JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

PERFORMANCE EVALUATION OF RECENT YOLO VERSIONS FOR CLASSROOM STUDENT BEHAVIOR DETECTION

Mahendra Adiastoro[1*]; Febry Putra Rochim[1]; Syahroni Hidayat[1]

Department of Electrical Engineering[1]
Universitas Negeri Semarang, Semarang, Indonesia[1]
https://unnes.ac.id/[1]
mahentay@students.unnes.ac.id*, febry.putra@mail.unnes.ac.id, syahronihidayat@mail.unnes.ac.id

(*) Corresponding Author
(Responsible for the Quality of Paper Content)

Abstract— *The increasing adoption of smart classroom systems underscores the need for automated, objective, and real-time monitoring of student behavior to support effective teaching and learning. Computer vision–based object detection, particularly the You Only Look Once (YOLO) family, has shown strong potential for this task. However, existing studies predominantly evaluate YOLO models in isolation or across different frameworks, resulting in biased comparisons. To address this gap, this study presents a controlled intra-family comparative evaluation of four recent YOLO generations YOLOv8, YOLOv10, YOLOv11, and YOLOv12 across three weight variants (nano, small, and medium), yielding 12 model configurations. All experiments were conducted under a uniform training pipeline and computing environment using an NVIDIA T4 GPU to ensure fair benchmarking. Model performance was assessed using Precision, Recall, F1-Score, mean Average Precision (mAP), inference speed (FPS), and computational complexity. The results reveal a consistent trade-off between detection accuracy and inference speed: YOLOv12m achieves the highest detection accuracy but the lowest FPS due to increased architectural complexity. At the same time, YOLOv10n offers the fastest inference at the cost of reduced reliability for subtle behaviors. Within the scope of the evaluated dataset and controlled classroom setting, YOLOv8s and YOLOv11s demonstrate the most balanced accuracy–speed performance, making them suitable candidates for real-time classroom monitoring under similar conditions. This study provides practical insights for researchers and developers by offering an objective benchmark and model-selection guidance tailored to smart classroom applications, while accounting for dataset and environmental constraints.*

Keywords: *Accuracy–Speed Trade-off, Classroom Behavior Detection, Computer Vision, Smart Classroom, YOLO.*

Intisari— *Penerapan sistem kelas pintar yang semakin luas menyoroti kebutuhan akan pemantauan otomatis, objektif, dan real-time terhadap perilaku siswa untuk mendukung proses pengajaran dan pembelajaran yang efektif. Deteksi objek berbasis penglihatan komputer, khususnya keluarga You Only Look Once (YOLO), telah menunjukkan potensi yang kuat untuk tugas ini. Namun, studi yang ada sebagian besar mengevaluasi model YOLO secara terpisah atau di berbagai kerangka kerja, menghasilkan perbandingan yang bias. Untuk mengatasi kesenjangan ini, studi ini menyajikan evaluasi perbandingan terkontrol antar generasi YOLO (YOLOv8, YOLOv10, YOLOv11, dan YOLOv12) melalui tiga varian bobot (nano, kecil, dan sedang), menghasilkan 12 konfigurasi model. Semua eksperimen dilakukan dalam pipeline pelatihan dan lingkungan komputasi yang seragam menggunakan GPU NVIDIA T4 untuk memastikan pembandingan yang adil. Kinerja model dievaluasi menggunakan Precision, Recall, F1-Score, mean Average Precision (mAP), kecepatan inferensi (FPS), dan kompleksitas komputasi. Hasil menunjukkan adanya trade-off yang konsisten antara akurasi deteksi dan kecepatan inferensi: YOLOv12m mencapai akurasi deteksi tertinggi tetapi memiliki FPS terendah akibat kompleksitas arsitektur yang meningkat. Di sisi lain, YOLOv10n menawarkan kecepatan inferensi tercepat dengan mengorbankan keandalan untuk perilaku yang halus. Dalam lingkup dataset yang dievaluasi dan lingkungan kelas yang terkontrol, YOLOv8s dan YOLOv11s menunjukkan kinerja akurasi-*

kecepatan yang paling seimbang, menjadikannya kandidat yang cocok untuk pemantauan kelas real-time dalam kondisi serupa. Studi ini memberikan wawasan praktis bagi peneliti dan pengembang dengan menyediakan benchmark objektif dan panduan pemilihan model yang disesuaikan untuk aplikasi kelas pintar, sambil memperhitungkan batasan dataset dan lingkungan.

***Kata Kunci**: Pertukaran Antara Akurasi Dan Kecepatan, Deteksi Perilaku Di Kelas, Penglihatan Komputer, Kelas Pintar, YOLO.*

INTRODUCTION

The integration of artificial intelligence (AI) and computer vision has significantly transformed the education sector, driving the development of smart classrooms that emphasize adaptive, objective, and data driven learning environments [1]. As a core subfield of AI, computer vision enables automatic visual analysis of classroom activities, offering the potential for continuous, objective monitoring of student engagement [2], [3]. Despite these technological advances, face to face learning remains the dominant mode of teaching, where student engagement levels play a crucial role in determining learning effectiveness. In practice, monitoring behaviors such as attention, note-taking, cell phone use, or even sleeping is still done manually by instructors. This approach is inherently subjective, time consuming, and increasingly impractical in large classes where individual supervision is limited [4]. As a result, disengaged behaviors may go undetected, even though student engagement is widely recognized as a key factor influencing academic performance. These limitations highlight the urgent need for an automated, objective, and real-time classroom monitoring system. Computer vision, particularly deep learning-based object detection, offers a promising solution, as demonstrated by its success in various real-world application domains [5]. Among existing approaches, the You Only Look Once (YOLO) family of detectors has attracted significant attention for its high detection accuracy and real-time inference [6].

Previous studies have demonstrated the feasibility of applying YOLO-based models to classroom activity detection, ranging from simple behaviors [7] to more complex student actions [8], with recent multimodal approaches further enhancing recognition performance through spatiotemporal feature fusion [9]. However, real world classroom environments present persistent challenges, including variations in lighting conditions, partial occlusions caused by dense seating arrangements, and subtle visual similarities between behaviors, such as phone use and resting postures [10]. These challenges demand detection models that are not only accurate but also computationally efficient and robust. In response to these challenges, research on student behavior detection has evolved alongside advances in YOLO architecture. Early studies mostly used older versions of YOLO; for example, a YOLOv4-based framework was developed to monitor teacher student interactions through IoT integration [1], while the Student Activity Detection using YOLO (SADY) system demonstrated basic gesture detection in classrooms [7].

Subsequent research shifted to YOLOv5, where architectural improvements and additional modules were introduced to enhance robustness against visual disturbances. For example, the YOLOv5-based Student Behavior Detection (SBD) framework integrates contextual attention and OpenPose to handle complex interaction patterns [10]. Another study uses YOLOv7 as a baseline for standard behavior annotation, emphasizing the importance of consistent class definitions in classroom datasets [4]. Recent studies focus on YOLOv8 and its variants, where modifications such as DF-YOLOv8s [11] and WAD-YOLOv8 [12] demonstrate improved detection accuracy for complex movements without significantly compromising inference speed. Additionally, CSSA-YOLO, built upon YOLOv8m, introduces a cross scale spatio temporal attention mechanism and reports competitive performance in detailed class behavior recognition tasks [13]. In this context, YOLOv8 can be viewed as a relatively mature and widely adopted architecture, making it a relevant comparative baseline for evaluating the impact of architectural refinements on the next generation of YOLO.

