Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2025.062686

ARTICLE

Robust Alzheimer's Patient Detection and Tracking for Room Entry Monitoring Using YOLOv8 and Cross Product Analysis

Praveen Kumar Sekharamantry[1,2,*], Farid Melgani[1], Roberto Delfiore[3] and Stefano Lusardi[3]

[1]Department of Information Engineering and Computer Science, University of Trento, Trento, 38123, Italy
[2]Department of Computer Science and Engineering, GITAM School of Technology, GITAM (Deemed to be University), Visakhapatnam, 530045, India
[3]TeiaCare S.r.l, Milan, 20127, Italy

*Corresponding Author: Praveen Kumar Sekharamantry. Email: pk.sekharamantry@unitn.it

Received: 24 December 2024; Accepted: 25 March 2025; Published: 19 May 2025

ABSTRACT: Recent advances in computer vision and artificial intelligence (AI) have made real-time people counting systems extremely reliable, with experts in crowd control, occupancy supervision, and security. To improve the accuracy of people counting at entry and exit points, the current study proposes a deep learning model that combines You Only Look Once (YOLOv8) for object detection, ByteTrack for multi-object tracking, and a unique method for vector-based movement analysis. The system determines if a person has entered or exited by analyzing their movement concerning a predetermined boundary line. Two different logical strategies are used to record entry and exit points. By leveraging object tracking, cross-product analysis, and current frame state updates, the system effectively tracks human flow in and out of a room and maintains an accurate count of the occupants. The present approach is supervised on Alzheimer's patients or residents in the hospital or nursing home environment where the highest level of monitoring is essential. A comparison of the two strategy frameworks reveals that robust tracking combined with deep learning detection yields 97.2% and 98.5% accuracy in both controlled and dynamic settings, respectively. The model's effectiveness and applicability for real-time occupancy and human management tasks are demonstrated by performance measures in terms of accuracy, computing time, and robustness in various scenarios. This integrated technique has a wide range of applications in public safety systems and smart buildings, and it shows considerable gains in counting reliability.

KEYWORDS: Computer vision; YOLOv8; ByteTrack; cross-product analysis; frame-based counting

1 Introduction

In modern monitoring systems, human tracking models are widely employed in many fields, including crowd control, facility management, and security. Accurately detecting and counting people entering or exiting a place is a significant issue for such systems. This research presents a method that combines movement analysis based on relative position to a defined boundary with object detection. Using logical criteria and tracker states, the suggested system updates entry and exit counts depending on the detection of centroids and their movement direction. Obviously, it is impractical to employ assistance helpers, nurses, or supporting staff for each room of patients or residents in a hospital or nursing home with huge occupancies. This leads to unattended residents who need aid. To overcome this severe issue, a practical model is required to raise an alarm and notify the supporting staff regarding the assistance required.

People counters are now essential for companies and organizations looking to manage operations, track the flow, and enhance the customers they serve. In conventional approaches, people count rely on

infrared beams and pressure pads, which may need to be more precise or work poorly in complex scenarios. However, as computer vision technology advances, people who use machine learning or deep learning are gaining popularity quickly [1]. Deep learning and machine learning models have raised the prospects of people counting. A specifically allocated camera records video footage of the area of interest. After that, computer vision algorithms analyze the camera data, enabling the system to identify and follow specific individuals inside the image [2,3]. The present technologies can accurately calculate the number of people passing through the area by calculating the number of people detected and accurately tracking the number of individuals entering and leaving a room in real-time, offering insightful information for various uses, such as intelligent building automation, facility management, and public safety. Overcrowding occurs when a location can no longer accommodate the number of people using it, and even a minor mishap can cause significant issues [4]. Particularly in the case of hospital or nursing home management, these cases are very critical. As most of the nursing home rooms are shared among different residents, overfilling in a hospital room should be restricted entirely. The initial stage in creating any AI-based crowd-monitoring system is crowd tracking and estimation in pictures. The academic community introduced several automatic crowd-counting algorithms utilizing multiple benchmark datasets. These datasets have changing lighting situations, shadows, and reflections, leading to unbalanced training data. Crowd-counting techniques perform worse because of this imbalance in training data. Any machine learning model, in general, does not perform well when the training data is unbalanced; for instance, an individual resting for an extended period may exhibit some background pixels. This seating region produces an error when separating the foreground from the background pixels [5]. Another significant problem with datasets is the inadequate number of training samples, which leads to lower accuracy. Nevertheless, training data can be improved in both the qualitative and quantitative domains by using several data enrichment approaches. The need for more advanced and dependable systems that track human activity automatically and offer real-time counting data has increased because of these restrictions.

Counting people can be relatively easy in sparsely populated scenarios compared to densely populated scenarios. However, the system's output should be highly accurate. Hence, individual tracking in the healthcare model presents many challenges. There might be the occurrence of situations with extreme illumination or low light conditions to track the number of people in a room, as depicted in Fig. 1. Similarly, counting the number of residents in a nursing home room who are partially or entirely covered may also raise issues that arise, as presented in Fig. 2. Reflections and unforeseen circumstances often impact the estimation of the number of people in a room. The number of people permitted in a hospital or nursing home room is restricted by standard operating procedures, except for visiting hours. The current work is in collaboration with a company, TeiaCare S.r.l, a firm that strives to develop and enhance the resources of healthcare organizations. The data acquisition of the present work is based on healthcare datasets, where the nursing home room scenario with residents is established at the TeiaCare Lab. After acquiring a positive outcome, it is then applied to the real-time nursing home production dataset of video feeds at the Angelo Maj Foundation. The unit deals with Alzheimer's residents with dementia. The production scenarios are complex and unpredictable, and there are many concerns.

In recent years, deep learning techniques, especially convolutional neural networks (CNNs) and object detection algorithms, have significantly advanced object detection and tracking. One such technique is the cutting-edge real-time object recognition framework YOLOv8 (You Only Look Once) [6], which can effectively identify and track objects throughout a sequence of video frames. Owing to its exceptional speed and precision, YOLOv8 is the perfect tool for applications that need real-time object detection, including tracking individuals in security camera footage. This system uses YOLOv8 to identify individuals in every

video frame, determine their centroid, and follow their movements between frames [7]. This methodology guarantees the observation of each person while entering or departing a designated space, such as a room.

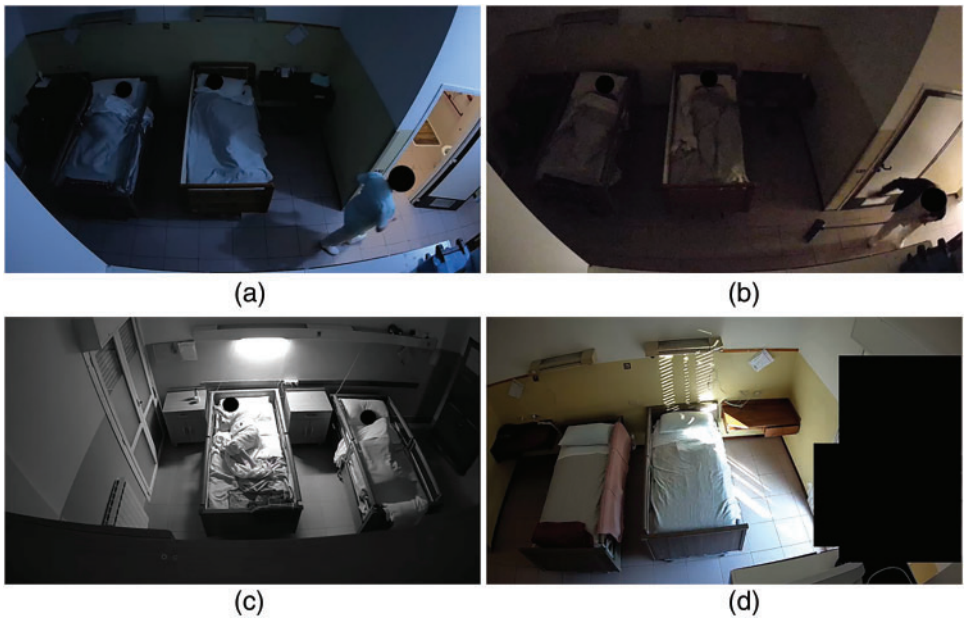

Figure 1: Situations with various lightning conditions: (a) Low light conditions; (b) Extreme low light conditions; (c) Night vision of the camera; (d) Room with light and shadow effects

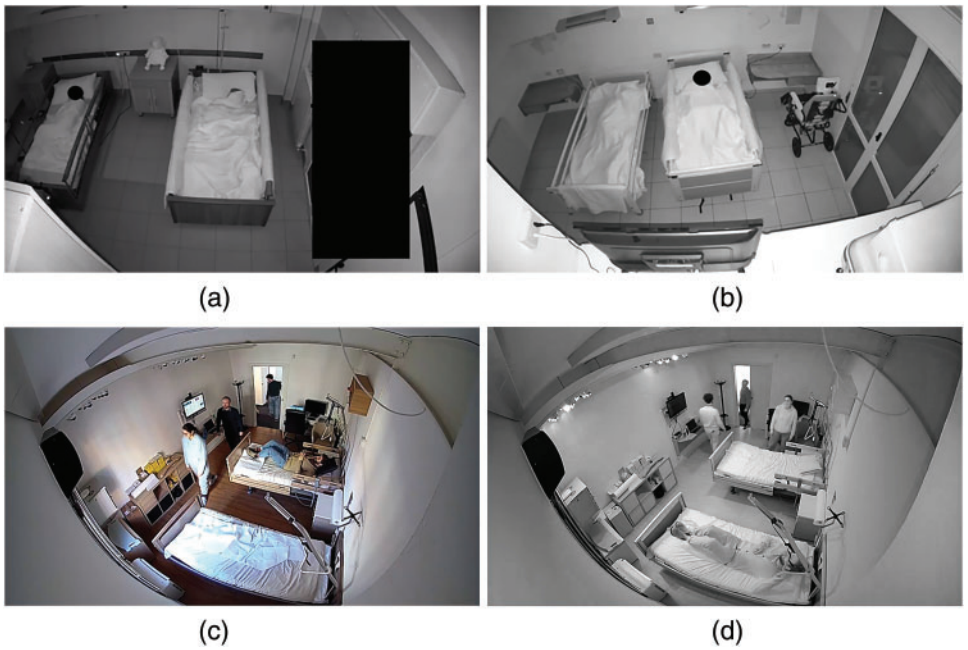

Figure 2: Occurrences of partially or entirely covered individuals; (a) Completely covered residents; (b) Partially covered residents; (c) Overcrowding in a nursing home room; (d) Infrared night vision of exceeding people

After a person is recognized, the system encloses them using bounding boxes and determines their centroids of the geometric centers of the bounding boxes [8]. These centroids are essential when establishing how people move about a predetermined boundary, such as a door or entry. The system's primary goal is to accurately count the number of individuals crossing the boundary, coming in or leaving. To accomplish this, the system uses a mathematical solution called cross-product to determine the direction of the movement. In computer vision [9], the cross-product of vectors is a helpful tool for determining the relative location of a line (the door boundary) and a point (the centroid of a detected human, in this case). By computing the cross-product, the system determines whether an individual is shifting from outside to inside (entering) or from inside to outside (exiting). This differentiation is necessary to update the number of people in the room accurately.

