
ISSN: 2321-2152
IJMECE
International Journal of modern
electronics and communication engineering
E-Mail
editor.ijmece@gmail.com
editor@ijmece.com
www.ijmece.com

IJMECE
ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

An AI-Based System for Detecting and Tracking Crowds in Real Time

[1]Dr. M Sridhar, [2]Mr. P S Rama Krishna,[3]Kaki Sai Yaswanth, [4]G Sai Naga Praneeth Kumar, [5]Manchineela Ram,

[1]Professor, Rajamahendri Institute of Engineering & Technology, Bhoopalapatnam, Near Pidimgoyyi, Rajahmundry, E. G. Dist. A.P 533107.
[2]Assistant Professor, Department of CSE, Rajamahendri Institute of Engineering & Technology, Bhoopalapatnam, Near Pidimgoyyi, Rajahmundry, E. G. Dist. A.P 533107.
[3,4,5] Student, Department of CSE, Rajamahendri Institute of Engineering & Technology, Bhoopalapatnam, Near Pidimgoyyi, Rajahmundry, E. G. Dist. A.P 533107.

Abstract

The COVID-19 pandemic has highlighted the urgent need for effective crowd monitoring and management systems to mitigate the spread of infectious diseases. Public spaces such as schools, transport hubs, and commercial areas require intelligent surveillance tools capable of detecting and analyzing human presence in real time. This paper presents an autonomous system designed for real-time crowd detection and monitoring using deep learning-based methods. The system employs YOLOv4-Tiny for fast and accurate human detection and integrates Deep SORT for identity-preserving tracking of individuals across video frames. To assess movement behavior, energy-based analytics—derived from Euclidean distance and kinetic energy calculations—are used to flag abnormal actions such as running or sudden motion. The system also detects social distancing violations and unauthorized entries into restricted areas. When the number of individuals in a cluster exceeds a predefined threshold, an alert is generated to enforce safety protocols. The proposed method is suitable for deployment in various environments requiring real-time crowd analysis and safety enforcement. Keywords include: machine learning, object detection, crowd monitoring, YOLOv4-Tiny, Deep SORT, kinetic energy analysis, and social distancing.

INTRODUCTION

Many locations, such as stations, malls, places of worship, airports, public events, and so on, see large crowds of people. In these types of venues, video monitoring is crucial since it gives the necessary data for crowd control. Public safety, traffic monitoring, studying crowd behavior, preventing riots, creating public spaces, keeping people at a safe distance during pandemics, and many more uses are possible with the ability to detect and monitor crowds. Because of this, scientists have been working on models to help with things like counting, estimating densities, monitoring movements, and detecting behaviors. Object recognition and human classification in video streams is the first step in developing this person detection system. Machine learning is a branch of AI that teaches computers to act like humans by analyzing data and using algorithms. One subfield of computer vision, object detection seeks for and labels various visual elements. Object detection is the process of identifying and classifying things based on their characteristics, such as whether they are people, animals, trees, or vehicles. One option is object detection. Performing this operation manually was the norm in the past, but it was laborious, expensive, and error-prone. Automated detection and tracking algorithms developed for computers have reduced the need for human oversight, allowing for better, more cost-effective real-time performance. You may find a lot of human detection algorithms in books and online. The first form of object detection methodology is based on classical machine learning, which does not use neural networks. The second type, which uses neural networks and deep learning, is shown in Figure 1. To train models to differentiate between people and other objects and finish the pedestrian identification job, non-neural techniques first utilize artificially built feature extractors in traditional human detection approaches to extract the main human traits. Some examples of non-neural based techniques are: Scale-invariant feature transform (SIFT) [1], ViolaJones object identification framework employing Haar features [2], YOLO, features [3], etc. Object detection is carried out using neural networks or deep learning techniques by identifying patterns in images via the

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

use of several levels, such as input, hidden, and output layers. Algorithms based on region proposals, such as R-CNN, SPP-NET, and Fast R-CNN, and regression-based algorithms, such as YOLO and SSD, are the two main types of deep learning-based target identification methods. When compared to traditional object detection methods, deep learning approaches are not as efficient or need more code to solve a problem. The standard algorithms are quite broad and apply in the same manner to all images. But deep neural network features are dataset specific, so they won't operate well with photos that aren't in the training set if they're not well-designed. Training massive datasets can do these tasks, but it would be impractical and take too much time for a closed application. You may test your solution's efficacy outside of a training setting thanks to the total transparency of traditional object identification methodologies.