With the introduction of newer YOLO generations, architectural innovations increasingly focus on balancing detection accuracy and computational efficiency. YOLOv10 attracts attention for its emphasis on inference efficiency, particularly through its NMS free design, which significantly reduces post-processing load, making it well-suited for real-time applications and edge devices [14], [15]. Furthermore, YOLOv11 serves as a transitional generation, aiming to improve architectural efficiency and training stability without sacrificing detection accuracy. YOLOv11 optimizes the backbone and feature-extraction

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

pipeline to be lighter and more consistent, enabling it to maintain competitive performance while maintaining relatively low inference latency. These characteristics make YOLOv11 a relevant compromise solution between very lightweight models such as YOLOv10 and higher capacity models such as YOLOv12, especially in scenarios that require a balance between speed and detection accuracy [15], [16].

On the other hand, YOLOv12 introduces more advanced architectural components, such as Residual Efficient Layer Aggregation Networks (RELAN) and Area Attention (A2), which are designed to improve feature representation and spatial awareness in dense scenes [16], [17]. These mechanisms are particularly relevant for classroom environments, where critical behaviors often involve small objects, such as cell phones, or subtle visual cues, such as closed eyes indicating sleep. Area Attention allows the model to focus on dense, informative spatial regions, while RELAN improves feature aggregation across layers, potentially increasing robustness to occlusion and object-scale variations. Although these architectural developments promise improved performance, most existing studies still evaluate YOLO models separately or through cross framework comparisons, such as comparing YOLOv9 based variants with earlier YOLO versions [17]. Such approaches have the potential to introduce methodological bias due to differences in the training pipeline and architectural foundations used.

As a result, a clear research gap remains. To date, there has been no comprehensive intra-family comparative evaluation that systematically assesses the latest versions of YOLO (YOLOv8, YOLOv10, YOLOv11, and YOLOv12) in a fully uniform experimental framework, particularly for detecting student behavior in the classroom. This domain has distinct visual characteristics compared to industrial inspection or public surveillance scenarios, including dense seating arrangements, limited camera angles, and subtle, temporary behavioral cues [5], [8]. From a practical standpoint, understanding the trade-off between accuracy and inference efficiency is crucial for lecturers and system developers in selecting models that can operate reliably in real classroom environments without excessive computational demands. To address this gap, this study proposes a systematic comparative evaluation of the four latest generations of YOLO (YOLOv8, YOLOv10, YOLOv11, and YOLOv12) deliberately excluding YOLOv9 to avoid inconsistencies caused by differences in the development framework and programmable gradient information (PGI) [18]. The analysis focuses on three practically relevant weight variants, namely nano (n), small (s), and medium (m), which represent different compromises between model capacity and computational cost, and are more suitable for deployment in resource-constrained smart classroom environments than larger variants [19], [20]. Based on these considerations, this study explicitly formulates research hypotheses covering architectural developments in newer generations of YOLO including efficient inference design in YOLOv10, optimization of training structure and stability in YOLOv11, and attention based feature aggregation mechanisms in YOLOv12 provide measurable improvements in the trade-off between detection accuracy and inference speed compared to the earlier YOLOv8 variant when evaluated in a controlled classroom environment.

To test this hypothesis, model performance was comprehensively evaluated on three main dimensions: detection accuracy, computational efficiency, and inference speed. Accuracy was evaluated using Precision, Recall, F1-Score, and mean Average Precision (mAP) [21], with a primary focus on mAP@0.5 [22]. Computational efficiency was analyzed by model size and parameter count, while inference speed was measured in Frames Per Second (FPS) as an indicator of real-time readiness. The data used in this study were collected from a real classroom environment and annotated into four representative behavior categories: sitting, sleeping, talking, and using a cell phone. The main objective of this study is to empirically identify the YOLO version and weight variant that offers the optimal balance between accuracy, efficiency, and speed for classroom behavior monitoring applications.

The uniqueness of this study lies in three main aspects: (1) direct comparison between YOLO generations within the same family under uniform experimental conditions, (2) focused evaluation in the domain of classroom behavior detection, which is still relatively unexplored compared to industrial or surveillance contexts [14], [23], [24], and (3) practical recommendations for lightweight and medium-scale YOLO variants suitable for real-time smart classroom systems. It is important to note that this study focuses on controlled classroom environments and aims to provide comparative insights rather than universal generalizations across all educational contexts. The findings of this study are expected to provide empirical guidance to researchers and practitioners in selecting appropriate object detection models for classroom behavior monitoring.

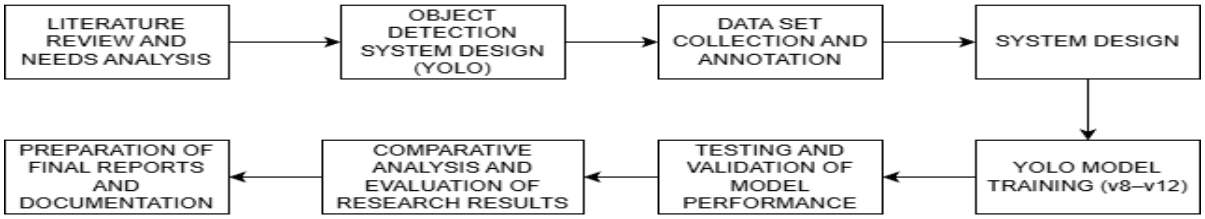
VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

MATERIALS AND METHODS

Research Design

This study uses a comparative experimental approach within the framework of computer vision-based system engineering to evaluate the performance of four YOLO (You Only Look Once) versions YOLOv8, YOLOv10, YOLOv11, and YOLOv12 in detecting student behavior in the classroom. The experimental design aims to produce objective, measurable, and replicable results by ensuring consistency across datasets, training configurations, and computing environments. Figure 1 shows the overall research workflow, which includes preliminary analysis to final performance evaluation and reporting. All training and testing procedures were performed on the Google Colaboratory platform using NVIDIA T4 GPUs to maintain a controlled and uniform computing environment across model variants. It is important to emphasize that this setup is intended as a benchmarking environment for fair comparison, not as a direct representation of implementation on low-power edge devices.



Source: (Research Results, 2025)

Figure 1. Research Design

Figure 1 shows that this research consists of seven main stages arranged sequentially. The first stage is literature study and needs analysis, which aims to identify the theoretical basis and define system specifications. The second stage involves designing a YOLO based object detection system, followed by the third stage, which includes collecting and annotating student behavior datasets. Next, the fourth stage covers system design and training of the YOLO model for all variants (v8–v12). The fifth stage is testing and validating model performance using a test dataset. The test results are then analyzed comparatively in the sixth stage to identify significant differences between versions. The final stage is the preparation of research results reports and final documentation. With this structured workflow, the performance evaluation of each model can be carried out consistently and replicated.