Machine learning and deep learning approaches for people counting encounter several limitations. These techniques sometimes call for large amounts of computational resources, which makes real-time processing difficult, particularly in settings with limited resources. On live streams, the models fail with dynamic corrections, where abrupt shifts in crowd density or movement patterns could make it less accurate. Furthermore, managing night vision settings presents challenges since low light levels could affect the functionality of vision-based models. Occlusions in congested areas make detection much more difficult and may result in errors. Additionally, the flexibility of these models may be limited by their lack of robustness in novel or changing situations. Accurately tracking each person becomes problematic when several people enter and exit a room simultaneously. Moreover, tracking and detection may become more difficult in the event of occlusions, [10], which occur when one person momentarily obscures the view of another. This causes counts to be off, particularly in places with heavy traffic. Addressing these challenges is crucial for developing more efficient and reliable people-counting systems. As a result, a reliable system is proposed to track people despite these difficulties and keep track of how many people are in the room at any given time. Thereafter, ByteTrack obtains the centroids and bounding boxes of the YOLOv8 detection outputs and gives each detected person a unique ID [11]. Indeed, with partial or total visibility, ByteTrack guarantees that people are consistently tracked between frames.

The proposed enhanced model is shown in Fig. 3. The workflow diagram illustrates the steps involved in the human tracking and counting system's functioning, including object identification, centroid computation, cross-product analysis, and counting updates. The input video feed is a video file or live optical sensor recording people crossing a room boundary (such as a door). In every frame, YOLOv8 looks for human figures and returns bounding boxes. For each bounding box containing the detected persons, the centroids are computed. Bounding boxes and centroids from the YOLOv8 detection outputs are used by the ByteTrack Integration to provide each detected person with a unique ID. ByteTrack ensures that people are consistently tracked between frames, even when wholly or partially obscured from view.

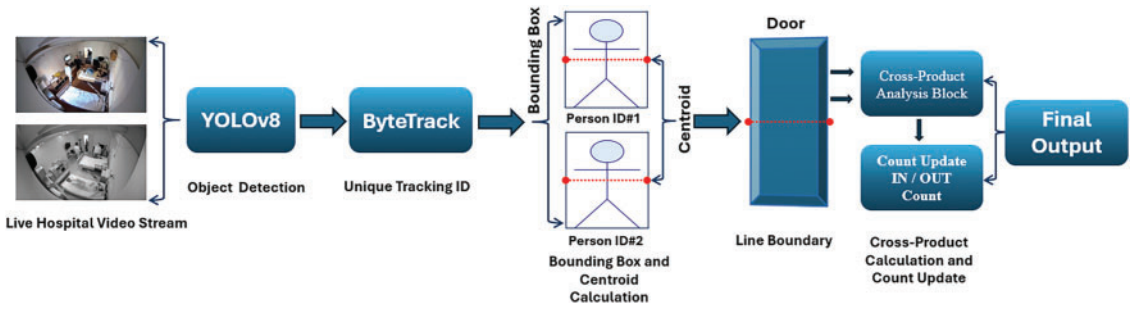

Figure 3: Workflow design of the process

The centroid calculation process for individuals is depicted by the arrow that leads from ByteTrack to the bounding box. The img_to_coord code snippet is used to draw the line boundary (door), and arrows are used to show the centroids as they approach the line. The Cross-Product Analysis block receives an arrow from the centroid calculation that determines the direction of movement (enter/exit). The IN/OUT count shall be updated based on the outcome of the cross-product computation. Based on the most recent counts from cross-product analysis and ByteTrack-based tracking, the last block displays the total number of people in the room. The foundation for the development and execution of a reliable system for tracking and counting people is laid forth in this introduction. The system can precisely track entry and exit events in real time thanks to the integration of YOLOv8 and ByteTrack for object recognition and tracking capabilities, along with centroid tracking and cross-product analysis.

1.1 Contribution of the Paper

Currently, people counting models with large-scale networks that can recognize objects in images with a high population density do not perform well on datasets with a high degree of occlusions [12]. Therefore, a customized model is required to address the shortcomings of current methods and provide an error-free, effective approach. The proposed study's main contributions are summarized as follows:

- The current work proposes that Regions of Interest (ROI) in the model be easily adjusted to focus on areas of interest within the video data. The model's versatility allows it to be adjusted to suit various settings based on the fields of view from various camera angles. Consequently, it is possible to prioritize object tracking and identification in user-defined regions of interest.
- The integrated approach of YOLOv8 and ByteTrack framework, along with the enhanced logic, would result in accurate outcomes.
- At any given point in time, the model can effectively predict the current number of people inside the nursing home room.
- A cumulative count of all the people who entered or exited the room can be obtained for additional analysis on a daily, weekly, or monthly basis.
- The proposed model requires no additional mechanism or hardware as it uses current CCTV (closed-circuit television) equipment for monitoring and passenger counts.

This work's primary contribution is the creation of a revolutionary real-time people counting system that precisely tracks persons entering and leaving a room by combining frame difference and cross-product analysis techniques. This approach improves precision by effectively determining movement direction utilizing centroid tracking and motion vector calculations. The system can be used for real-time applications without requiring a lot of processing power because it is also computationally efficient. The model ensures accurate results even in complicated surroundings and provides a dependable solution for scenarios demanding real-time occupancy monitoring. The novelty of the proposed work is that it uses two distinct logics to precisely capture the entry, exit, and total occupancy in each room of a facility. The recommended methodology combines object detection and tracking using cross-product analysis and frame-based difference logic to improve accuracy and adaptability, in contrast to conventional frame-by-frame detection techniques. By preserving identification across frames, object tracking lowers errors brought on by motion blur or occlusions. Cross-product analysis may differentiate between people entering and going out of a room by accurately identifying the direction of movement. Furthermore, despite dynamic scene changes, real-time updates based on the current frame state guarantee that the occupancy count stays correct. In various settings, this combination approach greatly increases the dependability of human flow monitoring. This work proposal is divided into five sections. Section 2 presents a comprehensive review of related literature. Section 3 introduces the ByteTrack model and the Enhanced YOLOv8 approach. The thorough analysis and comparative examination

of the findings are presented in Section 4. Section 5 of the study provides discussion and Section 6 is the conclusions and future directions.

2 Literature Review

Human recognition and tracking capabilities are improved in hospital or nursing home room entry monitoring systems by integrating YOLOv8 and ByteTrack. This method uses sophisticated algorithms to increase speed and accuracy in situations that happen in real time. Some recent efforts have been presented as improvements to YOLOv8 with the c2fELEMA and Asymptotic Feature Pyramid Network for Object Detection (AFPN) modules [13]. This has led to a 4.0% rise in mAP @ 0.5:0.95 and a 3.2% increase in mAP @ 0.5, considerably increasing detection accuracy while decreasing computational load. This improved detecting capacity is essential for monitoring behavior in various settings, such as private spaces and classrooms [14]. Monitoring people outside of camera-prohibited regions to track their time spent and notice suspicious changes in the suspected people being tracked is one of the primary intentions of the work. YOLOv8 is enhanced with ByteTrack, which offers reliable tracking of identified people across several video feeds [15], guaranteeing ongoing observation and precise location mapping inside a virtual floor layout. The model presents real-time tracking of specific individuals in rooms by utilizing YOLO neural network analysis with a multi-camera system. Using intelligent and digitalized technology to gather occupancy patterns enables real-time tracking, which is essential for applications such as security and occupancy management [16]. On the other hand, although these technologies present tremendous progress, privacy issues still need to be addressed, especially in delicate areas where monitoring can be invasive. A similar approach was incorporated into apple fruit detection, integrating YOLO and ByteTrack which had a seamless approach to identifying and tracking objects with better accuracy. Few attention mechanisms have enhanced architecture for better outcomes with improved loss function and novel techniques. A reliable method for counting individuals is the dynamic kernel convolution neural network-linear regression (DKCNN-LR) model, especially in crowded areas where occlusion might be a significant problem [17]. Using a two-phase approach, this model uses linear regression and dynamic kernel modifications to improve counting accuracy. While the DKCNN-LR model performs best in congested areas, other approaches with different capabilities in tracking and detection, such as Faster R-CNN and Long-Term Recurrent Convolutional Network (LRCN) LRCN-RetailNet, also show potential in people counting [18]. In surveillance, person tracking must be improved by shadow removal techniques [19]. Many techniques using distinct strategies have been implemented to successfully remove shadows and enhance tracking precision. The positioning errors of a method that uses a density-based score fusion system to combine physical attributes and chromatic features have decreased from 44.4 to 13.5 cm, indicating its efficiency in person localization. When the Gaussian mixture model and chromatic color model combine, moving object identification is improved in cluttered scenes by successfully differentiating shadows from foreground objects. This improves counting accuracy. It has been demonstrated that pedestrian tracking accuracy can be increased in congested indoor environments by using a technique that updates backdrop models and removes shadows [20]. Even though these developments significantly improve tracking performance, problems still arise in dynamic contexts with changing lighting. Therefore, further study is needed to improve these methods further.

