# COUNTING AND TRACKING OBJECTS FOR CLASSROOM MANAGEMENT AT DONG NAI TECHNOLOGY UNIVERSITY

Phuc Thinh Do*, Quoc Ky Hoang, Ngoc Tien Bui

*Dong Nai Technology University*
*Corresponding author: *Phuc Thinh Do, dophucthinh@dntu.edu.vn*

**GENERAL INFORMATION**

Received date: 27/03/2024

Revised date: 05/05/2024

Accepted date: 11/07/2024

**KEYWORD**

*Object detection;*
*Object tracking;*
*Object counting;*

**ABSTRACT**

In the context of advancing technology and increasing demand for efficient management, the application of an automated system capable of counting and tracking objects has become indispensable for improving the management process. This task imposes high requirements on processing time and accuracy. The system size and deployment capabilities also require special attention, particularly in managing classrooms at universities. In this study, we propose a system for counting the number of students and tracking their entry and exit in classrooms at Dong Nai Technology University. The system will provide the current number of students in the class, the number of students entering and leaving the class, and the current status of the lecturer. Additionally, we have built the dataset and selected object recognition methods to ensure that the system can deploy operations in real-time. Experimental results show that the system achieves significant accuracy and operational speed when used in classroom monitoring.

## 1. INTRODUCTION

Classroom management plays a crucial role in the teaching and learning process within the university education environment. At Dong Nai Technology University, with its large and diverse student population, classroom management presents numerous challenges. Traditionally, classroom management processes have relied on manual methods, where instructors manually count the number of students present and take notes, which are then reported to the monitoring unit regarding attendance. This manual approach can lead to wasted time and effort for both instructors and students, as well as challenges in providing accurate information about the current status of the classroom. Additionally, Dong Nai Technology University encourages instructors to maintain comprehensive oversight of the entire classroom during teaching sessions to prevent students from engaging in unrelated activities. To achieve this, an effective classroom monitoring and management system is necessary. The idea of this system involves utilizing classroom surveillance cameras to monitor and count the number of students entering and exiting the room. To implement this idea, we employ object detection and tracking algorithms tailored to our specific use case, where the objects of interest are students and instructors.

With the development of artificial intelligence in general and computer vision in particular, many image-processing methods have been proposed and applied in various practical scenarios. These

methods can be utilized to address the issues of object counting and tracking in classrooms. Some object detection methods, such as YOLO (You Only Look Once): YOLOv1, YOLOv2, YOLOv3 (*Redmon et al.*)..., and more recently developed versions like YOLOv8 (*Glenn et al.),* YOLOv9 (*Wang et al.),* achieve fast object detection with high accuracy, enabling real-time applications. However, models in the YOLO series often require an additional stage to eliminate redundant bounding boxes using the non-maximum suppression (NMS) algorithm (Figure 1), which increases computational costs.

Recently, with the emergence of Transformers (*Vaswani et al.*) in the field of natural language processing, many methods in the field of computer vision have also applied Transformers to their models, especially the DETR (Detection Transformer) model (Carion et al.) for object detection. This model eliminates the need for applying NMS during object detection. From there, many improved models have also been developed such as Deformable-DETR (*Zhu et al.*), DINO (*Zhang et al.*), and especially recently RT-DETR (*Lv et al.*). This is an end-to-end improved model capable of running in real-time. In this paper, we choose the RT-DETR model to apply to our system because of its speed and accuracy when deployed in real-world scenarios. Additionally, we use the Ultralytics library to track objects with RT-DETR more easily.

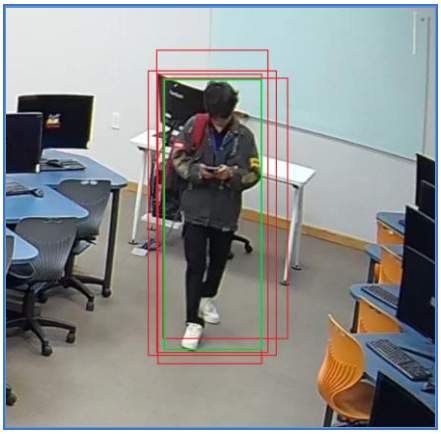

**Figure 1.** Multiple bounding boxes define the same object. The red boxes are predicted boxes, and the green boxes are ground truth.

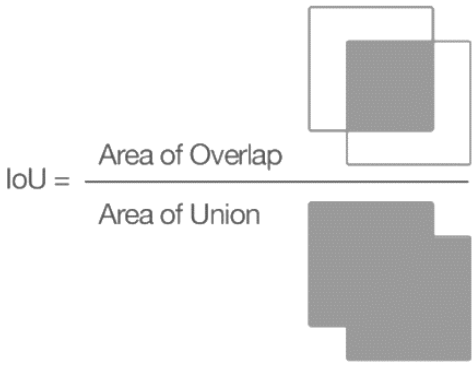


**Figure 2.** Describing how to calculate the IoU score.

## 2. RELATED WORK

The current object detection methods can be divided into two-stage, single-stage, and end-to-end models.