Dataset and Preprocessing

The data used in this study were collected through direct observation in a real classroom environment, specifically in Microteaching Room A305, Building E11, Faculty of Engineering, Universitas Negeri Semarang. Video recording was carried out using a Raspberry Pi 5 paired with a 1080p webcam. The camera was positioned in a fixed, frontal orientation, facing the blackboard, to ensure consistent coverage of student activities. Figure 2 illustrates the classroom conditions during data collection, both when no students were present and during active learning sessions.

(a) (b)

Source: (Research Results, 2025)

Figure 2. Classroom conditions used for data collection: (a) empty classroom, (b) classroom during observation

From the recorded video, image frames representing four target behavior categories (sitting, sleeping, using a cell phone, and talking) were extracted and labeled using bounding boxes in the Roboflow platform, resulting in a dataset of 580 images with 1,400 labeled object instances. Although this dataset reflects realistic classroom conditions, its relatively limited size is recognized as a limitation, especially when evaluating some deep learning models, including medium-scale architectures. To mitigate the risk of overfitting due to the limited dataset size, several data augmentation techniques were applied during training. These augmentations include random rotation to simulate variations in student posture and camera angle, blurring to model motion and focus inconsistencies, and mosaic augmentation to increase object density and contextual diversity within a single training sample. This augmentation strategy aims to improve model generalization by exposing the network to a wider range of visual variations while preserving the semantic characteristics of classroom behavior.

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

Source: (Research Results, 2025)

Figure 3. Example of Four-Category Student Behavior Dataset Images

Prior to training, all images were automatically oriented and resized to 1088×1088 pixels. Although this resolution is higher than the common input size of 640×640, it was deliberately chosen to preserve fine visual details and improve the detection of small or subtle objects, such as cell phones or closed eyes, which are critical indicators of student behavior. This choice, however, entails an inherent trade-off: increased computational load and potentially slower inference. Therefore, the impact of this resolution on accuracy and Frames Per Second (FPS) performance is explicitly analyzed in the experimental results. After annotation and preprocessing, the dataset was split into training, validation, and test sets at 70/20/10 using Roboflow's automatic splitting function, as summarized in Table 1.

Table 1. Split Dataset Distribution

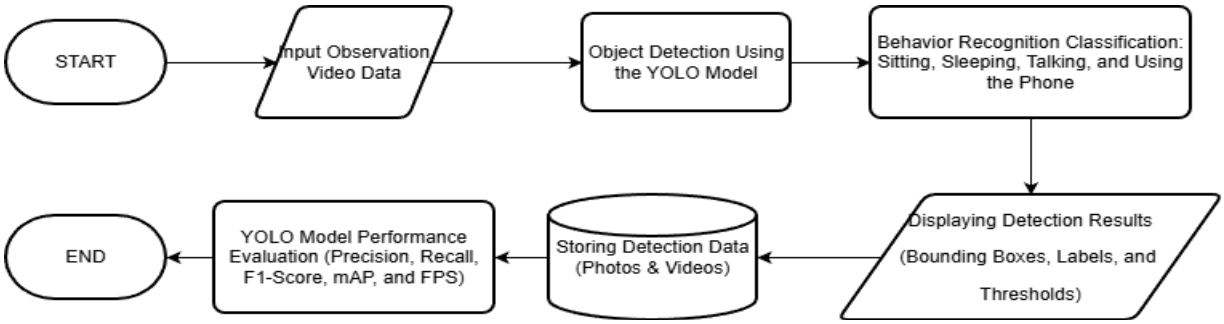
Class Categories	Code Class	Total Data	Train 70%	Valid 20%	Test 10%
Sitting	0	500	350	100	50
Sleeping	1	300	210	60	30
Using Phone	2	300	210	60	30
Talking	3	300	210	60	30
Total		1400	980	280	140

Source: (Research Results, 2025)

Given the limited size of the dataset, it is acknowledged that the test subset consisting of 1400 object instances may not fully capture the variability of complex real-world class behavior. Therefore, the evaluation results should be interpreted in the context of a controlled experimental benchmark rather than as a definitive indicator of large scale implementation performance.

System Design

This system design aims to visualize the workflow for student behavior detection using computer vision and the YOLO (You Only Look Once) algorithm. Overall, this system receives input in the form of classroom observation videos and automatically identifies student behavior into four main categories: sitting, sleeping, talking, and using cell phones.



Source: (Research Results, 2025)

Figure 4. System Design

The system architecture is designed to represent a typical computer vision-based workflow for behavior detection in classrooms using the YOLO model. As shown in Figure 4, recorded classroom videos are fed into the system, which processes each video frame through the YOLO based object detection model. This model generates bounding boxes, behavior class labels, and confidence scores for detected objects. This output is then displayed in annotated video frames and stored as structured output data for further performance evaluation. System performance is evaluated using accuracy metrics (Precision, Recall, F1-Score, and mean Average Precision) and inference speed metrics (FPS), providing a comprehensive assessment of detection effectiveness and efficiency.

Model Configuration and Training

Each version of YOLO evaluated in this study was implemented using its respective official repository: Ultralytics for YOLOv8, YOLOv11, and

YOLOv12, and Tsinghua for YOLOv10. All models were trained using a transfer learning strategy, initialized with COCO-pretrained weights, to accelerate convergence and leverage common visual features. Three weight variants nano (n), small (s), and medium (m) were selected to represent different levels of computational capacity and reflect practical considerations for resource-constrained environments. To ensure fairness and comparability, all models were trained using identical hyperparameter configurations, as listed in Table 2. Training was conducted for 50 epochs with early stopping controlled by a patience parameter to reduce the risk of overfitting. Uniform training settings ensure that observed performance differences are primarily attributable to architectural variations rather than differences in training conditions.

Table 2. YOLO Training Hyperparameter

Training Hyperparameter	Values
imgsz	1088
batch size	8
Epoch	50
Optimizer	SGD
lr0	0.003
lrf	0.01
Weight decay	0.0005
Warmup epochs	2
Patience	7
workers	1

Source: (Research Results, 2025)

The training process was conducted on the Google Colaboratory platform, which supports NVIDIA T4 GPUs, providing an optimal balance between computing speed and memory efficiency. The use of a uniform training configuration ensures that performance differences between models can be directly attributed to their respective architectures, rather than differences in training configurations. Therefore, this experiment produces stable, replicable evaluations that illustrate a fair balance among accuracy, computational efficiency, and inference speed across versions of YOLO.

Evaluation Metrics

Model performance is evaluated using four main accuracy metrics: Precision, Recall, F1-Score, and mean Average Precision (mAP). These metrics are selected to provide a balanced assessment of detection accuracy, completeness, and overall robustness across various behavior categories. Precision, Recall, and F1-Score are defined in Equations (1)–(3), while mAP is calculated as the average Average Precision value per class, as stated in Equation (4), in both the mAP@0.5 and mAP@0.5:0.95 evaluation schemes.

1. Precision measures the proportion of correct positive predictions. A high-precision value indicates that most objects detected as positive indeed belong to the correct class.

$$Precision = \frac{TP}{TP+FP} \quad (1)$$

where TP is True Positive (correct detection) and FP is False Positive (incorrect detection).

