zone 3.

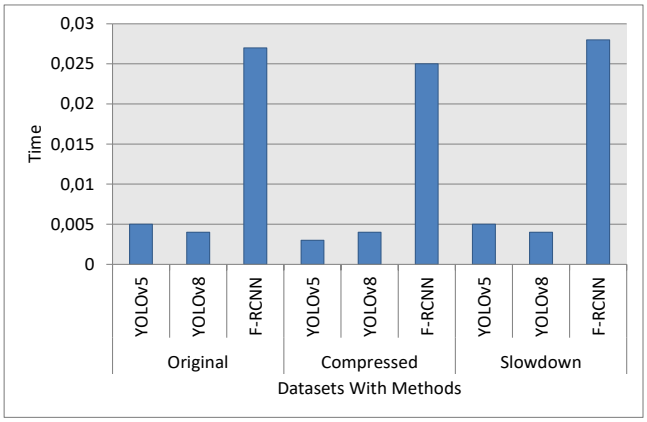

Fig. 9. Time process comparison of zone 3.

Fig. 8 and Fig. 9 focuses on Zone 3, where YOLO v5 and YOLO v8 continue to perform well in the original dataset, both YOLO v5 and YOLO v8 exhibit remarkably consistent and high-performance levels. They each achieve mAP of 96.2%, indicating a highly accurate ability to detect and identify occupants within this zone. Additionally, these methods demonstrate substantial Accuracy and Recall, reflecting their precision and reliability in correctly identifying true positives without missing significant detections. The analysis of the compressed dataset highlights the exceptional adaptability of YOLO v8. It achieves a notable mAP of 93.5%, maintaining high Accuracy and Recall rates despite the challenges posed by data compression. This performance suggests that YOLO v8 is particularly suited for scenarios where data integrity might be compromised or where bandwidth limitations necessitate data compression. In the context of the Slowdown dataset, both YOLO v5 and YOLO v8 continue to excel. They demonstrate robustness in mAP, Accuracy, and Recall, underscoring their effectiveness even under conditions that may affect the speed or flow of data input. Their high performance in this dataset is indicative of their ability to maintain reliability and accuracy in less-than-ideal operational environments. Faster R-CNN, while showing competitive performance in certain scenarios, does not consistently match the overall performance metrics of YOLO v5 and YOLO v8. This observation suggests that while F-RCNN can be effective in specific contexts, it may not be the optimal choice for all scenarios, particularly those represented in Zone 3.

In Zone 1, YOLO v8 achieves an outstanding mAP of 99.2%, indicating its high precision in detecting occupants. The Accuracy of 98.4% showcases its ability to correctly classify occupants, while the Recall of 98.6% demonstrates its capability to capture almost all actual occupants. Furthermore, YOLO v8 maintains a swift execution time of 0.004 seconds, indicating efficiency in processing. In Zone 2, YOLO v8 continues to demonstrate its efficacy with mAP of 78.3%, reflecting its strong performance in detecting occupants. The Accuracy of 66.1% suggests that it correctly classifies occupants in the zone. Moreover, the Recall of 84.9% underscores its ability to capture a significant portion of actual occupants. Despite a slightly longer execution time of 0.024 seconds compared to Zone 1, it remains efficient. Zone 3 further emphasizes the efficacy of YOLO v8, with mAP of 96.2% showcasing its precision in occupant detection. The Accuracy of 90.1% reflects its high classification accuracy, and the Recall of 93.7% indicates its ability to capture the majority of actual occupants. YOLO v8 maintains an efficient execution time of 0.004 seconds in this context. Overall, YOLO v8 demonstrates remarkable efficacy, consistently achieving high mAP values across all three zones, signifying precise occupant detection. Its competitive Accuracy and Recall values further validate its effectiveness. Additionally, its efficient execution times indicate that YOLO v8 combines both efficacy and efficiency, making it a strong candidate for occupant detection tasks in various scenarios.