### 2.1. Two-stage model

Representative models in this category include the famous trio R-CNN (*Girshick et al.*), Fast R-CNN (*Girshick et al.*), Faster R-CNN (*Ren et al.*). These models operate in two stages: the first stage involves the use of a Region Proposal Network (RPN) to identify regions in the image that may contain objects, while the second stage predicts the objects within these identified regions. While these methods offer high accuracy, they are relatively slow and challenging to execute in real-time applications.

### 2.2. Single-stage model

Methods in this category include SSD (*Liu et al.*), and the YOLO series. These models eliminate the need for a Region Proposal Network (RPN), resulting in a single-stage architecture that enables faster processing and real-time applications. However, to obtain final detection results, these models must apply NMS to remove redundant bounding boxes with Intersection over Union (IoU) (Figure 2) values below a selected threshold. This additional step introduces computational costs, and the accuracy of predictions can be affected by the threshold selection.

### 2.3. End-to-end model

The integration of Transformer architecture into the object detection problem has shown promising results, with models like DETR eliminating the need for NMS and offering real-time object detection capabilities. By leveraging attention mechanisms, prediction heads in these models can effectively avoid duplicative predictions on the same object. Building upon DETR, *Zhu et al.* introduced the Deformable-DETR model, which enhances training convergence speed. Similarly, the DINO model by *Zhang et al.* utilizes contrastive denoising to refine anchor selection, leading to stable training and fast convergence. More recently, a research team at Baidu proposed the RT-DETR model, which enhances object query selection using IoU-aware queries, enabling the model to prioritize objects with both high classification scores and high IoU values.

## 3. PROPOSED METHOD

The proposed system is an end-to-end model, which simplifies the training process. Our model leverages the RT-DETR model for object detection due to its precise detection capabilities and real-time execution. Additionally, we integrate the Ultralytics tool to enhance object tracking. This library supports object recognition across various versions of the YOLO series, allowing us to conveniently compare our current approach with YOLO.

### 3.1. Object detection model

The RT-DETR model, developed by the engineering team at Baidu, has been shown to outperform YOLOv8 regarding both speed and accuracy. According to their publication, RT-DETR achieves a faster frame rate (74fps compared to YOLOv8's 50fps when using GPU T4) and higher accuracy (54.8% Average Precision compared to YOLOv8's 53.9% Average Precision when evaluated on the COCO val 2017 dataset). Therefore, we have chosen the RT-DETR model as the backbone for detecting students in the classroom.

**Figure 3.** The view of the surveillance camera in the classroom.

### 3.2. Object counting method

After obtaining the object detection model, the next step is to extract camera footage to analyze the classroom space. Figure 3 displays the view captured by the surveillance camera in the classroom. To count the number of students in the classroom, we define a polygonal counting area within the classroom (Figure 4) to track and count objects within this region. The counting area is defined by vertices, with each vertex represented by a pair of coordinates (x, y) arranged clockwise. Due to the angle of the surveillance camera, we opted for a pentagonal counting area in our experiment. For example, a list of coordinates for the vertices of a pentagon could be [(100, 600), (1712, 160), (2700, 220), (2200, 1450), (400, 1450)].

Once the counting area is established, counting students in the classroom becomes straightforward: the number of students equals the number of detected objects within the counting area minus one. Moreover, to avoid the lecturer sitting in one place while teaching, we also added a rectangular counting area at the lecturer's seat.

**Figure 4.** Description of counting areas. The blue line represents the polygonal counting area for counting students, while the yellow line represents the rectangular area designated for the instructor.

### 3.3. The object tracking method

To track the number of students entering and exiting the classroom, we define a monitoring area as a quadrilateral frame (Figure 5) to observe student movement. When a student passes through the left side of the frame, it is counted as entering the classroom, whereas passing through the right side is counted as exiting. Alternatively, the approach can be adapted to use the top or bottom sides of the frame. However, this method faces challenges in practice due to glass classroom walls allowing the camera to observe both inside and outside. Additionally, determining if a bounding box enters the monitoring area is complex, leading to miscounting when students from adjacent classrooms pass through the designated area.