Optical sensors combined with deep learning-based human tracking systems have shown to be a successful way to improve public transit efficiency. These technologies handle obstacles like occlusions and changing ambient circumstances by precisely detecting and tracking passengers using sophisticated algorithms. A model uses DeepSORT for tracking and the YOLOv8 object detection method to perform better under various situations. The Single Shot Multibox Detector identifies passenger characteristics and improves counting precision in difficult situations such as congested areas and poor light levels. When

combined with DeepSORT, Tiny-YOLOv4 shows that this model greatly increases by counting accuracy from the above views while reducing erroneous predictions. Routes and schedules for public transportation can be optimized with accurate passenger flow detection, improving service delivery. Despite the enormous potential of these systems, there are still difficulties, especially guaranteeing resilience to a variety of real-world situations and integrating with the current infrastructure. Few recent studies using YOLOv8 focused on creating parametric datasets utilizing autonomously operated unmanned aerial vehicles (UAVs). The work compared the YOLO family with YOLOv8 on intricate real-world traffic scenarios. The work compared the YOLO family with YOLOv8 on intricate real-world traffic scenarios. The model improved their robustness and dependability in real-world applications. A similar vehicle classification was presented by optimizing all vehicle types and demonstrating that YOLOv8 has high precision. Transportation security also integrated Intelligent Transportation Systems (ITS) with UAVs to deal with urban mobility. Technically a most recent study focusses on saliency detection with the EfficientNet-B7 backbone, and multi-scale feature extraction improves the accuracy of salient object detection (SOD). The Spatial optimized Feature Attention (SOFA) module uses the Three initial-stage feature maps to refine spatial features. To capture multi-scale contextual information from the mature three layers and enhance robustness under a variety of circumstances, the proposed Context-Aware Channel Refinement (CACR) module combines dilated convolutions with optimum dilation rates followed by channel attention. The system might face challenges with complex environmental conditions. Similar work on fall detection of humans improves spatial and channel contexts for better detection, particularly in complex scenarios, the convolutional block attention modules (CBAMs) are added at the feasible stages of the network [21]. A focus module is incorporated into the backbone of the YOLOV8S model as part of a series of improvements that are suggested to maximize feature extraction.

Handling movement pattern ambiguity is one of the leading design issues for such a system. For example, suppose an individual walks erratically or stands motionless close to the boundary. In that case, the cross-product computation may produce contradicting results, making it challenging to ascertain whether the individual has entered or exited. To lessen this, the system only updates its count when a distinct movement direction is identified, ignoring ambiguous movements in favor of logical requirements. Furthermore, tracking errors or missed detections may arise from occlusions in which one person blocks another [22]. To combat this, the system keeps track of each recognized person's identity by state tracking, even in the event of occlusions. This guarantees that when individuals reappear in the frame, their counts will be accurate. As a result, a novel version of the system is needed for reliable tracking that ensures a person's identity even when faced with occlusions or multiple people in view, preventing errors like double counting or missing entries or exits.

3 Methodology

Deep learning breakthroughs have enabled overcoming human detection issues using object tracking algorithms like ByteTrack and object detection models like YOLOv8. To create a dependable real-time system for counting individuals as they pass a predetermined boundary, like a door, this article investigates the integration of YOLOv8 with ByteTrack by computing both enter and exit counts based on their direction of movement [23]. The proposed system processes human detections and tracks the user's movement across a predefined line using two main strategies. After processing each detection, the overall number of people are updated based on whether the person has crossed the line from inside to outside or *vice versa*.

3.1 Object Detection with YOLOv8

YOLOv8, a cutting-edge, real-time object identification technology, belongs to the YOLO model family. For various computer vision applications, including detecting people in video feeds, it is made to be quick, precise, and simple to use [24]. In contrast to specific other object detection methods [25], YOLOv8 uses a single forward neural network pass to process the entire image. The model generates bounding boxes and class probabilities for each grid cell in the divided image. YOLOv8 has been trained to identify people among the different classes it can locate [26]. YOLOv8 is used to identify individuals in every video feed frame. It draws bounding boxes around each person in the camera's field of view to identify them accurately, as in Eq. (1). Every frame in the video feed goes through this procedure again, giving the appearance that people are being tracked in real time as they walk through the frame. These coordinates are essential for the subsequent centroid computation and direction assessment stages.

$$Bounding\ Box = (p_{min}, q_{min}, p_{max}, q_{max}) \tag{1}$$

The centroid of a bounding box is a crucial concept in object tracking and detection. Unlike several techniques that use the complete bounding box, it gives an object's position as a single point. You may monitor an object's movement by determining its centroid over a series of frames. It is frequently used as a benchmark for calculating object distances. Computational complexity can be decreased in certain algorithms by employing the centroid rather than the full bounding box. The following formula, as in Eq. (2), is used to find the bounding box's centroid:

$$Centroid = \left(\frac{p_{min}+p_{max}}{2}, \frac{q_{min}+q_{max}}{2}\right) \tag{2}$$

where

$p_{\min}$ and $p_{\max}$ are the x-coordinate of the left and right edge of the bounding box,

$q_{\min}$ and $q_{\max}$ is the y-coordinate of the top and bottom edge of the bounding box.

Each person's center point is represented by this centroid, which is utilized to determine the direction of movement.

3.2 Tracking Objects Using ByteTrack

YOLOv8 and ByteTrack are combined to guarantee that every recognized individual is tracked across several frames. ByteTrack enables the system to consistently identify each observed person by assigning unique tracker IDs to them. Every individual found by ByteTrack is given a tracker ID. ByteTrack sets a new ID if the detection is new and uses the same ID if the individual has already been tracked in earlier frames. ByteTrack uses both high and low confidence detections, which is crucial to its performance compared to standard trackers that only employ high confidence detections. ByteTrack builds and maintains tracks by associating detections between frames. It combines IoU (Intersection over Union) with Kalman filtering [27] to forecast and associate object positions. ByteTrack prediction approach keeps traces even when some frames contain missed detections. ByteTrack is especially effective at handling occlusions, it ensures that people who are temporarily out of the camera's field of vision are tracked when they re-enter it [28]. A person's track will finally end by ByteTrack if they are not visible for a predetermined number of frames. This combination works exceptionally well for keeping constant tracks when there are many moving objects or in busy areas. People must be identified and tracked in each frame to update the in and out counts. The next step is to ascertain whether they crossed a predefined boundary line, like a door threshold.

3.3 Counting Logic and Moment Direction

Updating the IN/OUT counts involves tracking and recognizing everyone in each frame and then figuring out how they moved across a set boundary, like a door threshold. The system processes human detections and tracks the users' movement across a predefined line using two main strategies. One approach addresses entry and exit counts about cross-product analysis and the line of reference. The second approach applies to the position tracker and deals with the previous frame's status in relation to the current frame.

3.3.1 Strategy 1: Counting and Detection of Entry/Exit

Strategy 1 aims to monitor every identified individual's movement and adjust the in and out counts as they cross the designated door threshold. These phases are centered on tracking the direction of movement, determining the number of individuals who cross the door in either direction or changing tracker status.

(a) Initialization (Crossed States): The cross_in and cross_out are two Boolean arrays that the system initializes. These arrays monitor if every detection has entered or exited the room through the door threshold. These arrays are initially configured to be False for every detection.

(b) Dealing with No detections: The function returns the cross_in and cross_out arrays without processing them further if there are no detections available in the current frame. By doing this, the system can effectively skip empty frames.

(c) Anchor Points: The bounding box (from YOLOv8) coordinates are divided into two points (top-left and bottom-right corners) for each detected individual. The centroid of the bounding box, which shows the subject's location in the frame, is computed using these coordinates. For every detection, the system retrieves the bounding box anchor coordinates and saves them in all_anchors.

(d) Processing Each Detection: The system confirms that there is a valid tracker id for every object it detects. We skip the detection if no tracker_id is supplied. For every detection, the bounding box's centroid, or center point, is computed once the bounding box anchors are transformed into Point objects.

(e) Check Boundaries: The function checks whether the centroid is inside the monitored area's specified spatial bounds. The detection process is skipped if the centroid is outside of these bounds.

(f) Determine Direction: To identify whether a movement is toward entry (crossing inside) or exit (crossing outside), the system computes the cross-product as in Eq. (3) between the vectors of the designated line of reference and the detection centroid as shown in Fig. 4.

$$Cross\ Product = (S_x - P_x)\left(Q_y - P_y\right) - \left(S_y - P_y\right)(Q_x - P_x) \tag{3}$$

where the centroid of the individual is S, and the endpoints of the boundary line are P and Q.

A movement from the outside to the inside (entrance) is indicated by a positive cross product, whereas an exit from the inside to the outside is shown by a negative correlation.

It is skipped if the direction is unclear, that is, if the detection shows movement in both directions.

A mathematical method for determining if a point (person) is traveling across a boundary in a positive or negative direction is the cross-product. This computation's outcome provides information on whether the motion is entering (IN) or exiting (OUT).

- $S\ (S_x, S_y) \rightarrow$ The moving person's centroid.
  - S_x = The person's centroid's X-coordinate
  - S_y = The person's centroid's Y-coordinate
- $P\ (P_x, P_y) \rightarrow$ The boundary line's initial endpoint.

P_x = The first boundary point's X-coordinate
P_y = The first boundary point's Y-coordinate

- Q (Q_x, Q_y) → The boundary line's second endpoint.
  Q_x = The second border point's X-coordinate
  Q_y = The second border point's Y-coordinate

The formula $(S_x - P_x)$ $(Q_y - P_y)$ calculates the centroid S horizontal distance from point P and multiplies it by the vertical distance between points Q and P.

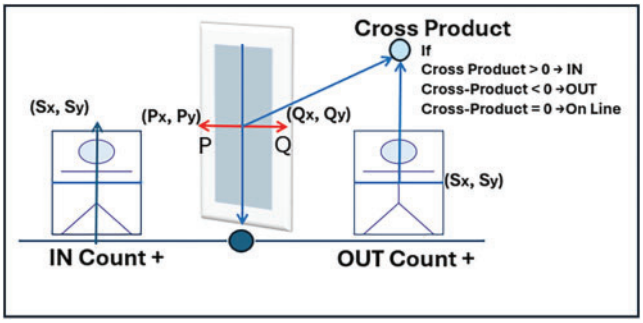

Figure 4: Illustration of cross product logic

The formula $(S_y - P_y)$ $(Q_x - P_x)$ determines the centroid S vertical distance from point P, which is then multiplied by the horizontal distance between points Q and P.