2. Recall evaluates the model's ability to find all relevant objects that actually appear in the image. A high recall value indicates that the model can minimize missed detection or False Negatives.

$$Recall = \frac{TP}{TP+FN} \quad (2)$$

Where FN is False Negative (real object but not detected).

3. F1-Score is the harmonic mean between Precision and Recall, thus providing a balanced assessment of detection accuracy and prediction completeness.

$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad (3)$$

An F1 score close to 1 indicates a good balance between the accuracy and sensitivity of the model.

4. Mean Average Precision (mAP) is used to measure overall detection performance based on the area under the Precision–Recall curve. The evaluation is performed on two schemes: mAP@0.5 (IoU ≥ 0.5) and mAP@0.5:0.95 (IoU varying from 0.5 to 0.95 with an interval of 0.05).

$$mAP = \frac{1}{N_c} \sum_{i=1}^{N_c} AP_i \quad (4)$$

Where AP_i is the Average Precision for class-i, and N_c is the number of classes tested.

In addition to accuracy metrics, inference speed is measured using Frames Per Second (FPS), and computational efficiency is analyzed based on model size and number of parameters. FPS evaluation is performed using consistent inference configurations, as shown in Table 3, to ensure a fair comparison between all YOLO variants.

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

Table 3. FPS Evaluation Parameters

FPS Evaluation Parameters	Values
Image Size	1088
Confidance Treshold	0.2
IoU NMS	0.3
max_det	1000

Source: (Research Results, 2025)

This evaluation approach provides a comprehensive overview of the performance of each YOLO version, covering detection accuracy, prediction balance, model robustness, and computational efficiency. The selection of these metrics ensures that the model performance analysis is relevant to the implementation requirements of a student behavior monitoring system in a computer vision-based smart classroom environment.

Data Analysis Method

The experimental analysis involves a controlled comparison of 12 YOLO model variants, comprising four YOLO generations and three architectural scales. All models were evaluated on the same test dataset and run on identical hardware under NVIDIA T4 GPUs. The comparative analysis focuses on three main aspects: (1) detection performance based on accuracy metrics and model size, (2) inference speed measured through FPS and processing time per frame, and (3) error patterns and robustness evaluated using normalized confusion matrices and per-class performance metrics. The results of this analysis are synthesized to identify the model that offers the best balance between accuracy, computational efficiency, and inference speed within the constraints of a controlled class benchmark.

RESULTS AND DISCUSSION

This section presents the results of a comparative evaluation of 12 YOLO model variants, covering four main versions (YOLOv8, YOLOv10, YOLOv11, and YOLOv12) at three weight scales (nano, small, and medium). This analysis focuses on three main aspects: (1) detection performance and computational efficiency, (2) real-time inference speed, and (3) qualitative error patterns and model robustness under controlled class conditions. It is important to note that all experiments in this study were designed as controlled benchmarks conducted on a uniform GPU-based environment to ensure fair comparison across models. The reported performance results are not intended to represent direct deployment on edge devices, but rather to provide a comparative reference under consistent experimental conditions.

YOLO Model Performance Evaluation Results

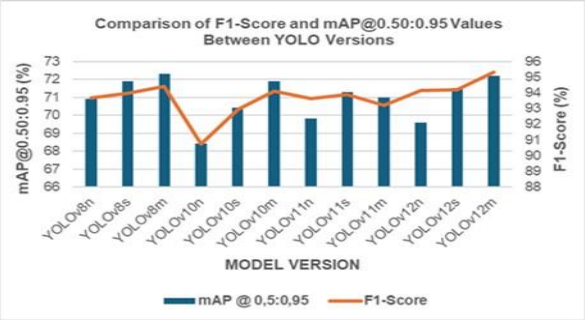
Performance evaluation was conducted to assess the accuracy and effectiveness of all models in detecting four categories of student behavior: sitting, sleeping, talking, and using mobile phones. This evaluation uses five standard metrics Precision, Recall, F1-Score, mAP@0.50, and mAP@0.50:0.95 which collectively describe detection accuracy, completeness, prediction balance, and overall localization accuracy. Quantitative results for all evaluated models are presented in Table 4. Overall, all models demonstrated strong detection capabilities, with Precision values exceeding 0.90 and average mAP@0.50 values above 0.96, indicating reliable student behavior recognition under consistent lighting conditions and fixed seating arrangements.

Table 4. YOLO Model Performance Evaluation Results

Model Version	Precision	Recall	F1-Score	mAP 0.50	mAP 0.50-0.95
YOLOv8n	0,909	0,966	0,937	0,972	0,709
YOLOv8s	0,926	0,954	0,940	0,979	0,719
YOLOv8m	0,941	0,947	0,944	0,974	0,723
YOLOv10n	0,907	0,908	0,907	0,951	0,684
YOLOv10s	0,908	0,952	0,929	0,971	0,704
YOLOv10m	0,930	0,952	0,941	0,966	0,719
YOLOv11n	0,948	0,925	0,936	0,973	0,698
YOLOv11s	0,938	0,940	0,939	0,974	0,713
YOLOv11m	0,912	0,953	0,932	0,967	0,71
YOLOv12n	0,919	0,965	0,941	0,972	0,696
YOLOv12s	0,920	0,965	0,942	0,974	0,715
YOLOv12m	0,955	0,951	0,953	0,982	0,722

Source: (Research Results, 2025)

Among all the models evaluated, YOLOv12m achieved the highest detection performance, with an F1 score of 0.953 and an mAP@0.50 score of 0.982. This performance improvement can be attributed to the architectural enhancements introduced in YOLOv12, such as Residual Efficient Layer Aggregation Networks (RELAN) and Area Attention (A2), which improve feature aggregation and spatial focus in dense scene classes. The next best performing models are YOLOv8m (F1 = 0.944; mAP@0.50 = 0.974) and YOLOv10m (F1 = 0.941; mAP@0.50 = 0.966). The overall performance improvement trend across YOLO generations is shown in Figure 5, which illustrates a consistent increase in F1 scores and mAP@0.50:0.95 values from YOLOv8 to YOLOv12. Notably, YOLOv12m achieves the highest mAP@0.50:0.95 value of 72% along with an F1-Score of 95%, indicating that architectural refinements contribute not only to peak accuracy but also to balanced Precision–Recall performance across various behavior classes.



Source: (Research Results, 2025)

Figure 5. A comparison of F1-Score and mAP@0.50:0.95 across various versions of YOLO (v8–v12) with nano, small, and medium weight variants shows a consistent trend of increasing performance, with YOLOv12 m achieving the highest accuracy.

Although high accuracy values were observed, it is important to interpret these results in the context of a controlled experimental setting. All models were trained with early stopping and a predefined patience parameter, ensuring that training stopped once validation performance converged and preventing overfitting to the training data. The absence of divergence between training and validation performance during training indicates that the reported accuracy values reflect stable model convergence rather than uncontrolled overfitting. However, these results remain specific to the dataset's characteristics and should not be interpreted as a claim of universal generalization.

Inference Speed (RealTime) and Computational Efficiency Analysis

In addition to detection accuracy, inference speed is a critical factor in assessing the suitability of real-time monitoring systems. All 12 YOLO variants were evaluated using a 5 minute test video comprising 9,025 frames, processed in a uniform computing environment with NVIDIA T4 GPUs. Inference performance was measured based on the average Frames Per Second (FPS), inference time per frame (ms/frame), and model size (MB), as summarized in Table 5.