B. Discussion

The study Occupancy Measurement in Under-Actuated Zones presents significant results in regards to the effectiveness of the YOLO v8 model in accurately detecting and quantifying occupants in difficult environments. The research, which was conducted through the compilation of a comprehensive dataset via video observations in the student corner room at Universitas Trilogi, demonstrates the superior performance of the YOLO v8 model in occupant detection, particularly in dynamic under-actuated zones with varying occupancy patterns and complex environmental conditions. The model's real-time detection capabilities, high accuracy in identifying occupants, and efficient object localization highlight its adaptability and robustness in diverse situations. The study's key findings include the model's ability to precisely identify and count occupants in real-time within the segmented zones of the student corner room, showcasing its spatial accuracy and object localization proficiency. The research's quantitative metrics, including mean Average Precision (mAP), Accuracy, Recall, and execution time, highlight the model's effectiveness in accurately identifying true positive detections while minimizing false positives and negatives. Additionally, the YOLO v8 model's swift execution time further emphasizes its efficiency in data processing and real-time results delivery. Overall, the research findings suggest that the YOLO v8 model has the potential to revolutionize occupant detection systems in under-actuated zones, offering a promising solution for optimizing occupancy monitoring and management in complex environments. The study lays the groundwork for future research and development in the field of object detection and occupancy measurement, specifically focusing on addressing the unique challenges presented by under-actuated zones. The results of this study provide valuable insights into the capabilities of the YOLO v8 model and its potential applications in various industries. The research findings are a significant contribution to the field of occupancy measurement and highlight the potential of the YOLO v8 model as a solution for optimizing occupancy monitoring and management in challenging environments. The study's results also suggest that the YOLO v8 model could be a useful tool for a variety of industries, including but not limited to, security, safety management, and facilities management. The research findings are a valuable resource for academics, researchers, and professionals working in the field of occupancy measurement and object detection.

V. CONCLUSION

This study presents a comprehensive evaluation of occupant detection methods across three distinct zones using YOLO v8. The quantitative analysis demonstrates the efficacy and efficiency of YOLO v8 in occupant detection tasks. In Zone 1, YOLO v8 exhibits exceptional performance with a high mAP of 99.2%, indicating precise detection. The Accuracy of 98.4% and Recall of 98.6% further underscore its effectiveness. Additionally, YOLO v8 maintains an efficient execution time of 0.004 seconds, making it a suitable choice for real-time applications. Zone 2 showcases YOLO v8's efficacy with a respectable mAP of 78.3%, suggesting robust occupant detection. Despite a lower Accuracy of 66.1%, the Recall of 84.9% demonstrates its ability to capture a significant proportion of actual occupants. YOLO v8's execution time of 0.024 seconds in this zone remains efficient. In Zone 3, YOLO v8 continues to perform effectively, achieving mAP of 96.2%, indicating precise detection. The Accuracy of 90.1% and Recall of 93.7% highlight its capability to classify and capture occupants accurately. YOLO v8's efficient execution time of 0.004 seconds makes it a valuable choice for this scenario. The results suggest that YOLO v8 is a robust and efficient method for occupant detection in various zones. Its high precision and competitive recall values make it a promising solution for real-world applications. Future work in this research can explore further optimization of YOLO v8 for occupant detection by considering different datasets and environmental conditions. Additionally, the integration of advanced deep learning techniques and hardware acceleration can enhance both the accuracy and speed of occupant detection systems. Further research can also focus on addressing challenges related to occlusions and multi-object tracking in complex scenarios, advancing the field of occupant detection in smart environments.

ACKNOWLEDGMENT

Thank to Indonesian Ministry of Education, Culture, Research and Technology for giving research funding and Universitas Trilogi for giving permission to use the library the object of research.

REFERENCES

[1] G. Serale, M. Fiorentini, A. Capozzoli, D. Bernardini, and A. Bemporad, “Model Predictive Control (MPC) for enhancing building and HVAC system energy efficiency: Problem formulation, applications and opportunities,” Energies, vol. 11, no. 3, 2018, doi: 10.3390/en11030631.

[2] K. Sun, Q. Zhao, and J. Zou, “A review of building occupancy measurement systems,” Energy Build., vol. 216, p. 109965, 2020, doi:

10.1016/j.enbuild.2020.109965.

[3] J. Brooks, S. Kumar, S. Goyal, R. Subramany, and P. Barooah, "Energy-efficient control of under-actuated HVAC zones in commercial buildings," Energy Build., vol. 93, pp. 160–168, 2015, doi: https://doi.org/10.1016/j.enbuild.2015.01.050.