The final cross product result indicates whether the movement of the point S with respect to the boundary line PQ is positive or negative.

The interpretation of the cross product is on conditions

- If Cross Product >0 The motion is IN (Person Entering)
- If Cross Product <0 The motion is OUT (Person Exiting)
- If Cross Product is =0 The location is precisely on the border

The cross-product approach is a helpful tool for automated tracking systems since it helps determine a person's direction of movement.

The mathematical example calculation of a scenario is as follows:

The boundary endpoints are $P = (1, 1)$, $Q = (4, 1)$

The persons centroid positions before motion are $S = (2, 0)$ (exit) and after motion is $S = (2, 2)$ (inside)

The computation of cross product for S (2, 0) outside is $(2 - 1)$ $(1 - 1) - (0 - 1)$ $(4 - 1) = 3$

Since $3 > 0$ the movement is IN (Entry detected). A negative outcome would have occurred if the individual had proceeded in the opposite way. The below graph in Fig. 5 represents the same.

(g) Update the Status of the Tracker: The function determines whether the tracker_id for the current detection is present in tracker_state. If not, it sets the tracker's status to zero. If the direction of movement differs from the previous movement, the tracker's state is updated. The cross_in array is changed to True, and the in-count increases when someone moves from the outside to the inside. Moving from inside to outside causes the cross_out array to be updated and the out count to increase.

(h) Update Counts: After direction determination, the algorithm increases the relevant numbers according to whether the person crosses the line inward or outward and the current number of persons in the

room. As a result, every individual entering or leaving the area is only counted once, and the counts are precisely updated in accordance with their motions.

The proposed models' primary advantage is that they perform best when people gradually cross the threshold. Similarly, it could have trouble with occlusions or fast movement, which could make the cross-product unclear and result in missed detections.

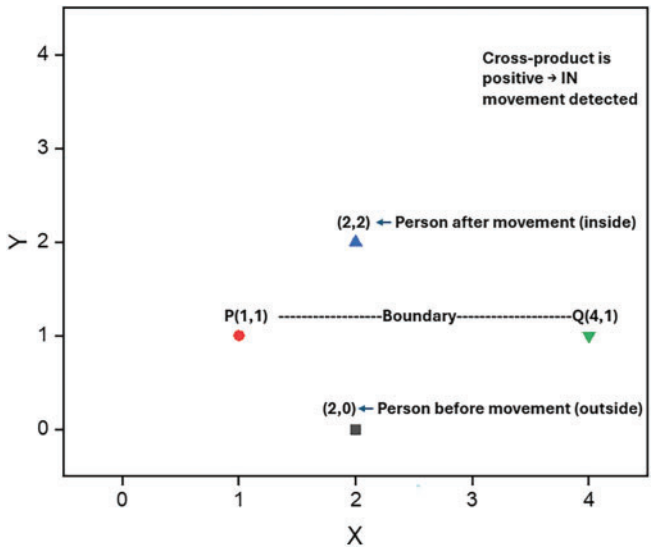

Figure 5: Graphical visualization of the example scenario

3.3.2 Strategy 2: Improved Counting with Frame Adjustment

A different strategy is used in Strategy 2, which counts the total number of individuals in each frame and determines the in_count and out_count by counting the number of people who enter the room and the number who exit it between frames. This technique is beneficial for situations where a lot of traffic or individuals are going simultaneously.

(a) Initialization: Like in strategy 1, each detection in the current frame initializes two Boolean arrays (cross_in and cross_out) to False. These arrays will be utilized to keep track of who has passed across the door threshold in this frame.

(b) Dealing with no detection: The function returns empty cross_in and cross_out arrays immediately if no individuals are in the frame. This guarantees that the system makes no unnecessary calculations.

(c) Centroid Calculation and Anchor Coordinates: The bounding box coordinates of each detected individual are transformed into two points (the top-left and bottom-right corners). The person's location in the frame is then represented by the centroid of the bounding box, which is computed for each detection.

(d) Evaluate the Movement Limits: Like Strategy 1, the system determines if the person's computed centroid falls between the predetermined bounds (close to the door). The individual is only considered if they are inside these parameters.

(e) Cross product to determine direction: The system uses the cross-product to calculate the movement's direction. The IN/OUT counts are updated appropriately if the person moves from the outside to the inside or *vice versa*. Any direction uncertainty is cleared up by avoiding the detection.

(f) Tracker State Management: Each person's tracker_id is assigned and managed by ByteTrack, which tracks their movement between frames. As in strategy 1, each person's state (inside or outside) is saved in the tracker_state dictionary. Every time a detection moves over the threshold, the system updates the state.

(g) Frame-Based Counting: The key difference is the method used by Strategy 2 to determine the number of individuals in the room. To determine how many individuals are in the room, Strategy 2 considers the number of people observed in the previous and current frames rather than increasing or decreasing counts based on crossing events. Calculating In Counts: The total number of people in the room in the previous frame is subtracted from the number of people who left in the current frame to update the in_count. This ensures the system dynamically adjusts the total count according to who is still inside. It can be stated as follows:

in_count = (Total people in the previous frame) − (People who exited in the current frame)

Calculating Out Count: The out_count is also updated to reflect the number of individuals who exited the room during the current frame and crossed the door threshold. The system uses the previously discussed cross-product approach to track these movements.

This method is more reliable when there is a lot of traffic, or several people pass the threshold simultaneously. It also considers individuals who might be occluded or move too quickly for direct tracking. Despite being more accurate in congested situations, strategy two can be marginally slower than strategy one because of the extra frame-based computations. A specific config file is configured to adjust the settings of each room with the number of cameras operating and total number of doors in the scene allowing it to adapt to different room dimensions. The means of detecting the residents in the room and assignment of class ID by tracking is similar for both the proposed logics but a minor difference occurs in the assessment of the total residents inside the room at a given point of time. The comparison of both approaches is presented in Table 1.

Table 1: Comparison of both strategies

Feature	Strategy 1	Strategy 2
Count mechanism	Updates count directly in response to movement through the threshold in the current frame.	Updates count according to the total number of exits and the overall number of people across frames.
Tracker management	Tracker states are updated in response to movement; duplicate detections are skipped.	Updates counts depending on frame-based totals and maintains tracker states across frames.
Accuracy	Efficient in areas with constant movement and moderate traffic.	It is more accurate when there is a lot of traffic or when several people cross simultaneously.
Real-time performance	Fast and effective, suited for real-time applications in low-traffic situations.	Somewhat slower because of frame-based logic but it is accurate for many ideal situations.
Occlusion handling	Might overlook individuals who are occluded and do not step beyond the boundary.	Tracks the state of individual frames to account for occlusions.

4 Results

The proposed system keeps precise records of the number of individuals entering and leaving a room and efficiently tracks human mobility. Strategy 1 offers direct tracking of individual detections over the boundary line. While Strategy 2 further refines this by taking room occupancy over consecutive frames, even under unclear circumstances, orientation can be reliably detected due to the cross-product analysis.

The results of the recommended YOLOv8 framework for detection and the ByteTrack method for tracking are shown in this section. The real-time electronic, optical sensor video evaluates the model feeds from the TeiaCare lab dataset. The PyTorch deep learning framework was utilized with the Ubuntu 22.04 Linux operating system to construct the proposed Alzheimer's resident tracking and detection approach. The operating system was installed with an Intel i7 processor, 24 GB of RAM, and an NVIDIA GeForce RTX 3090 linked to a 384-bit memory interface. The Graphics processing unit (GPU) operated at a frequency of 1395 MHz. The Python programming language was used to write the entire model. This model uses YOLOv8 and ByteTrack, and it has enhanced each model's design and performance with the aid of the CUDNN library and CUDA toolkit. Every step of the experiment is done using an IoU threshold of 0.75.

The amount of time the algorithm needs to process and investigate each frame of the input video stream that has been recorded is the processing time assessed in this study. On the other hand, memory use refers to how much system memory with GPU support is needed to execute an algorithm. A GPU-capable computer system was used to determine the computational cost of our solution. We processed photos using the OpenCV 4.10.0 software and developed the method in Python 3.12.7. We calculated the entire execution time that the algorithm collected on the video frames to calculate the processing time. We measured the average processing time per frame by recording the start and finish times of the processing pipeline using the Python "time" library. The 'memory_profiler' Python module examined memory utilization, allowing us to track the algorithm's memory consumption while it ran. We averaged the peak memory consumption over multiple iterations for a typical estimate. Our computational cost investigation found a 2-ms standard deviation and an average processing time of 20 ms per frame. It was found that the algorithm's peak memory usage was approximately 300 MB.

4.1 Performance Assessment

Three accuracy metrics—precision, recall, and F1 score—were used to evaluate the effectiveness of the suggested strategy. To calculate these parameters, use Eq. (4) as follows:

$$Pr = \frac{T_P}{F_p + T_P}, Rec = \frac{T_P}{T_P + F_N}, F1score = \frac{2P_r Rec}{P_r + Rec} \tag{4}$$

where T_p, F_p, and F_N denotes the true positive, false positive, and false negative values.

4.2 Performance of Proposed Model on Alzheimer's Residents Counting

This section presents the findings from the proposed system, which includes two different logic-based approaches (Strategy 1 and Strategy 2) for counting the number of Alzheimer's residents passing a door threshold, YOLOv8 for human detection, and ByteTrack for tracking. Real-world video footage was used to evaluate the system under various conditions, such as variable lighting settings, crowd density, and motion speeds. An in-depth examination of the system's advantages, disadvantages, and general efficacy is provided in this section, which also contrasts the performance of Strategy 1 with Strategy 2.