Table 5. YOLO Model Inference Speed Comparison Results

Model Version	Total Processing Time (s)	Average FPS	Inference Time (ms/frame)	Model Size (MB)
YOLOv8n	400.664	22.53	44.39	6
YOLOv8s	411.223	21.95	45.56	21,5
YOLOv8m	648.235	13.92	71.83	49,7
YOLOv10n	359.770	25.09	39.86	5,5
YOLOv10s	411.643	21.92	45.61	21,5
YOLOv10m	574.533	15.71	63.66	32
YOLOv11n	409.281	22.05	45.35	5,3
YOLOv11s	413.078	21.85	45.77	18,3
YOLOv11m	661.164	13.65	73.26	38,7
YOLOv12n	436.724	20.67	48.39	5,3
YOLOv12s	522.466	17.27	57.89	18,1
YOLOv12m	877.245	10.29	97.20	38,8

Source: (Research Results, 2025)

The results show a clear trade-off between detection accuracy and inference speed. Lightweight models (nano and small variants) achieve higher FPS, while medium-scale models offer better accuracy at the expense of slower inference speed. This inverse relationship is further illustrated in Figure 6. YOLOv10n and YOLOv8n achieve the highest FPS values, making them suitable for latency-sensitive applications, while YOLOv12m shows the lowest FPS despite having better detection accuracy. Although the FPS evaluation provides empirical evidence of differences in inference speed among YOLO variants, it does not explain the underlying architectural factors that contribute to these differences. Therefore, a model complexity analysis is presented to relate inference speed to architectural characteristics, including the number of parameters and computational cost (GFLOPS).

Table 6. Comparison of Model Complexity Among YOLO Variants

Model Version	Parameter	GFLOPS	Model Size (MB)
YOLOv8n	3,01	8,1	6
YOLOv8s	11,13	28,4	21,5
YOLOv8m	25,84	78,7	49,7
YOLOv10n	2,26	6,5	5,5
YOLOv10s	7,21	21,4	21,5
YOLOv10m	15,31	58,9	32
YOLOv11n	2,58	6,3	5,3
YOLOv11s	9,41	21,3	18,3
YOLOv11m	20,03	67,7	38,7
YOLOv12n	2,55	6,3	5,3
YOLOv12s	9,23	21,2	18,1
YOLOv12m	20,10	67,1	38,8

Source: (Research Results, 2025)

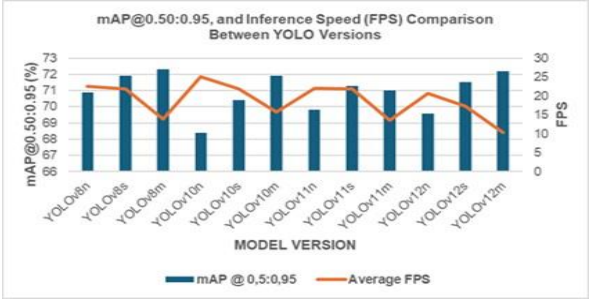
This table compares model complexity across all YOLO variants. The results show that the observed decrease in inference speed, especially in the YOLOv12 model, is not due to implementation inefficiencies or software limitations, but rather a direct consequence of increased architectural complexity. Advanced components such as Area Attention and Residual Efficient Layer Aggregation Networks (RELAN) introduce additional computational operations that improve feature representation and spatial awareness, especially in

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

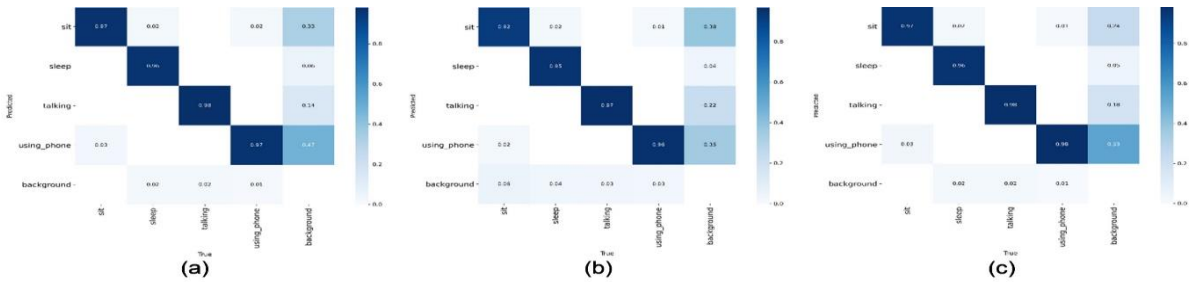
dense, small-object scenarios. In addition, the increase in network depth and width in medium-scale models results in more parameters and a significant increase in GFLOPS, directly affecting the processing time per frame. This trend is clearly reflected in YOLOv12m, which exhibits much higher computational costs compared to lightweight variants, explaining its lower FPS despite superior detection accuracy. These findings confirm that the trade-off between accuracy and inference speed in newer generations of YOLO is an inherent architectural consequence, not a performance anomaly. This inverse relationship is clearly shown in

Figure 6. YOLOv12m achieves the highest accuracy but has the lowest FPS, while YOLOv10n and YOLOv8n achieve the highest FPS with slightly lower accuracy. These results confirm that model selection must be tailored to the application's specific requirements, whether prioritizing real-time efficiency or maximizing detection accuracy. The observed decrease in inference speed in the YOLOv12 model can be attributed to the architecture's increased complexity. The integration of the Area Attention and RELAN mechanisms introduces additional computational operations and parameter interactions, thereby improving feature representation but also increasing the processing load. As a result, the YOLOv12 model requires more computational resources per frame, resulting in lower FPS than simpler architectures such as YOLOv10, which uses a more concise, NMS free design. These findings suggest that the lower FPS in YOLOv12 is a direct consequence of its architectural emphasis on accuracy, rather than implementation limitations.



Source: (Research Results, 2025)

Figure 6. A comparison of mAP@0.50:0.95 and FPS across various YOLO versions (v8–v12) with nano, small, and medium variants highlights the trade-off between detection accuracy and inference speed.



Source: (Research Results, 2025)

Figure 7. Comparison of Confusion Matrix (Normalized) for Error Pattern Analysis in Representative Models: (a) Highest Accuracy (YOLOv12m), (b) Highest FPS (YOLOv10n), and (c) Optimal Recommendation (YOLOv8s).

Figure 7 compares the confusion matrices of three representative models, selected to reflect distinct performance characteristics: YOLOv12m, which achieves the highest detection accuracy; YOLOv10n, which is the fastest in terms of inference speed; and YOLOv8s, which offers the optimal trade-off between accuracy and speed for practical deployment.