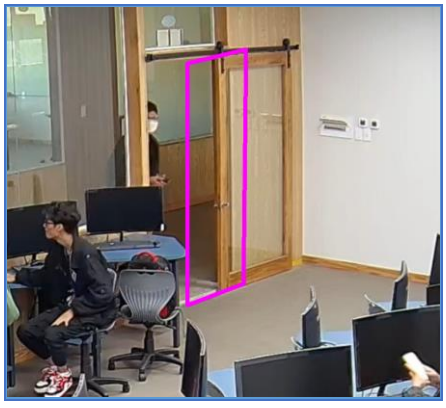

**Figure 5.** The counting area for students entering and leaving the classroom, on the right side, is positioned inside the door frame to avoid mistakenly counting students walking outside the hallway.


**Figure 6.** Counting the number of students entering and exiting the classroom. The red curve indicates the tracked paths of the students.

To address this issue, we track students using the centroid of the bounding box detected by the model. Instead of evaluating the IoU between the bounding box and the quadrilateral frame, we monitor whether the centroid coordinates of the student's bounding box move into the monitoring area (Figure 6).

### 3.4. Classroom monitoring system

We utilized the Ultralytics library to implement our approach, which supports various object detection and tracking methods. This includes functionalities for model training, fine-tuning, and access to pre-trained weights of models trained on standard datasets such as COCO 2017 - an extension of MS COCO (*Lin et al.*)). As a result, these models have acquired a specific understanding of human features, allowing them to leverage this knowledge when transitioning to other features, such as students. Ultralytics also supports models from the YOLO series, facilitating easier comparisons between different models.

## 4. EXPERIMENTS

### 4.1. Dataset and setups

To acquire training data for the model, we captured videos from surveillance cameras in classrooms at Dong Nai Technology University. Frames were randomly extracted from these videos and manually annotated. In total, we collected 300 images with 9500 annotations. Although the dataset size is limited due to the manual collection process, we proceeded to fine-tune the model on this dataset without retraining it from scratch. We divided the dataset into three parts for evaluation: training, validation, and testing. The dataset was divided as follows: 250 images for training, 50 images for validation, and 100 images for testing.

The machine configuration used for training included an i7-7820HK@2.90GHz CPU, GTX 1080 GPU, and 64 GB RAM. We kept the training model settings default as per the Ultralytics guidelines. We chose the RTDETR-X model with 67 million parameters to initialize the model weights. Additionally, since the weight set of the YOLOv9 model was not available at the time of the experiment, we also experimented with the YOLOv8-X weight set, which has 68 million parameters, for comparison with RTDETR-X.

### 4.2. Experiment Results

After the training process, we evaluated the performance of the model using three metrics: Precision, Recall, and F1-Score, calculated based on True Positive (TP), True Negative (TN), False Positive (FP), and False Negative (FN) according to formulas (1), (2), and (3). To determine whether a sample is Positive or Negative, we used the IoU score. A sample is considered Positive if the IoU ratio is greater than 0.5; otherwise, it is considered Negative.

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

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

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

We constructed a Confusion Matrix **(Table 1)** to illustrate the experimental results further. **Table 2** shows a relatively high Precision score, indicating accurate detection of students due to the model's understanding of human characteristics. However, the Recall score is low due to the limited training data size, which is insufficient for the model to detect all students in the classroom. **Table 2** also illustrates the comparison results between RT-DETR and YOLOv8 when applied to the system. Both models achieve high Precision scores, but RT-DETR slightly outperforms YOLOv8 regarding Recall scores despite YOLOv8 having more parameters than RT-DETR.

**Table 1.** Confusion matrix for the experimental results.

| | Actual: Person | Actual: Not Person |
|---|---|---|
| Predicted: Person | TP = 1736 | FP = 6 |
| Predicted: Not Person | FN = 513 | TN |

**Table 2.** Evaluation results on the metrics.

| **Metric** | **RT-DETR** | **YOLOv8** |
|---|---|---|
| Recall | 0.77 | 0.75 |
| Precision | 0.99 | 0.99 |
| F1-Score | 0.87 | 0.85 |

When applied for classroom monitoring, the model performs well under low-light conditions. **Figures 7 and 8** demonstrate the superior detection capability of the RT-DETR model

compared to YOLOv8, especially for small-sized objects. However, the model yields unsatisfactory results in some frames (Figure 9). This is because students sitting in obscured or distant positions from the camera result in smaller display sizes, affecting the model's performance.