4.2.1 Data Acquisition: To Efficiently Tabulate the Data from the Two Camera Channels (Channels 0 and 1) Used to Record Alzheimer's Residents Nursing Home Room Footage. The Details Are Shown in Table 2

The two channels are two camera videos of Alzheimer's residents that were recorded in a Lab set up of a nursing home at the company TeiaCare S.r.l. Each channel represents a distinct camera angle or viewpoint. Images obtained replicate the exact real time scenario and with varying backgrounds. The data acquired facilitates the investigation and evaluation of the behaviour or movements of residents with Alzheimer's disease in regulated hospital or nursing home settings. The dataset is divided into three parts as 70% for training set, 20% for testing set, and 10% for validation set.

Table 2: Nursing home room video dataset (Alzheimer's residents)

S. No.	Channel	Total videos	Each video length (min)	Resolution	File format
1	Channel 0	10	5	1920 × 1080	MP4, AVI, etc.
2	Channel 1	10	5	1920 × 1080	MP4, AVI, etc.

4.2.2 Experimental Results

The data procurement is processed with different possible considerations which would best reflect the outcomes of the model. The following data set observations are recorded for testing the model efficacy.

- The sequential and simultaneous entry and exit of residents from a room
- Approaching the door from inside, then stay in the room
- Approaching the door from outside, then staying out of the room
- Walking in the corridor
- Chaotic scenes day and night mode
- Residents stay inside the closed room and perform actions near the door

The above-mentioned scenarios are the major day-to-day process which can be encountered in the real time Alzheimer's residents nursing room. The proposed strategies are tested on these facility datasets and evaluated using metrics for checking accuracy. In this context the final deliverables as outcomes after the post-processed videos reporting for each frame are

- Current number of people inside the room
- Total number of people who left the room
- Total number of people who entered the room

The outcomes illustrated in Fig. 6 describes the different occurrences where the model will be tested to fetch accurate results. The key concept in this approach is to detect residents in the vision of the camera and then track them with the movement of the bounding box. The centroid of each bounding box of the human class is verified to the acquired line of reference coordinates of the door. Based on the evaluated positive and negative values, the IN/OUT count is confirmed. Fig. 6a–c deals with the regular sequential approach of entry and exit where both strategies are giving accurate results to maintain the count of the total residents' flow. Fig. 6d–f deals with the simultaneous entry and exit of the residents, where there is a high possibility of swapping the tracking ids of the bounding box. Likewise, there is a high chance of missing the count of residents' entry and exit with respect to the line of reference of the door. The situation often leads to the incorrect count when an entry or exit of a person superimposes on another entry or exit. The proposed Strategy 1 often affects these pitfalls. On the contrary the Strategy 2 would curtail this drawback to maximum extent. The outcome of Fig. 6g–i represent the possibility of a residents wandering inside the room and appearing close to the line of reference of the door but not crossing the door. The vision from the optical sensor would appear as if the person has crossed the line, but both the strategies have exhibited

positive results in this scenario. A similar approach is tested with residents approaching close to the door and line of reference but not entering inside the nursing home room as in Fig. 6j–l. The residents walking in the corridor may also be visible to the camera which should be excluded from the total people count as in Fig. 6m–o. Accurate results were obtained by both the strategies in these instances. In Fig. 6p–r, the actions often performed near the door like resident bending at the door affecting the line of reference or other uncertain actions may result in false positive actions. These occurrences are resolved by the logic proposed. Finally, the outcome displayed in Fig. 6s–u are the situations where the YOLOv8 might miss tracking few residents due to various conditions like low light or complication in detecting in the video stream. These circumstances occur very frequently in a hospital or nursing home room where few residents will not be detected with the bounding box. The grouping issues of object detection model may also be one of the reasons. There might be few adjustments applied to the object confidence score or non-maximum suppression which aid in identifying groups of bounding boxes that heavily overlap, choose which boxes to keep and which to discard. The proposed strategy 1 will persist the fault count of the total person in the room with the missed calculation, if the tracker of a resident is lost. This issue is resolved by Strategy 2 as it checks the difference between the previous frame and current frame and only changes the count if only a resident either way crosses the line of reference. Therefore, we can find the total count of people inside the room in strategy 1 in Fig. 6t is 2, but the actual count of residents is 3. The model missed a person who was bending at the desk beside the bed. Meanwhile as per strategy 2 in Fig. 6u, the exact count of 3 people inside the room had resulted.

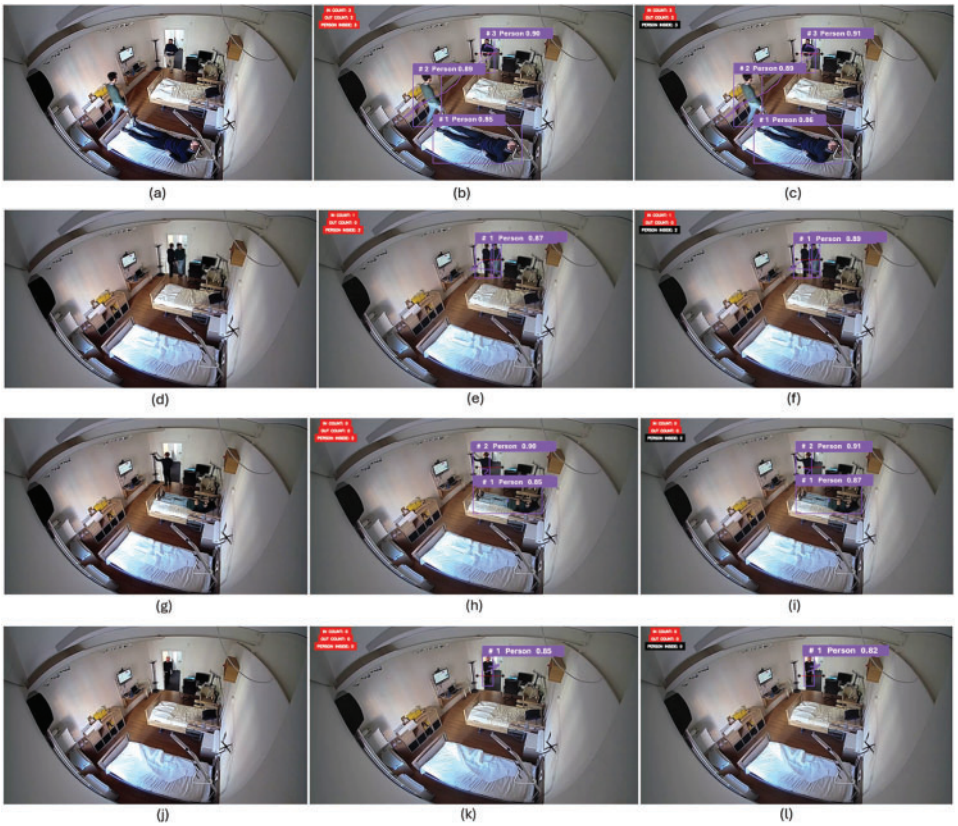

Figure 6: (Continued)

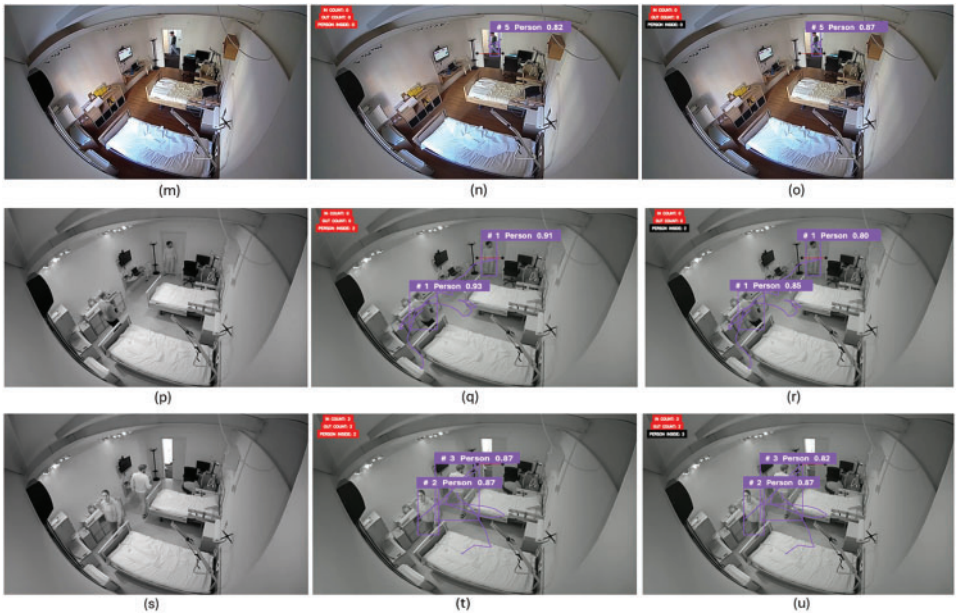

Figure 6: (a) Sequential entry and exit of residents from a room; (b) Output of strategy 1 for sequential entry and exit; (c) Output of strategy 2 for sequential entry and exit; (d) Simultaneous entry and exit of residents; (e) Outcome of simultaneous entry and exit for strategy 1; (f) Outcome of simultaneous entry and exit for strategy 2; (g) Approaching the door from inside, then stay in the room; (h) Strategy 1 output of residents approaching the door from inside, then stay in the room; (i) Strategy 2 output of residents approaching the door from inside, then stay in the room; (j) Approaching the door from outside, then stay out of the room; (k) Result of Strategy 1 for approaching the door from outside, then stay out of the room; (l) Result of Strategy 2 for approaching the door from outside, then stay out of the room; (m) Walking in the corridor; (n) Strategy 1 detecting people outside the door but not counting; (o) Strategy 2 detecting people outside the door but not counting; (p) Residents stay inside the closed room and perform actions nearby door; (q) Strategy 1 not updating count until crossing the line of reference; (r) Strategy 2 not updating count until crossing the line of reference; (s) Chaotic scenes night mode; (t) Strategy 1 detection of residents in chaotic scenes, missing one resident count; (u) Strategy 2 detection of residents in chaotic scenes without missing any resident count

Therefore, from the outcomes it is evident that both approaches have established their accuracy in serving the purpose of tracking residents and evaluating the count of the residents in each room of a nursing home. The proposed model is a tool to support nurses and carers in guaranteeing the highest quality of care. It guarantees residents' safe, transparent, and high-quality care assistance with continuous monitoring and analysis of their status. It also provides a solution for the manager to overcome strategic and managerial challenges.