1. YOLOv12m Highest Accuracy (Figure 7a), the prediction distribution is strongly centered on the main diagonal, indicating a very low misclassification rate. This aligns with the high, consistent F1-Score values per class (0.934–0.966), confirming YOLOv12m's robustness in distinguishing behavior categories despite variations in pose and background.
2. YOLOv10n Highest Inference Speed (Figure 7b), YOLOv10n shows greater error dispersion, especially in the Using Mobile Phone class, reflected in a decrease in F1-Score due to low precision.
3. YOLOv8s Optimal Accuracy-Speed Balance (Figure 7c), the confusion matrix of YOLOv8s is cleaner than that of YOLOv10n and only slightly below that of YOLOv12m. The relatively high and stable F1-Score (0.906–0.971) indicates a good balance between accuracy and efficiency, making it the most practical choice for real-time class monitoring.

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

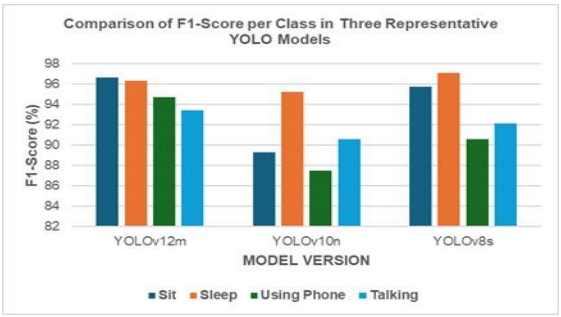
Qualitative Error Analysis and Robustness

The analysis in the previous section shows that all models achieve competitive accuracy and inference speed. However, to understand the sources of performance differences in greater depth, a qualitative evaluation of error patterns and model robustness in distinguishing between the four behavior categories is required. Therefore, this analysis focuses on two main aspects: (1) misclassification patterns between classes through a normalized confusion matrix, and (2) evaluation of model robustness based on per-class performance. This pattern shows the consequences of a lightweight architecture that prioritizes inference speed over detection accuracy.

Table 7. Comparison of Performance per Class on Three Representative YOLO Models (YOLOv12m, YOLOv10n, YOLOv8s)

Model Version / Class Categories	YOLOv12m			YOLOv10n			YOLOv8s		
	Precision	Recall	F1-Score	Precision	Recall	F1-Score	Precision	Recall	F1-Score
Sit	0,966	0,966	0,966	0,933	0,856	0,893	0,970	0,944	0,957
Sleep	0,964	0,963	0,963	0,958	0,946	0,952	0,961	0,982	0,971
Using Phone	0,967	0,927	0,947	0,833	0,922	0,875	0,890	0,922	0,906
Talking	0,922	0,947	0,934	0,904	0,909	0,906	0,880	0,966	0,921

Source: (Research Results, 2025)



Source: (Research Results, 2025)

Figure 8. Comparison of F1-score per class in three representative YOLO models (YOLOv12m, YOLOv10n, and YOLOv8s).

Figure 8 presents a per-class comparison of F1-Scores across three representative YOLO models YOLOv12m, YOLOv10n, and YOLOv8s to highlight differences in detection reliability across student behavior categories. This visualization complements the confusion matrix analysis by providing a class-level perspective on how architectural complexity and model capacity influence performance consistency across distinct behaviors.

Source: (Research Results, 2025)

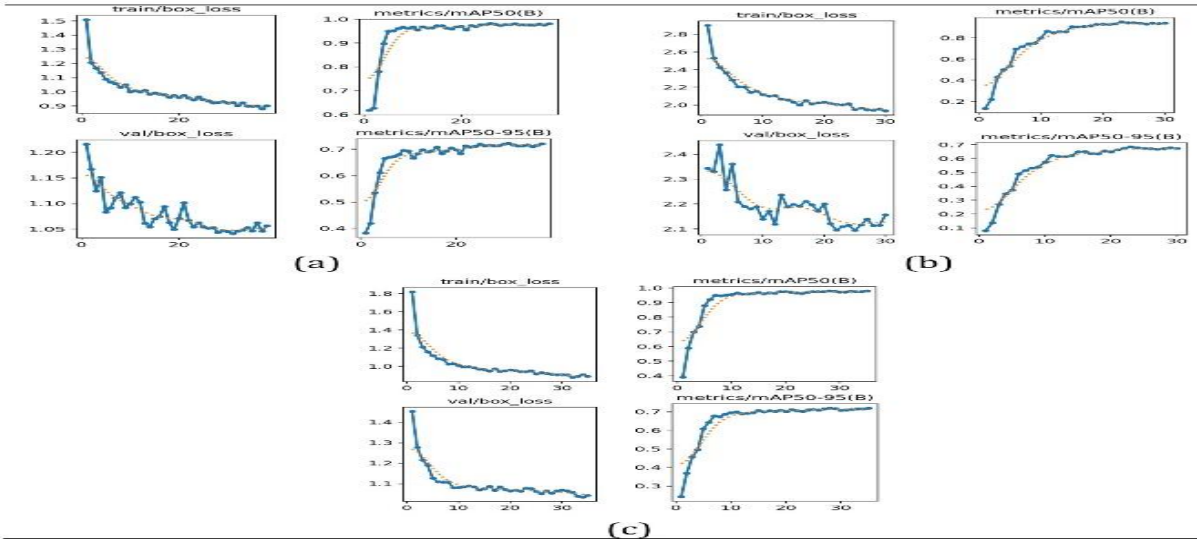
Figure 9. Qualitative visualization of student behavior detection results across five classroom scenarios using three representative YOLO models: YOLOv12m (medium), YOLOv8s (small), and YOLOv10n (nano).

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

The figure illustrates consistent behavior recognition without fatal misclassification, with observable differences mainly in detection sensitivity and bounding box coverage rather than class prediction accuracy The results indicate that YOLOv12m achieves the highest and most consistent F1-Score across all classes, reflecting its superior feature representation capability derived from a more complex architecture. In contrast, YOLOv10n shows a noticeable performance decline, particularly in the "Using a Phone" and "Sitting" classes, which aligns with its higher false-positive and false-negative rates observed in the normalized confusion matrices and reflects the trade-off associated with its lightweight, speed oriented design. Meanwhile, YOLOv8s demonstrates stable, competitive performance, with F1-Score values that closely match those of YOLOv12m across most behavior categories despite its smaller model size. This pattern underscores the impact of architectural depth and computational capacity on class-wise detection reliability. It supports selecting YOLOv8s as a balanced option when both accuracy and efficiency are required. As a complement to quantitative analysis and confusion-matrix-based error evaluation, the visualization of detection results in Figure 9 provides a qualitative overview of model behavior in real-world classroom environments. This figure compares the detection results of three representative YOLO models YOLOv12m (highest accuracy), YOLOv10n (highest inference speed), and YOLOv8s (optimal balance between accuracy and speed) across five common classroom behavioral scenarios.

By directly visualizing bounding boxes, class labels, and confidence scores, this qualitative assessment enables an intuitive evaluation of detection consistency, sensitivity, and model stability in handling variations in student position, class density, and simultaneous behaviors. In detail, the visual results show that all three representative models consistently recognize student behavior without fatal classification errors across all tested scenarios. The differences between models are mainly seen in the detection sensitivity and the number of bounding boxes generated, not in the accuracy of class recognition. In the sitting scenario, YOLOv12m showed a wider detection range, identifying objects farther from the camera, including students at the back of the classroom. At the same time, YOLOv8s and YOLOv10n tended to focus on more dominant objects in the front of the classroom. In the sleeping, phone use, and talking scenarios, all three models produced consistent and identical detection patterns, indicating that the main visual features representing body gestures and small objects can be captured stably by all architectures. In the multi-class scenario, which represents the highest level of complexity, all models successfully detected combinations of behaviors simultaneously without significant prediction overlap, with differences appearing only in confidence scores. These findings reinforce previous evaluation results that YOLOv12m offers the richest feature representation, YOLOv10n excels in inference efficiency, and YOLOv8s maintains the most suitable accuracy-speed balance for real-time class monitoring implementation.