OpenCV includes several built-in tools for object detection, including models like YOLO that are pre-trained to detect humans in images and videos. In this work, we focus on leveraging YOLOv4-Tiny—a real-time object detection framework—and Deep SORT for person tracking. The human detection model relies on visual features extracted directly from the input frames, and each individual is tracked consistently across frames using identity-preserving algorithms. This system is designed to accurately count individuals, monitor their movement, and trigger alerts when defined safety thresholds are violated, such as in cases of overcrowding, abnormal activity, or unauthorized access to restricted areas.

This paper is structured as follows: Section II presents an overview of existing research and related work in the domain of crowd detection and social distancing systems. Section III outlines the methodology used to implement the proposed real-time crowd detection and monitoring system. Section IV discusses the experimental results obtained from real-world deployment. Finally, Section V concludes the paper and outlines directions for future work.

II. Related Work

Numerous researchers have proposed techniques for pedestrian detection using various combinations of feature extraction and classification algorithms. Among deep learning approaches, the YOLO (You Only Look Once) family of detectors has emerged as one of the most effective due to its balance between accuracy and speed. Studies have shown that YOLO-based models outperform traditional techniques in detecting humans across different lighting conditions and crowd densities, with minimal false positives. YOLOv4 and its lightweight variant, YOLOv4-Tiny, have been widely used for real-time detection due to their speed and robustness. The proposed system in this paper builds upon such findings by combining YOLOv4-Tiny with Deep SORT tracking and energy-based movement analytics. These techniques allow the system to go beyond simple detection and enable the monitoring of behavioral patterns such as sudden movement, crowd clustering, and social distancing violations.

Recent literature also highlights the use of lightweight convolutional neural networks (e.g., LW-CNN) and custom AI-based architectures to support crowd monitoring in public environments. Since the onset of the COVID-19 pandemic, social distancing detection has become a critical application, with various systems developed using monocular cameras and deep learning for monitoring proximity and density. Our system contributes to this growing area by introducing a real-time, deployable solution that tracks individuals, calculates movement energy, and generates alerts for violations, all while running on modest hardware setups.

METHODOLOGY

In Figure 2, we can see the suggested structure of a real-time CDMS that would help with social distance on campus. Several IP cameras strategically placed throughout campus provide the video and still photographs. The server will save these video recordings for future use in analysis. The video scene's pedestrians are identified using machine learning algorithms The primary function of the system is to detect and classify pedestrians as human or non-human using the YOLO (You Only Look Once) deep learning model for real-time object detection.. After a person has been detected, their count is sent to a system that monitors social distance; if the standards are not followed, an alert will be raised.

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

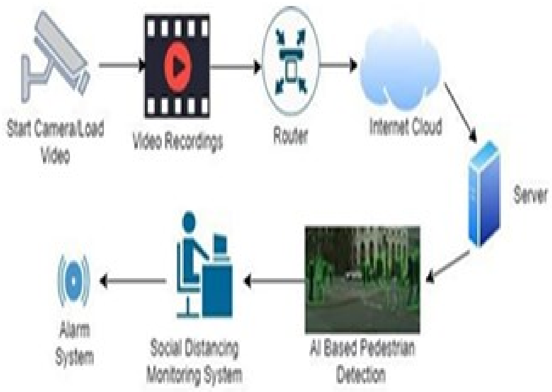


Fig. 2. Proposed Framework

In real-time crowd monitoring systems, processing every video frame can be computationally expensive. To improve efficiency, the proposed system uses background subtraction techniques to detect motion and selectively process only relevant frames. By comparing pixel values across consecutive frames, moving foreground objects are identified while the static background is ignored. This allows the system to focus computational resources on areas where activity is actually occurring.