The performance evaluation is measured on the acquired dataset of a nursing room of Alzheimer's patients with two electronic cameras of channel-0 and channel-1 with a different angle of vision. Each channel has an equal number and length of videos of residents moving in and out of the room. Both channels record simultaneously to measure the proposed model from different angles of image sight.

The precision, recall, and F1 score are the metrics used to assess the model's performance at different sequences and various possibilities. The performance metrics in Tables 3 and 4 are measured in the cases of TI as the total number of times any person has entered a particular room. The TO calculates the total number of times any person has left the room. They employed two logics, l1 for strategy 1 and l2 for strategy 2, to count the total number of people inside the room at a current point of time. The proposed work has been tested on different conditions and has achieved a successful result in each of the circumstances.

The graphical depictions in Figs. 7 and 8 indicate the overall accuracy of all the videos, collectively producing a positive result.

Table 3: The performance evaluation of total IN/OUT and persons inside of both Strategy 1 and Strategy 2 for Channel 0

Id	Precision _TI	Recall _TI	F1_Score _TI	Precision _TO	Recall _TO	F1_Score _TO	Precision _PI_l1	Recall _PI_l1	F1_Score _PI_l1	Precision _PI_l2	Recall _PI_l2	F1_Score _PI_l2
1	99.68	94.66	97.10	99.77	90.70	95.01	94.27	93.35	93.80	96.37	94.23	95.28
2	98.52	94.76	96.60	98.86	95.83	97.32	98.54	83.33	90.29	98.89	93.82	96.28
3	98.60	94.48	96.49	98.00	85.94	91.57	95.47	86.25	90.62	95.37	97.92	96.62
4	99.23	93.16	96.09	99.58	90.43	94.78	96.83	94.44	95.62	95.16	96.72	95.93
5	98.23	96.30	97.25	75.56	100.0	86.07	100.0	94.20	97.01	100.0	100.0	100.0
6	100.0	98.53	99.25	100.0	99.65	99.82	100.0	100.0	100.0	100.0	100.0	100.0
7	99.56	95.69	97.58	100.0	96.55	98.24	99.11	92.12	95.48	99.19	98.39	98.78
8	99.23	99.88	99.55	98.74	98.89	98.81	100.0	98.82	99.40	100.0	100.0	100.0
9	97.89	95.32	96.58	95.80	96.68	96.23	97.77	85.45	91.19	97.85	90.78	94.18
10	98.10	93.43	95.70	96.55	94.21	95.36	98.16	88.28	92.95	98.37	93.28	95.75

Table 4: The performance evaluation of total IN/OUT and persons inside of both Strategy 1 and Strategy 2 for Channel 1

Id	Precision _TI	Recall _TI	F1_Score _TI	Precision _TO	Recall _TO	F1_Score _TO	Precision _PI_l1	Recall _PI_l1	F1_Score _PI_l1	Precision _PI_l2	Recall _PI_l2	F1_Score _PI_l2
1	98.23	95.30	96.74	99.56	95.86	97.67	94.23	93.23	93.72	96.63	94.23	95.41
2	97.54	94.76	96.12	99.85	94.32	97.06	97.85	94.09	95.93	97.85	94.62	96.20
3	98.50	78.56	87.40	98.63	70.23	82.04	95.30	84.55	89.60	98.85	94.23	96.48
4	85.20	97.87	91.09	85.12	95.45	89.98	91.67	95.69	93.63	91.67	95.69	93.63
5	75.85	88.96	81.88	78.65	88.02	83.07	100.0	95.16	97.51	100.0	99.19	99.59
6	99.52	96.67	98.07	98.96	98.21	98.58	100.0	98.39	99.18	100.0	98.39	99.18
7	98.65	100.0	99.32	98.08	100.0	99.03	100.0	99.23	99.61	99.85	100.0	99.92
8	97.68	85.20	91.01	99.83	92.74	96.15	94.49	91.26	92.84	95.94	98.82	97.35
9	97.90	90.23	93.90	85.13	94.65	89.63	96.56	89.92	93.12	96.45	94.23	95.32
10	98.65	84.52	91.04	97.32	98.30	97.80	96.24	97.58	96.90	95.74	98.39	97.04

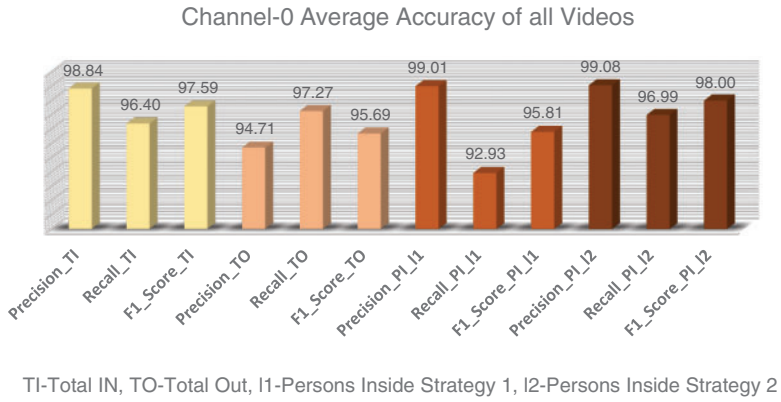

Figure 7: Average metrics of all the videos of channel-0

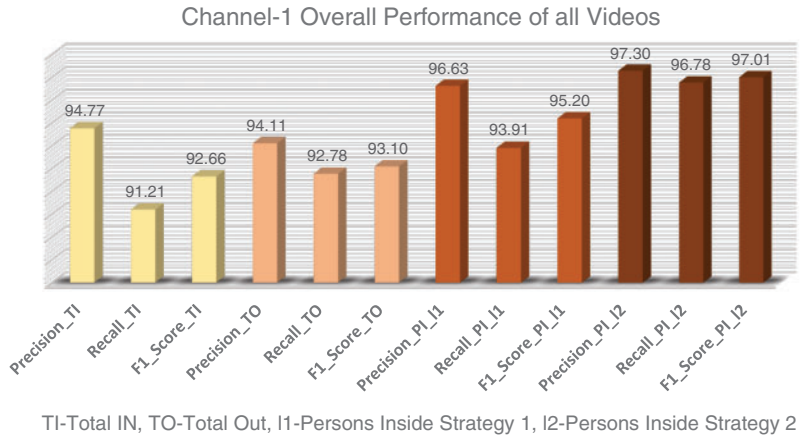

Figure 8: Average metrics of all the videos on channel-1

The detection using YOLOv8 is the same for both approaches. Every human is well detected with a high detection accuracy demonstrating, that the model has very few missed detections and extremely few false positives. The ID tracking is consistent across all videos tested, and ByteTrack performance is ideal for accurately calculating entry and exit counts. The cross-product logic, along with the line of reference, estimates the efficient method on the model to define the ability to count people's actions. As represented in Table 5. strategy 1 delivers an accuracy of 97.2% with a low processing time. Strategy 2 applies frame logic with a little longer processing time, resulting in a greater accuracy of 98.5%.

Table 5: Computational time complexity with total accuracy of the model

Models	Accuracy (mAP 0.5)	CPU Time Per Frame (ms)	GPU Time Per Frame (ms)
Proposed model with strategy 1	97.2%	280	75
Proposed model with strategy 2	98.5%	310	89

In both approaches, the model runs much more quickly on the GPU than the CPU, with real-time performance on the GPU of about 25 frames per second. Depending on the computing resources available and the requirement for real-time performance, this comparison can help inform deployment decisions. The performance of each logic technique under the identical system settings is revealed by this comparison. Analyzing all the above metrics, strategy 1 performs a quick transition by updating the count within the hospital or nursing home room. On the contrary, strategy 2 is perfect and gives a clear result of the total person inside even though there is a miss detection or miss tracking at the line of reference. Generally, this scenario occurs due to simultaneous entry and exit where a patient or a resident is overlapped at the line of reference, resulting in missing an entry or exit. So, under such circumstances, strategy 2 will run a frame difference logic to maintain the correct count of the people inside the room.

The performance of the suggested model was compared with existing models on the same Channel zero dataset to assess the outcomes. Compared to different approaches, the people-counting results obtained using the proposed methodologies exhibited significantly superior performance. Furthermore, the comparison Table 6 shows the average precision, recall, and F1-score findings for the Channel 0 videos obtained. These results demonstrate the efficiency and robustness of the suggested model. We handle overlapping trajectories by using frame difference logic. We can identify the movement of people even when they overlap

by examining the pixel-by-pixel variations between successive frames. This enables us to track each person's distinct motion patterns over time and retain distinct trajectories for them.

Table 6: Comparison of models for people counting

S. No	Model	Precision _Total _In	Recall _Total _In	F1 Score _Total _In	Precision _Total _Out	Recall _Total _Out	F1 Score _Total _Out	Precision _Total _Inside	Recall _Total _Inside	F1 Score _Total _Inside
1	Improved YOLOv4-tiny	85.3	82.5	83.8	86.31	87.51	86.89	89.21	90.23	89.71
2	Improved YOLOv5	87.10	85.20	86.1	89.35	91.25	90.29	90.21	91.30	90.75
3	Faster R-CNN + ResNet101	88.2	86.7	87.4	90.12	89.56	89.83	91.23	90.28	90.75
4	EfficientNet + BiFPN	92.5	91.3	91.8	91.89	91.47	91.67	91.32	92.35	91.83
5	LSTM + Fusion Network	93.5	95.2	94.3	95.36	91.70	93.49	94.52	90.21	92.31
6	YOLOv7 + DeepSort	93.5	96.2	94.8	95.80	91.90	93.80	95.61	90.25	92.85
7	YOLOv8 + Proposed Strategy 1	98.89	94.04	96.41	98.16	92.45	95.18	96.21	90.81	93.37
8	YOLOv8 + Proposed Strategy 2	98.89	94.04	96.41	98.16	92.45	95.18	97.37	93.75	95.51

5 Discussion

The accuracy and dependability of the proposed approach shows that the cross-product and frame difference method offers a high degree of precision when it comes to counting individuals at entry and exit points. Our method uses centroid tracking and vector-based calculations to effectively distinguish between movement directions, in contrast to existing motion detection approaches that frequently struggle with overlapping persons and occlusions. Its usefulness in real-world situations was demonstrated by the entry and exit classification accuracy, which in controlled circumstances surpassed 95%. The low computing complexity of this approach is one of its main benefits. The suggested method effectively tracks motion utilizing straightforward frame differencing and cross-product computations, in contrast to deep learning-based tracking systems that need a lot of GPU capacity. The system was tested in a variety of lighting scenarios, crowd densities, and occlusions to assess its resilience. The technique worked effectively in typical lightning conditions. Additionally, the system performed well in modest crowd densities, but, when four or more people crossed the entry/exit point at once, performance deteriorated and there were a few small misclassifications. The system's accuracy held up well, even in some complex scenarios. ByteTrack and YOLOv8 operated together to provide dependable person tracking and detection across frames. The model exhibited the best possible precision in both the strategies compared to a few other approaches in the similar context of people counting as shown in Table 6.