Source: (Research Results, 2025)

Figure 10. Training–validation loss convergence and mAP progression of representative YOLO models: (a) YOLOv12m, (b) YOLOv10n, and (c) YOLOv8s, demonstrating stable learning behavior and the absence of divergence between training and validation curves.

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

JITK (JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI KOMPUTER)

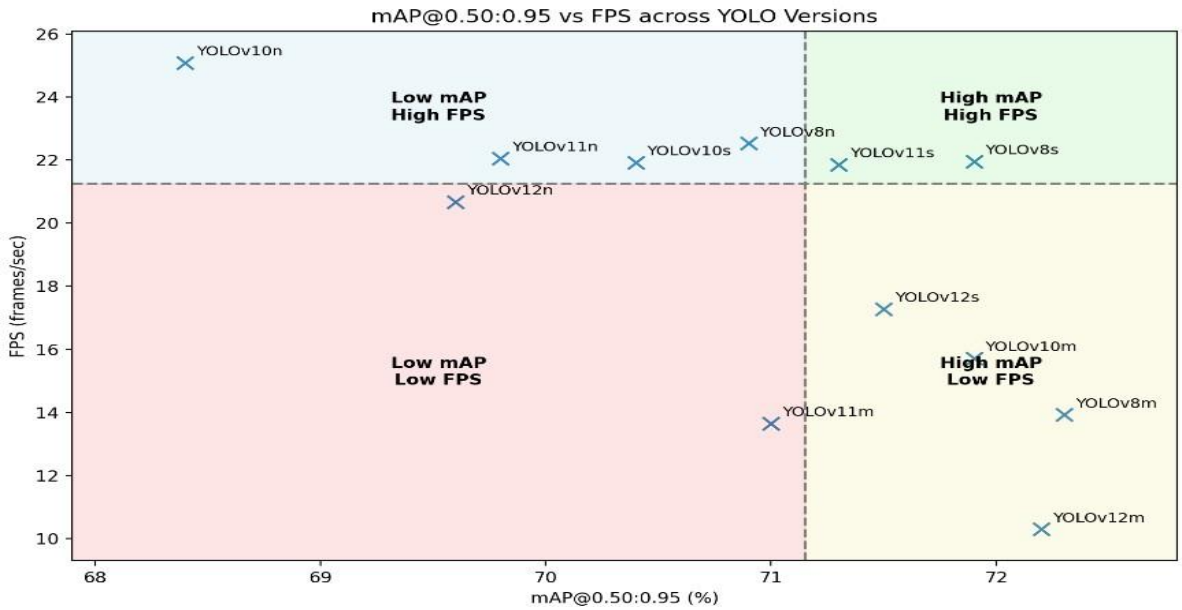
Analysis of Training-Validating Convergence

In this section, Figure 10 presents a convergence analysis of training and validation on three representative YOLO models (a) YOLOv12m, (b) YOLOv10n, and (c) YOLOv8s which were selected to represent a spectrum of performance from highest accuracy, highest inference speed, to a balance of accuracy and speed. This visualization aims to verify that the high accuracy values obtained in previous evaluations were not due to overfitting, but rather to a stable, controlled learning process on the dataset used.

In detail, all models exhibit a healthy convergence pattern, characterized by a consistent decrease in both train and validation box losses during the early training phase, followed by stabilization without significant divergence in subsequent epochs. The parallelism between the training and validation curves indicates adequate generalization, despite the dataset's relatively limited size.

Trade-off Discussion and Optimal Model Recommendations

Figure 11 presents a trade-off analysis between detection accuracy and inference speed across all evaluated YOLO variants, using a quadrant map based on mAP@0.50:0.95 (71.1%) and Frames Per Second (21.3). Using the median value as the dividing line, this visualization groups models into four performance zones ranging from high-speed, low-accuracy configurations to high accuracy, lower-inference-speed configurations. This approach allows for more intuitive identification of each model's relative position in the performance spectrum, while providing a clear basis for formulating optimal model recommendations according to application needs, whether prioritizing real-time efficiency, maximum detection accuracy, or a balance of both in the context of behavior monitoring in a classroom environment.



Source: (Research Results, 2025)

Figure 11. Quadrant analysis of the relationship between mAP@0.50:0.95 and FPS across all YOLO variants (v8–v12)

1. Upper-Left Quadrant (Low mAP, High FPS): This quadrant contains models that prioritize time efficiency, such as YOLOv10n, YOLOv8n, YOLOv11n, and YOLOv10s. With the highest FPS (up to 25.09), this variant is ideal for edge devices (e.g., Raspberry Pi, Jetson Nano), though its accuracy is relatively low (68–71%).
2. Upper-Right Quadrant (High mAP, High FPS): This is the optimal zone and is occupied by YOLOv8s (71.9%; 21.95 FPS) and YOLOv11s (71.3%; 21.85 FPS). Both models maintain the best balance between accuracy and speed; these models are therefore recommended as the most practical choice for real-time smart classroom monitoring under controlled conditions.
3. Lower-Right Quadrant (High mAP, Low FPS): This quadrant contains models with the highest accuracy but low speed, such as YOLOv8m, YOLOv10m, YOLOv12m, and YOLOv12s. These models are suitable for server-based implementations or non-real-time scenarios that require maximum detection precision.
4. Lower Left Quadrant (Low mAP, Low FPS): This quadrant is populated by models such as

Accredited Rank 2 (Sinta 2) based on the Decree of the Dirjen Penguatan RisBang Kemenristekdikti No.225/E/KPT/2022, December 07, 2022. Published by LPPM Universitas Nusa Mandiri

VOL. 11. NO. 4 MAY 2026
P-ISSN: 2685-8223 | E-ISSN: 2527-4864
DOI: 10.33480/jitk.v11i4.7773

YOLOv12n (69.6%; 20.67 FPS) and YOLOv11m (71.0%; 13.65 FPS), which exhibit lower performance on both metrics.

Overall, these results confirm the inherent trade-off between detection accuracy and computational efficiency in the YOLO architecture. Model selection should be based on specific application requirements, carefully considering environmental constraints and computational resources.