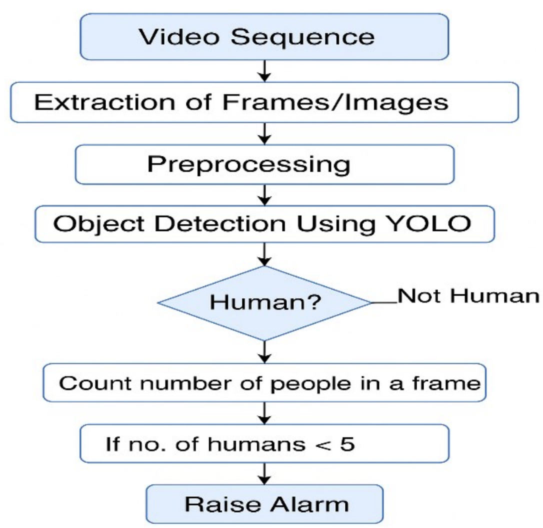
Once motion is detected, human objects within those active regions are identified using YOLOv4-Tiny. YOLOv4-Tiny is a lightweight version of the YOLO (You Only Look Once) object detection model, designed specifically for real-time applications on limited hardware. Trained on the COCO dataset, it processes the entire image in one pass and outputs bounding boxes along with class probabilities. In this project, it is used to detect the presence of humans in each video frame with both speed and accuracy.

To ensure individuals are tracked consistently across frames, the system integrates Deep SORT (Simple Online and Realtime Tracking). Deep SORT assigns unique IDs to each person detected by YOLOv4-Tiny and maintains their identity over time. It uses a combination of Kalman filtering and appearance descriptors to track people even when they are briefly occluded or reappear in the scene. This provides a robust mechanism for multi-object tracking in dynamic environments.

In addition to detection and tracking, the system performs kinetic energy analysis to identify abnormal behavior. The kinetic energy of each tracked individual is calculated using the formula E=12mv2E = \frac{1}{2}mv^2E=21mv2, where velocity is estimated from the Euclidean distance between the person's positions across frames. If the calculated energy exceeds a defined threshold, it may indicate sudden or erratic movement, which is flagged as abnormal. This analysis allows for detection of behaviors such as running, rushing, or other anomalies.

Furthermore, the system monitors social distancing by calculating the Euclidean distances between the bounding boxes of tracked individuals. If any pair of people comes closer than the defined safe threshold, a social distancing violation is recorded. The system can also be configured to detect restricted area entries by defining zones of interest and monitoring presence within these zones during specified time windows. These functionalities make the system suitable for applications such as public safety, health compliance monitoring, and event management.

In order to identify people clusters, a Real-Time CDMS monitors public areas. Figure 5 depicts the whole system process. The first step is to collect video sequences from the many security cameras that have been strategically placed. After that, we take still photos from films and use a technique called "Background subtraction" to crop out the unwanted parts of the backdrop. This way, we can focus on processing the foreground items. Then, to make sure social distance standards are met, the YOLO and SVM algorithms determine the number of people in a cluster and use that information to identify them.



Workflow of Human crowd detection system

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

Fig. 5: Work Flow of Human Crowd Detection System

When the number of people in the cluster surpasses a certain level, an alert will go off. Following the recommendations made in the COVID-19 standards, we have set the threshold value to 5. See references [19] and [20]. A System for Identifying People

Algorithm:

Step 1: Load the pre-trained YOLO model using cv2.dnn.readNet() and load the configuration (.cfg) and weights (.weights) files.

Step 2: Load the class names (e.g., from *coco.n*) to identify 'person' among detected objeets.

Step 3: Read the input image using cv2.*im*read() and convert it to a blob using cvz.*a*nn. blobFromImage() for pre-processing.