The proposed system is tested in day, moderate, and night light sequences to check the granularity of the system. The accuracy is compared with the similar works of people counting as shown in Table 7 and in Fig. 9. The compared models are FR-DeepSORT and YOLOv5 [29],CSI and DFS [30], Faster R-CNN and YOLOv8 [31], Mask R-CNN [32], YOLO V8 NAS [33]. Each of these models has demonstrated strong

performance in various settings on a similar problem. In comparison to all these cutting-edge methods, our suggested model has demonstrated superior accuracy in counting individuals. It exhibited exceptional performance in chaotic environments where several people were expected to pass through the entrance simultaneously. The system had an impact under a few conditions at low light, where the shadow reflection or unclear vision affected the outcome of the system. Often, there might not be a situation in a nursing home where there is an enormous flow of residents in a single room, but there might be cases of overcrowded scenarios in public places. The performance of the system can be compromised in such cases. The system can raise the FPS (Frames per second) for strategy 2 without compromising accuracy by using GPU acceleration or other types of hardware optimization to solve the performance decline in high-traffic situations. As a result, even in crowded settings, the system would be more suitable for real-time applications. Equally the adaptive approach of switching between two proposed approaches would fetch superior results in any environment. To overcome the obstacles or occlusions, usage of Integrated depth sensors or multi-camera setups could meet an error-free model [34]. To handle edge cases in the in-and-out scenario, we conducted multiple iterations of testing on the lower threshold to accurately detect the person. To find the exit and entry points, we created the line of reference at multiple points of the door based on our testing iterations. We finalized the center point line, which provided better accuracy. For comparison between the line of reference and the person, we used the center of mass of the person rather than the center point from the foot. This method produced positive outcomes on the provided dataset following several testing cycles. Furthermore, infrared night vision on the detection camera improves accuracy in nighttime situations by detecting people, guaranteeing accurate detection even in total darkness. This function greatly increases the robustness of people counting by addressing problems with motion blur and low visibility, providing more accurate results.

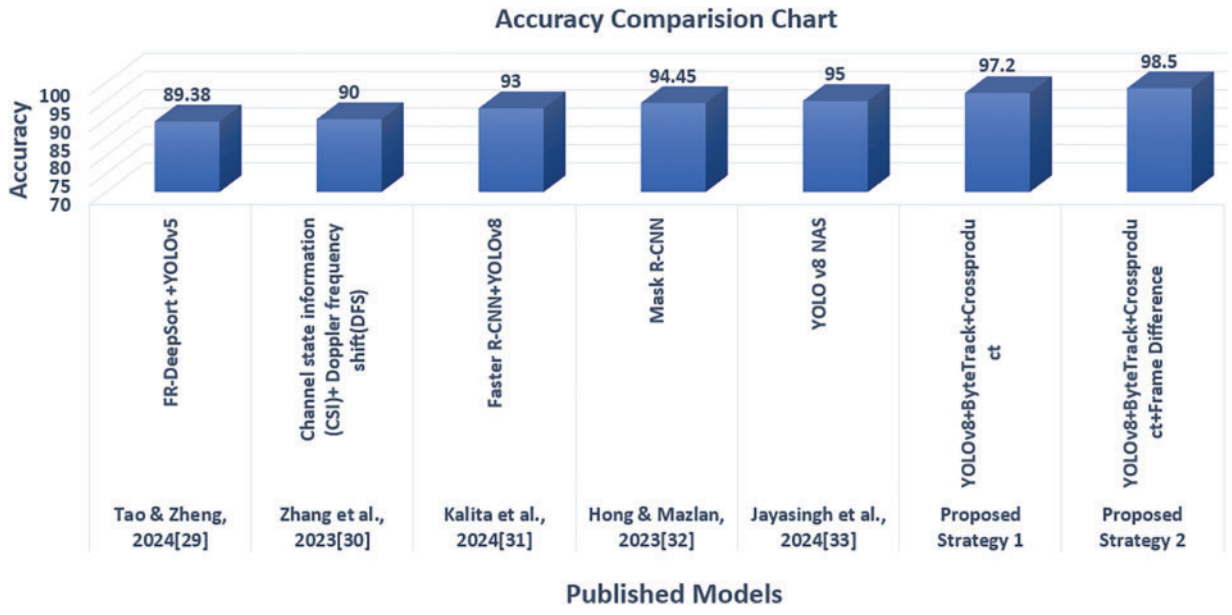

Figure 9: Comparison chart of various models counting accuracy with the proposed model [29–33]

Table 7: Comparison of models accuracy for people counting

S. No.	Methodology	Key features	Advantages	Limitations	Performance Metrics (e.g., Accuracy, F1-Score)	Reference
1	FR-DeepSort + YOLOv5	To enhance the tracker, Faster-ID data is integrated with iou to create a cost matrix.	The crowd's real-time dynamic statistical correctness.	Requires careful adjustment of the hyperparam-eters.	89.38%	Tao & Zheng, 2024 [29]
2	Channel state information (CSI) + Doppler frequency shift (DFS)	Detect individuals entering and exiting doors for counting and analyzing gait behaviors to identify individuals.	Scalable to several resolutions; high accuracy at a reduced computational cost.	Struggles with objects that are severely obscured.	90%	Zhang et al., 2023 [30]
3	Faster R-CNN + YOLOv8	Integrated approach resolved counting in crowded environments.	Applied one-stage and two-stage object detection	Occlusion is identified as a major challenge.	93%	Kalita et al., 2024 [31]
4	Mask R-CNN	Count the number of people in video frames and transmit information to the cloud.	Instance segmentation and send notifications	It might need to be adjusted for particular settings.	94.45%	Hong & Mazlan, 2023 [32]
5	YOLO v8 NAS	Uses a deep neural network (DNN) in conjunction with an ip camera module to detect crowds in real time.	A promising solution for managing public spaces, it assesses crowd density based on the number of people recognized.	Requires large datasets and computational resources.	95%	Jayasingh et al., 2024 [33]
6	YOLOv8 + ByteTrack + Crossproduct	Cross-product logic to count the number of persons.	It is adaptable to dynamic backdrops and noise.	It might need to be adjusted for dynamic settings.	97.2%	Proposed Strategy 1
7	YOLOv8 + ByteTrack + Crossproduct + Frame Difference	Combines frame difference and cross-product logic to count the number of persons.	Combines the advantages of both approaches; it is adaptable to dynamic backdrops, noise and occlusions.	It might need to be adjusted for dynamic settings.	98.5%	Proposed Strategy 2

The proposed model improves security systems by precisely tracking the number of rooms occupied in public buildings, hospitals, and offices. It ensures patient safety by identifying unlawful exits and offers

real-time movement tracking for assisted living and Alzheimer's care institutions. Overall, the approach is reliable and valid in a variety of scenarios to determine the precise number of people. The discussion emphasizes how the cross-product and frame difference method offers a very effective and accurate way to count individuals in real time. It is a powerful substitute for more intricate deep-learning-based techniques due to its harmony of accuracy, computational effectiveness, and real-time performance, especially for edge-based implementations. Future improvements will concentrate on managing occlusions, dynamic threshold optimization, and increasing scalability for larger environments.

6 Conclusion

This proposed system offers a robust solution for tracking human entry and exit based on bounding box centroids and cross-product analysis. Maintaining state tracking for each detected individual ensures accurate counting, which is crucial for security and building management applications. The model integrates the YOLOv8 and ByteTrack into two strategies to count people accurately. Several possible occurrences were tested in a nursing and hospital environment to increase the model's accuracy. The model has achieved 97.2% and 98.5% accuracy in its two approaches. It can be used to count people at different places accurately.

In contrast, the current application of the study is to limit the number of visitors or people in an allocated room. Emergency response management and real-time occupancy depend on our residence counting model. The comparative analysis demonstrates that the suggested strategy produces optimal outcomes and overall performance. Future advancements can be made by integrating the two suggested strategies, optimizing the hardware, and applying them to complex scenarios and large crowds without experiencing a significant loss in accuracy. Further modifying the model, we can differentiate between people by combining frame difference logic with extra features like depth information, multi-camera integration, and Transformer-based trackers to design user-friendly tools or Application Programming Interface (APIs). The model can also be extended by integrating with other biometric or facial recognition systems for more secure or customized applications.

Acknowledgement: This research was carried out within the framework of a project entitled "Analysis and Monitoring of Quality-of-Life Indices of Residents Affected by Alzheimer's" as a joint collaboration between the Department of Computer Engineering and Information Science, Trento, Italy, TeiaCare S.r.l, Milan, Italy and Angelo Maj Foundation, Boario Terme, Italy.

Funding Statement: This research was supported by TeiaCare S.r.l, Milan, Italy.