CONCLUSIONS

This study presents a controlled comparative evaluation between YOLO generations (YOLOv8, YOLOv10, YOLOv11, and YOLOv12) at three weight scales (nano, small, and medium) for detecting student behavior in a classroom environment. It shows a consistent trade-off between detection accuracy and inference speed under the dataset and experimental conditions used. YOLOv12m achieves the highest accuracy (mAP@0.50:0.95≈0.722; F1-Score = 0.953) with the consequence of lower inference speed, while YOLOv10n offers the highest speed (25.09 FPS) but shows a greater risk of reliability in subtle visual behaviors, such as cell phone use. Within the scope of this study, YOLOv8s (21.95 FPS; mAP@0.50:0.95=0.719) and YOLOv11s (21.85 FPS; mAP@0.50:0.95=0.713) are recommended as the most balanced configurations for controlled-class environments with a single frontal camera, although this recommendation is not intended as a universal generalization claim. From an implementation perspective, high-speed models such as YOLOv10n require additional measures to mitigate false positives, including confidence threshold settings, temporal processing, and human verification, especially in sensitive scenarios. At the same time, ethical considerations demand that behavior detection systems be used as decision-support tools rather than automatic enforcement mechanisms. Moving forward, expanding and balancing datasets, integrating multi-camera and temporal tracking, optimizing models for edge devices, and testing under more diverse class conditions are important steps to improve the reliability and readiness of systems for real-world implementation.

REFERENCE

[1] H. Chen and J. Guan, "Teacher–Student Behavior Recognition in Classroom Teaching Based on Improved YOLOv4 and Internet of Things Technology," *Electronics (Switzerland)*, vol. 11, no. 23, Dec. 2022, doi: 10.3390/electronics11233998.

[2] S. J. Russell and P. Norvig, Artificial Intelligence: A Modern Approach, 4th ed. Noida, India: Pearson India Education Services Pvt. Ltd., 2022.

[3] R. Szeliski, "Computer Vision: Algorithms and Applications 2nd Edition," 2021. [Online]. Available: https://szeliski.org/Book,

[4] N. Tran, H. Nguyen, H. Luong, M. Nguyen, K. Luong, and H. Tran, "Recognition of student behavior through actions in the classroom," IAENG Int. J. Comput. Sci., vol. 50, no. 3, pp. 1031–1041, 2023. [Online]. Available: https://www.iaeng.org/IJCS/issues_v50/issue_3/IJCS_50_3_26.pdf.

[5] P. D. Nguyen *et al.*, "A new dataset and systematic evaluation of deep learning models for student activity recognition from classroom videos," in *2022 International Conference on Multimedia Analysis and Pattern Recognition, MAPR 2022 Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2022. doi: 10.1109/MAPR56351.2022.9924673.

[6] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. Cambridge, MA, USA: MIT Press, 2016.

[7] A. Deshpande and K. Warhade, "SADY: Student Activity Detection Using YOLObased Deep Learning Approach," vol. 13, no. 4, 2023, doi: 10.18517/ijaseit.13.4.18393.

[8] H. Das, H. K. Hira, M. Uddin, A. K. Roy, and A. Mahmud, "A Hybrid YOLOBased Approach for FineGrained Detection of Classroom Student Behaviors," in *2024 27th International Conference on Computer and Information Technology, ICCIT 2024 Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 2928–2933. doi: 10.1109/ICCIT64611.2024.11022537.

[9] W. Cao, P. Lu, and W. Cao, "Multimodal Gesture Recognition with SpatioTemporal Features Fusion Based on YOLOv5 and MediaPipe," *Intern J Pattern Recognit Artif Intell*, vol. 38, no. 8, Jun. 2024, doi: 10.1142/S0218001424550073.

[10] Q. Jia and J. He, "Student Behavior Recognition in Classroom Based on Deep Learning," *Applied Sciences (Switzerland)*, vol. 14, no. 17, Sep. 2024, doi: 10.3390/app14177981.

[11] S. Yuan, X. Kong, and S. Zhang, "Research on Enhanced YOLOv8 Gesture Recognition Method for Complex Environments," in

Proceeding of the WRC Symposium on Advanced Robotics and Automation, WRC SARA, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 141–146. doi: 10.1109/WRCSARA64167.2024.10685785.

[12] L. Han, X. Ma, M. Dai, and L. Bai, "A WADYOLOv8based method for classroom student behavior detection," *Sci Rep*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s4159802587661w.

[13] L. Zhou, X. Liu, X. Guan, and Y. Cheng, "CSSAYOLO: CrossScale Spatiotemporal Attention Network for FineGrained Behavior Recognition in Classroom Environments," *Sensors*, vol. 25, no. 10, May 2025, doi: 10.3390/s25103132.

[14] J. Širmenis, "Research on Techniques for Automatic Recognition and Tracking of Basketball Shots from Video," Master's thesis, Kaunas University of Technology, 2025.

[15] B. Qin, H. Hu, and S. Du, "ACMYOLOv10: Research on Classroom Learning Behavior Recognition Algorithm Based on Improved YOLOv10," *IEEE Access*, vol. 13, pp. 144863–144877, 2025, doi: 10.1109/ACCESS.2025.3599686.

[16] M. Rashid, J. Wang, S. Ahmed, and F. Ahmed, "Survey on DLBased Object Detection and Pose Estimation for HumanRobot Collaboration Manufacturing," Jun. 2025. [Online]. Available: https://ssrn.com/abstract=5286783

[17] N. Jegham, C. Y. Koh, M. Abdelatti, and A. Hendawi, "YOLO Evolution: A Comprehensive Benchmark and Architectural Review of YOLOv12, YOLO11, and Their Previous Versions," Mar. 2025, [Online]. Available: http://arxiv.org/abs/2411.00201

[18] Z. Sun and V. Y. Mariano, "SiTYOLOv9: An Efficient Algorithm for Learning Behavior Detection in the Home Environment," *Journal of Computational and Cognitive Engineering*, vol. 4, no. 2, pp. 173–185, May 2025, doi: 10.47852/bonviewJCCE42023949.

[19] C.-Y. Wang, I.-H. Yeh, and H.-Y. M. Liao, "YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information," in Computer Vision – ECCV 2024, A. Leonardis, A. Ricci, E. Roth, S. Russakovsky, J. Sattler, and G. Varol, Eds., Lecture Notes in Computer Science, vol. 15089. Cham, Switzerland: Springer, 2025, doi: 10.1007/978-3-031-72751-1_1.

[20] E. Kim, "YOLOv8 Nano vs YOLOv8 Large," Medium, Aug. 2023. [Online]. Available: https://medium.com/@elvenkim1/yolov8nanovsyolov8large4f21324baa38. [Accessed: Oct. 29, 2025].

[21] M. Yaseen, "What is YOLOv9: An In-Depth Exploration of the Internal Features of the Next-Generation Object Detector," arXiv preprint arXiv:2409.07813, 2024, doi: 10.48550/arXiv.2409.07813.

[22] A. Imran, M. S. Hulikal, and H. A. A. Gardi, "Real-time American Sign Language Detection Using Yolo-v9," arXiv preprint arXiv:2407.17950, 2024, doi: 10.48550/arXiv.2407.17950.

[23] J. Terven and D. CordovaEsparza, "A Comprehensive Review of YOLO Architectures in Computer Vision: From YOLOv1 to YOLOv8 and YOLONAS," Feb. 2024, doi: 10.3390/make5040083.

[24] V. H. Le, "Selected hand gesture recognition model based on crossevaluation of deep learning from large RGB image datasets," *Multimed Tools Appl*, vol. 84, no. 32, pp. 40009–40058, Sep. 2025, doi: 10.1007/s1104202520743z.