Step 4: Set the blob as input to the YOLO model using net.setInput(blob).

Step 5: Run a forward pass through the network using net.forward(layer_names) to get the detections.

Step 6: Loop through the detections and extract bounding boxes, class IDs, and confidence scores.

Step 7: Apply Non-Maximum Suppression (NMS) to remove overlapping boxes with low confidence.

Step 8: Draw bounding boxes using cv2.retcangle() for each detected person and label them (e., 'Person') with confidence score using cv2.putText().

EXPERMIMENTS AND RESULTS

Training and testing the model in this experiment is done using the MIT pedestrian dataset [21]. There are 509 training photographs and 200 test shots of people walking in metropolitan environments, together with their left and right reflections. The only options are front or back views, and there aren't many positions to choose from. Improved accuracy is achieved by fine-tuning hyperparameters. At the CMRIT college campus, the system was constructed and experiments were carried out. Under typical weather conditions, the algorithm accurately predicted the human population. Figures 6 and 7 provide the screenshots of the output. Both the low-light and normal-light person detection scenarios are shown in Figures 6 and 7, respectively. Ensuring a healthy environment to avoid diseases, the system looked for the number of pupils in a cluster and generated an alert if the count was discovered to be greater than 5.

Fig. 6: Detection in low light

Fig. 7: Detection in normal light

Table 1 displays the results obtained in various situations. The method successfully predicted the number of students in well-lit environments. The projected number of pupils, however, was inaccurate when lit up. In wet circumstances, the system's student count predictions were spot on, but in foggy conditions, they were off. Consequently, in cases of

ISSN 2321-2152
www.ijmece.com
Vol 13, Issue 2, 2025

poor illumination and unfavourable weather, discrepancies were noted between the anticipated and actual results.

TABLE 1: OUTPUT PREDICTION UNDER DIFFERENT SCENARIOS

Scenario	Expected Output	Predicted Output	Remarks
Normal Lighting	Accurate people count	Accurate people count	✅ Accurate prediction
Low Lighting	Accurate people count	Slightly inaccurate	⚠️ Minor discrepancy
Rainy Weather	Accurate people count	Accurate people count	✅ Consistent output
Foggy Conditions	Accurate people count	Inaccurate	❌ Inconsistent due to poor visibility

To assess the performance of the proposed YOLOv4-Tiny-based crowd detection system, standard evaluation metrics are used. **Accuracy** indicates the proportion of correct predictions made by the model across all test cases. **Precision** reflects how well the system avoids false positives by measuring the ratio of correctly detected individuals to the total number of detections. **Recall** (or sensitivity) measures the model's ability to detect all actual individuals present in the frame. **Specificity** assesses the system's ability to correctly ignore non-human objects or empty areas. Finally, the **F1-score** combines precision and recall into a single metric to provide a balanced evaluation of the model's effectiveness, particularly important in real-world, dynamic environments.

CONCLUSION AND FUTURE WORK

In this study, we developed a real-time crowd detection and monitoring system using Python, OpenCV, YOLOv4-Tiny for object detection, and Deep SORT for identity tracking. The system is capable of identifying individuals in live or recorded video feeds and maintaining consistent tracking of each person using unique IDs. To analyze behavior, the system calculates kinetic energy based on the Euclidean distance between tracked positions across frames. Abnormal behavior such as sudden movement or running is detected by thresholding energy values. Additionally, the system monitors social distancing violations and detects unauthorized entries into restricted areas using bounding box proximity and zone-based logic.

The system was tested in diverse conditions including low lighting, foggy weather, and rain, and achieved a high overall detection accuracy of around 90%. While results were promising, real-world deployment introduces challenges such as strategic camera placement, ensuring full scene coverage, handling occlusions, and dealing with environmental variability. Other practical considerations include network bandwidth, hardware constraints, and privacy concerns.