[4] J. Wang, N. C. F. Tse, T. Y. Poon, and J. Y. C. Chan, "A practical multi-sensor cooling demand estimation approach based on visual, indoor and outdoor information sensing," Sensors (Switzerland), vol. 18, no. 11, 2018, doi: 10.3390/s18113591.

[5] S. Sadrizadeh et al., "Indoor air quality and health in schools: A critical review for developing the roadmap for the future school environment," J. Build. Eng., vol. 57, 2022, doi: 10.1016/j.jobe.2022.104908.

[6] Y. Al horr, M. Arif, M. Katafygiotou, A. Mazroei, A. Kaushik, and E. Elsarrag, "Impact of indoor environmental quality on occupant well-being and comfort: A review of the literature," International Journal of Sustainable Built Environment, vol. 5, no. 1. Elsevier B.V., pp. 1–11, 2016. doi: 10.1016/j.ijsbe.2016.03.006.

[7] Z. Yang and B. Becerik-Gerber, "How does building occupancy influence energy efficiency of HVAC systems?," in Energy Procedia, Elsevier Ltd, 2016, pp. 775–780. doi: 10.1016/j.egypro.2016.06.111.

[8] F. Felgueiras, Z. Mourão, A. Moreira, and M. F. Gabriel, "Indoor environmental quality in offices and risk of health and productivity complaints at work: A literature review," J. Hazard. Mater. Adv., vol. 10, 2023, doi: 10.1016/j.hazadv.2023.100314.

[9] L. T. Molina, E. Velasco, A. Retama, and M. Zavala, "Experience from integrated air quality management in the Mexico City Metropolitan Area and Singapore," Atmosphere, vol. 10, no. 9. MDPI AG, 2019. doi: 10.3390/atmos10090512.

[10] Z. Pang, Z. O'Neill, Y. Chen, J. Zhang, H. Cheng, and B. Dong, "Adopting occupancy-based HVAC controls in commercial building energy codes: Analysis of cost-effectiveness and decarbonization potential," Appl. Energy, vol. 349, p. 121594, 2023, doi: 10.1016/j.apenergy.2023.121594.

[11] S. Taheri, P. Hosseini, and A. Razban, "Model predictive control of heating, ventilation, and air conditioning (HVAC) systems: A state-of-the-art review," J. Build. Eng., vol. 60, p. 105067, 2022, doi: 10.1016/j.jobe.2022.105067.

[12] J. Shi, N. Yu, and W. Yao, "Energy Efficient Building HVAC Control Algorithm with Real-time Occupancy Prediction," Energy Procedia, vol. 111, pp. 267–276, 2017, [Online]. Available: https://api.semanticscholar.org/CorpusID:114478674

[13] A. Capozzoli, M. S. Piscitelli, A. Gorrino, I. Ballarini, and V. Corrado, "Data analytics for occupancy pattern learning to reduce the energy consumption of HVAC systems in office buildings," Sustain. Cities Soc., vol. 35, pp. 191–208, 2017, [Online]. Available: https://api.semanticscholar.org/CorpusID:115920652

[14] C. Turley, M. Jacoby, G. Pavlak, and G. Henze, "Development and Evaluation of Occupancy-Aware HVAC Control for Residential Building Energy Efficiency and Occupant Comfort," Energies, vol. 13, no. 20. 2020. doi: 10.3390/en13205396.

[15] I. Chatisa, Y. A. Syahbana, and A. U. A. Wibowo, "Object Detection and Monitor System for Building Security Based on Internet of Things (IoT) Using Illumination Invariant Face Recognition," Kinet. Game Technol. Inf. Syst. Comput. Network, Comput. Electron. Control, 2023, doi: 10.22219/kinetik.v8i1.1622.