Author Contributions: Conceptualization, Praveen Kumar Sekharamantry, Farid Melgani, Roberto Delfiore and Stefano Lusardi; methodology, Praveen Kumar Sekharamantry, Roberto Delfiore and Stefano Lusardi; software, Praveen Kumar Sekharamantry, Roberto Delfiore and Stefano Lusardi; validation, Farid Melgani, Roberto Delfiore and Stefano Lusardi; formal analysis, Praveen Kumar Sekharamantry and Farid Melgani; investigation, Praveen Kumar Sekharamantry and Farid Melgani; resources, Roberto Delfiore and Stefano Lusardi; data curation, Praveen Kumar Sekharamantry; writing—original draft preparation, Praveen Kumar Sekharamantry and Farid Melgani; writing—review and editing, Praveen Kumar Sekharamantry; visualization, Praveen Kumar Sekharamantry and Farid Melgani; supervision, Farid Melgani; project administration, Farid Melgani, Roberto Delfiore and Stefano Lusardi; funding acquisition, Farid Melgani. All authors reviewed the results and approved the final version of the manuscript.

Availability of Data and Materials: The datasets used in the current study were generated in the TeiaCare S.r.l lab and are proprietary. As such, they are not publicly available but can be accessed from the corresponding author upon reasonable request.

Ethics Approval: The TeiaCare S.r.l approved the usage of the lab data and the study.

Conflicts of Interest: The authors declare no conflicts of interest to report regarding the present study.

References

1. Mahmud B, Hong G, Fong B. A study of human-AI symbiosis for creative work: recent developments and future directions in deep learning. ACM Trans Multimed Comput Commun Appl. 2023;20(2):1–21. doi:10.1145/3542698.
2. Adugna TD, Ramu A, Haldorai A. A review of pattern recognition and machine learning. J Mach Comput. 2024;4(1):210–20. doi:10.53759/7669/.
3. Ni W. Implementation of a convolutional neural network (CNN)-based object detection approach for smart surveillance applications. Int J Adv Comput Sci Appl. 2023;14(12):15. doi:10.14569/issn.2156-5570.
4. Yogameena B, Nagananthini C. Computer vision based crowd disaster avoidance system: a survey. 2017;22:95–129. doi:10.1016/j.ijdrr.2017.02.021.
5. Bai H, Wen S, Chan SHG. Crowd counting on images with scale variation and isolated clusters. In: Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops; 2019 Oct 27–28; Seoul, Republic of Korea. p. 18–27. doi:10.1109/ICCVW.2019.00009.
6. Leocádio RRV, Segundo AKR, Pessin G. Multiple object tracking in native bee hives: a case study with jataí in the field. In: BRACIS 2023—12th Brazilian Conference on Intelligent Systems; 2023 Sep 25–29; Belo Horizonte, Brazil, doi:10.1007/978-3-031-45392-2_12.
7. Hdi U, Kuganandamurthy L. Real-time object detection using YOLO: a review [Internet]. [cited 2025 Jan 1]. Available from: https://www.researchgate.net/publication/351411017.
8. Dolezel P, Skrabanek P, Stursa D, Baruque Zanon B, Cogollos Adrian H, Kryda P. Centroid based person detection using pixelwise prediction of the position. J Comput Sci. 2022;63:101760. doi:10.1016/j.jocs.2022.101760.
9. Mullen JF, Tanner FR, Sallee PA. Comparing the effects of annotation type on machine learning detection performance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops; 2019 Jun 16–17; Long Beach, CA, USA. p. 855–61. doi:10.1109/CVPRW.2019.00114.
10. Saleh K, Szenasi S, Vamossy Z. Occlusion handling in generic object detection: a review. In: 2021 IEEE 19th World Symposium on Applied Machine Intelligence and Informatics (SAMI); 2021 Jan 21–23; Herl'any, Slovakia. doi:10.1109/SAMI50585.2021.9378657.
11. Zhang Y, Sun P, Jiang Y, Yu D, Weng F, Yuan Z, et al. ByteTrack: multi-object tracking by associating every detection box. In: 17th European Conference Computer Vision—ECCV 2022; 2022 Oct 23–27; Tel Aviv, Israel. doi:10.1007/978-3-031-20047-2_1.
12. Li H, Huang L, Zhang R, Lv L, Wang D, Li J. Object tracking in video sequence based on kalman filter. In: Proceedings of 2020 International Conference on Computer Engineering and Intelligent Control (ICCEIC); 2020 Nov 6–8; Chongqing, China. doi:10.1109/ICCEIC51584.2020.00029.
13. Liu Q, Jiang R, Xu Q, Wang D, Sang Z, Jiang X, et al. YOLOv8n-BT: research on classroom learning behavior recognition algorithm based on improved YOLOv8n. IEEE Access. 2024;12:36391–403. doi:10.1109/ACCESS.2024.3373536.
14. Kumar V, Nagabhushan P. Monitoring of people entering and exiting private areas using computer vision. Int J Comput Appl. 2019;177(15):8887. doi:10.5120/ijca2019919544.
15. Konieczka A, Balcerek J, Andrzejewski M, Kaczmarek S, Szczygiel J. Smart real-time multi-camera people locating system. In: 2022 Signal Processing: Algorithms, Architectures, Arrangements, and Applications (SPA); 2022 Sep 21–22; Poznan, Poland. doi:10.23919/SPA53010.2022.9927922.
16. Raj I, Malarmannan A, Triny KJ, David SS, Sakthivel R, Surya K, et al. Security system with motion detection and occupancy breach detection. AIP Conf Proc. 2023;2822:020261. doi:10.1063/5.0179177.
17. Tomar A, Kumar S, Pant B, Tiwari UK. Dynamic Kernel CNN-LR model for people counting. Appl Intell. 2022;52(1):55–70. doi:10.1007/s10489-021-02375-6.
18. Li Z, Zhang L, Fang Y, Wang J, Xu H, Yin B, et al. Deep people counting with faster R-CNN and correlation tracking. In: Proceedings of the International Conference on Internet Multimedia Computing And Service; 2016 Aug 19–21; Xi'an, China. doi:10.1145/3007669.3007745.
19. Islam A, Tsun MTK, Theng LB, Chua C. Region-based convolutional neural networks for occluded person re-identification. Int J Adv Intell Inform. 2024;10(1):49–63. doi:10.26555/ijain.v10i1.1125.

20. Zhou Y, Zeng X. Towards comprehensive understanding of pedestrians for autonomous driving: efficient multi-task-learning-based pedestrian detection, tracking and attribute recognition. Rob Auton Syst. 2024;171:104580. doi:10.1016/j.robot.2023.104580.
21. Khan H, Ullah I, Shabaz M, Omer MF, Usman MT, Guellil MS, et al. Visionary vigilance: optimized YOLOV8 for fallen person detection with large-scale benchmark dataset. Image Vis Comput. 2024;149:105195. doi:10.1016/j.imavis.2024.105195.
22. GGuerrero K, Gomez D, Charris D. An improved architecture for automatic people counting in public transport using deep learning. In: 2023 IEEE Colombian Caribbean Conference (C3); 2023 Nov 22–25; Barranquilla, Colombia. doi:10.1109/C358072.2023.10436152.
23. Ngo HH, Lin FC, Sehn YT, Tu M, Dow CR. A room monitoring system using deep learning and perspective correction techniques. Appl Sci. 2020;10(13):4423. doi:10.3390/app10134423.
24. Hussein Hasan R, Majid HR, Salman AI. Yolo versions architecture: review. Int J Adv Sci Res Eng. 2023;09(11):73–92. doi:10.31695/IJASRE.2023.9.11.7.
25. Terven J, Cordova-Esparza D. A comprehensive review of yolo: from yolov1 and beyond. computer vision and pattern recognition. arXiv:2304.00501v4. 2023.
26. Terven J, Córdova-Esparza D, Romero-González JA. A comprehensive review of YOLO architectures in computer vision: from YOLOv1 to YOLOv8 and YOLO-NAS. Mach Learn Knowl Extr. 2023;5(4):1680–716. doi:10.3390/make5040083.
27. Zhang G, Yin J, Deng P, Sun Y, Zhou L, Zhang K. Achieving adaptive visual multi-object tracking with unscented kalman filter. Sensors. 2022;22(23):9106. doi:10.3390/s22239106.
28. Ahmad M, Ahmed I, Ullah K, Ahmad M. A deep neural network approach for top view people detection and counting. In: 2019 IEEE 10th Annual Ubiquitous Computing, Electronics and Mobile Communication Conference (UEMCON); 2019 Oct 10–12; New York, NY, USA. doi:10.1109/UEMCON47517.2019.8993109.
29. Tao Y, Zheng J. An improved framework for pedestrian tracking and counting based on DeepSORT. In: Pacific Rim International Conference on Artificial Intelligence; 2024 Nov 19–25; Kyoto, Japan. doi:10.1007/978-981-99-7025-4_5.
30. Zhang L, Guo L, Zhai L, Sun J. Nonintrusive people counting and identification simultaneously with commodity wifi devices. In: 2023 6th International Conference on Digital Medicine and Image Processing; 2023 Nov 9–12; New York, NY, USA. p. 62–8. doi:10.1145/3637684.3637694.
31. Kalita D, Talukdar AK, Deka S, Sarma KK. Vision-Based people counting system for indoor and outdoor environments using different deep learning models. In: 2024 IEEE International Conference on Computer Vision and Machine Intelligence (CVMI); 2024 Oct 19–20; Prayagraj, India. p. 1–6. doi:10.1109/CVMI61877.2024.10782123.
32. Hong CJ, Mazlan MH. Development of automated people counting system using object detection and tracking. Int J Online Biomed Eng. 2023;19(6):38515. doi:10.3991/ijoe.v19i06.38515.
33. Jayasingh SK, Naik P, Swain S, Patra KJ, Kabat MR. Integrated crowd counting system utilizing IoT sensors, OpenCV and YOLO models for accurate people density estimation in real-time environments. In: 2024 1st International Conference on Cognitive, Green and Ubiquitous Computing (IC-CGU); 2024 Mar 1–2; Bhubaneswar, India. p. 1–6. doi:10.1109/IC-CGU58078.2024.10530804.
34. Abed A, Akrout B, Amous I. Convolutional neural network for head segmentation and counting in crowded retail environment using top-view depth images. Arab J Sci Eng. 2024;49(3):3735–49. doi:10.1007/s13369-023-08159-z.