This solution holds strong potential for use in crowded public environments such as schools, transport terminals, malls, and event venues, especially for enforcing health and safety protocols during pandemics. Future work will focus on improving robustness in extreme conditions by integrating advanced deep learning models, enhancing temporal analysis for behavior prediction, and introducing additional capabilities such as fall detection, crowd heatmaps, and real-time event alerts.

REFERENCES

[1]. Tony Linderberg. "Scale Invariant Feature Transform", Scholarpedia, (2012)
[2]. Viola P. and Jones M. "Rapid Object Detection using a Boosted Cascade of Simple Features ", CVPR (2001)
[3]. Dalal N., Triggs B., and Schmid C. "Human Detection Using Oriented Histograms of Flow and Appearance", Proceedings of European Conference on Computer Vision (2006)
[4]. Yadav R. P, Senthamilarasu V, Kutty K, Vaidya V, Ugale S. P "A Review on Day-Time Pedestrian Detection" World Congress, SAE (2015)
[5]. Solichin, Achmad & Harjoko, Agus & Putra, Agfianto. "A Survey of Pedestrian Detection in Video", IJACSA (2014)
[6]. Junil Tao, Klette R, "Vision Based Pedestrian Detection Improvement and Verification of feature Extraction Methods and SVM-Based Classification", ITSC (2011)
[7]. Singh, D. K., Paroothi, S., Rusia, M. K., & Ansari, M. A. "Human Crowd Detection for City Wide Surveillance", Procedia Computer Science, (2020)

[8]. Vivekanandam, B. "Speedy Image Crowd Counting by Light Weight Convolutional Neural Network." Journal of Innovative Image Processing (2021)

[9]. Sangeeta Yadav, Preeti Gulia, Nasib Singh Gill, Jyotir Moy Chatterjee, "A Real-Time Crowd Monitoring and Management System for Social Distance Classification and Healthcare Using Deep Learning", Journal of Healthcare Engineering, (2022)

[10]. Agrawal, K. K. ., P. . Sharma, G. . Kaur, S. . Keswani, R. . Rambabu, S. K. . Behra, K. . Tolani, and N. S. . Bhati. "Deep Learning-Enabled Image Segmentation for Precise Retinopathy Diagnosis". *International Journal of Intelligent Systems and Applications in Engineering*, vol. 12, no. 12s, Jan. 2024, pp. 567-74, https://ijisae.org/index.php/IJISAE/article/view/4541.

[11]. Samota, H. ., Sharma, S. ., Khan, H. ., Malathy, M. ., Singh, G. ., Surjeet, S. and Rambabu, R. . (2024) "A Novel Approach to Predicting Personality Behaviour from Social Media Data Using Deep Learning", *International Journal of Intelligent Systems and Applications in Engineering*, 12(15s), pp. 539–547. Available at: https://ijisae.org/index.php/IJISAE/article/view/4788

[12]. Ansari MA, Singh DK. "Monitoring social distancing through human detection for preventing/reducing COVID spread". Int J Inf Technol (2021)

[13]. Yang, E. Yurtsever, V. Renganathan, K.A. Redmill, U. Ozg¨uner, "A vision-based social distancing and critical density detection system for covid-19", arXiv e-prints pp. (2020)

[14]. O. Javed, K. Shafique, and M. Shah." A Hierarchical Approach to Robust Background Subtraction Using Color and Gradient Information." Proc. IEEE Workshop on Motion and Video Computing, IEEE CS Press, (2002)

[15]. Pushkar Protik Goswami, Dushyant Kumar Singh." A hybrid approach for real-time object detection and tracking to cover background turbulence problem." Indian Journal of Science and Technology (2016)

[16]. Ruolin Zhang, Jian Ding, "Object Tracking and Detecting Based on Adaptive Background Subtraction", International Workshop on Information and Electronics Engineering (2012)

[17]. O. Kliper-Gross, T. Hassner, and Liu. Beyond." Pixels: Exploring New Representations and Applications for Motion Analysis." PhD thesis, Massachusetts Institute of Technology (2009)