[16] B. Pollard, L. Engelen, F. Held, and R. de Dear, "Activity space, office space: Measuring the spatial movement of office workers.," Appl. Ergon., vol. 98, p. 103600, 2021, [Online]. Available: https://api.semanticscholar.org/CorpusID:238580082

[17] A. Schirmer, A. Herde, J. A. Eccard, and M. Dammhahn, "Individuals in space: personality-dependent space use, movement and microhabitat use facilitate individual spatial niche specialization," Oecologia, vol. 189, pp. 647–660, 2019, [Online]. Available: https://api.semanticscholar.org/CorpusID:71146317

[18] A. Ibrahim, H. H. Ali, F. Abuhendi, and S. Jaradat, "Thermal seasonal variation and occupants' spatial behaviour in domestic spaces," Build. Res. Inf., vol. 48, pp. 364–378, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:208834947

[19] U. H. Gawande, K. Hajari, and Y. Golhar, "Pedestrian Detection and Tracking in Video Surveillance System: Issues, Comprehensive Review, and Challenges," EngRN Dyn. Syst., 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:214213201

[20] S. Drira and I. F. C. Smith, "A framework for occupancy detection and tracking using floor-vibration signals," Mech. Syst. Signal Process., vol. 168, p. 108472, 2022, doi: https://doi.org/10.1016/j.ymssp.2021.108472.

[21] S. Kumar, Vishal, P. Sharma, and N. Pal, "Object tracking and counting in a zone using YOLOv4, DeepSORT and TensorFlow," in Proceedings - International Conference on Artificial Intelligence and Smart Systems, ICAIS 2021, Institute of Electrical and Electronics Engineers Inc., 2021, pp. 1017–1022. doi: 10.1109/ICAIS50930.2021.9395971.

[22] M. Pervaiz, Y. Y. Ghadi, M. Gochoo, A. Jalal, S. Kamal, and D.-S. Kim, "A Smart Surveillance System for People Counting and Tracking Using Particle Flow and Modified SOM," Sustainability, vol. 13, no. 10, p. 5367, 2021, doi: 10.3390/su13105367.

[23] H. Elkhoukhi, M. Bakhouya, D. El Ouadghiri, and M. Hanifi, "Using Stream Data Processing for Real-Time Occupancy Detection in Smart Buildings," Sensors, vol. 22, no. 6, p. 2371, 2022, doi: 10.3390/s22062371.

[24] S. Wei, P. Tien, T. W. Chow, Y. Wu, and J. K. Calautit, "Deep learning and computer vision based occupancy CO2 level prediction for demand-controlled ventilation (DCV)," J. Build. Eng., vol. 56, p. 104715, 2022, doi: 10.1016/j.jobe.2022.104715.

[25] I. Papakis, A. Sarkar, A. Svetovidov, J. S. Hickman, and A. L. Abbott, "Convolutional neural network-based in-vehicle occupant detection and classification method using second strategic highway research program cabin images," Transportation Research Record, vol. 2675, no. 8. SAGE Publications Ltd, pp. 443–457, 2021. doi: 10.1177/0361198121998698.

[26] J. Du, "Understanding of Object Detection Based on CNN Family and YOLO," in Journal of Physics: Conference Series, Institute of Physics Publishing, 2018. doi: 10.1088/1742-6596/1004/1/012029.

[27] F. Joiya, "Object Detection: YOLO VS FASTER R-CNN," Int. Res. J. Mod. Eng. Technol. Sci., 2022, doi: 10.56726/irjmets30226.

[28] W. Li, "Analysis of Object Detection Performance Based on Faster R-CNN," in Journal of Physics: Conference Series, IOP Publishing Ltd, 2021. doi: 10.1088/1742-6596/1827/1/012085.

[29] C. Cao et al., "An Improved Faster R-CNN for Small Object Detection," IEEE Access, vol. 7, pp. 106838–106846, 2019, doi: 10.1109/ACCESS.2019.2932731.

[30] R. Girshick, "Fast R-CNN," in 2015 IEEE International Conference on Computer Vision (ICCV), 2015, pp. 1440–1448. doi: 10.1109/ICCV.2015.169.

[31] L. Rueda, K. Agbossou, A. Cardenas, N. F. Henao, and S. Kelouwani, "A comprehensive review of approaches to building occupancy detection," Build. Environ., vol. 180, p. 106966, 2020, doi: 10.1016/j.buildenv.2020.106966.

[32] D. Reis, J. Kupec, J. Hong, and A. Daoudi, "Real-Time Flying Object Detection with YOLOv8," 2023, [Online]. Available: http://arxiv.org/abs/2305.09972.