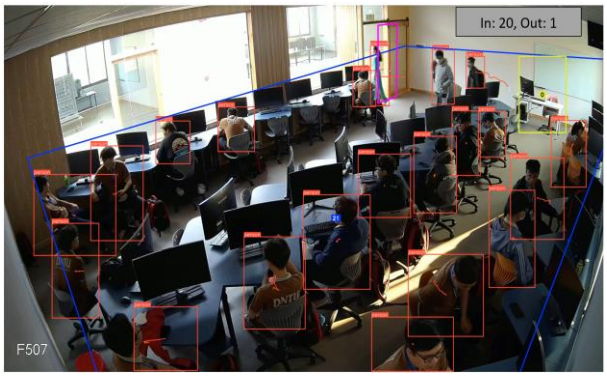


**Figure 7.** The counting results of the system using the RT-DETR model under low-light conditions.

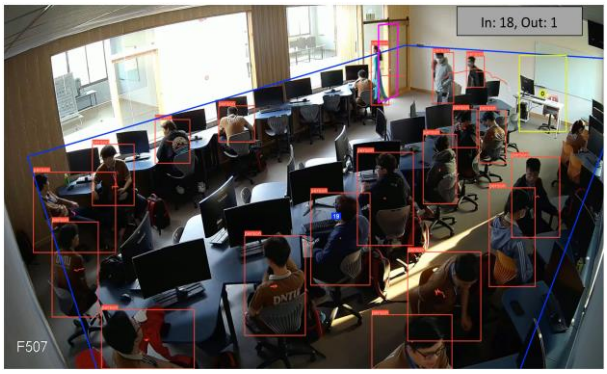


**Figure 8.** The counting results of the system using the YOLOv8 model under low-light conditions.


**Figure 9.** The results are not good in some frames when students are sitting far away from the camera.

## 5. CONCLUSION

In this paper, we applied an object detection model to count and track student activities in the classroom at Dong Nai Technology University. We constructed labeled data from classroom surveillance videos. The experiments demonstrate that the model can count and monitor the learning process of students and instructors with an acceptable level of accuracy. Additionally, we compared existing object detection models when applied to this problem. In future research, we aim to further develop the ability to detect abnormal behaviors in the classroom and apply lighter models suitable for use on low-configured devices.

## ACKNOWLEDGEMENT

We would like to express our sincere gratitude to the Department of Facilities Management and Dong Nai Technology University for providing the surveillance videos for conducting this research.

## REFERENCES

Carion, N., Massa, F., Synnaeve, G., Usunier, N., Kirillov, A., & Zagoruyko, S. (2020, August). End-to-end object detection with transformers. In European conference on computer vision (pp. 213-229). Cham: Springer International Publishing.

Girshick, R., Donahue, J., Darrell, T., & Malik, J. (2014). Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 580-587).

Girshick, R. (2015). Fast r-cnn. In Proceedings of the IEEE international conference on computer vision (pp. 1440-1448).

Glenn, J. (2024). YOLOv8 release v8.1.0. https://github.com/ultralytics/ultralytics/releases/tag/v8.1.0.

Lin, T. Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., ... & Zitnick, C. L. (2014). Microsoft coco: Common objects in context. In Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13 (pp. 740-755). Springer International Publishing.

Liu, W., Anguelov, D., Erhan, D., Szegedy, C., Reed, S., Fu, C. Y., & Berg, A. C. (2016). Ssd: Single shot multibox detector. In Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part I 14 (pp. 21-37). Springer International Publishing.

Lv, W., Xu, S., Zhao, Y., Wang, G., Wei, J., Cui, C., ... & Liu, Y. (2023). Detrs beat yolos on real-time object detection. arXiv preprint arXiv:2304.08069.

Redmon, J., & Farhadi, A. (2017). YOLO9000: better, faster, stronger. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 7263-7271).

Redmon, J., & Farhadi, A. (2018). Yolov3: An incremental improvement. arXiv preprint arXiv:1804.02767.

Redmon, J., Divvala, S., Girshick, R., & Farhadi, A. (2016). You only look once: Unified, real-time object detection. In Proceedings of the IEEE conference on computer vision and pattern recognition (pp. 779-788).

Ren, S., He, K., Girshick, R., & Sun, J. (2015). Faster r-cnn: Towards real-time object detection with region proposal networks. Advances in neural information processing systems, 28.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., ... & Polosukhin, I. (2017). Attention is all you need. Advances in neural information processing systems, 30.

Wang, C. Y., Yeh, I. H., & Liao, H. Y. M. (2024). YOLOv9: Learning What You Want to Learn Using Programmable Gradient Information. arXiv preprint arXiv:2402.13616.

Zhang, H., Li, F., Liu, S., Zhang, L., Su, H., Zhu, J., ... & Shum, H. Y. (2022). Dino: Detr with improved denoising anchor boxes for end-to-end object detection. arXiv preprint arXiv:2203.03605.

Zhu, X., Su, W., Lu, L., Li, B., Wang, X., & Dai, J. (2020). Deformable detr: Deformable transformers for end-to-end object detection. arXiv preprint arXiv:2010.04159.