

# Enhancing Human Detection and Counting Using a YOLO V8 Algorithm and Haar Cascade Refinement

**Nilima Kulkarni**

Department of Computer Science  
and Engineering, MIT School of Engineering, MIT  
Arts Design and Technology University,  
Pune, 412201, India  
[nilima.kulkarni@mituniversity.edu.in](mailto:nilima.kulkarni@mituniversity.edu.in)

**Siri Rao**

Department of Computer Science  
and Engineering, MIT School of Engineering, MIT  
Arts Design and Technology University,  
Pune, 412201, India  
[siri.rao03@gmail.com](mailto:siri.rao03@gmail.com)

**Shruti Karpe**

Department of Computer Science  
and Engineering, MIT School of Engineering, MIT  
Arts Design and Technology University,  
Pune, 412201, India  
[karpeshruti0@gmail.com](mailto:karpeshruti0@gmail.com)

**Sahil Chalke**

Department of Computer Science  
and Engineering, MIT School of Engineering, MIT  
Arts Design and Technology University,  
Pune, 412201, India  
[sahilchalke1990@gmail.com](mailto:sahilchalke1990@gmail.com)

**Anushka Singh**

Department of Computer Science  
and Engineering, MIT School of Engineering,  
MIT Arts Design and Technology University,  
Pune, 412201, India  
[anushkarishusingh2003@gmail.com](mailto:anushkarishusingh2003@gmail.com)

## **Abstract:**

*The present study introduces a novel methodology to tackle the issue of detecting human counts within a specific frame. This study's primary goal is to count the number of people by updating the popular YOLO (You Only Look Once) V8 algorithm, which primarily focuses on object recognition. By modifying YOLO to concentrate on human objects, the suggested study aims to increase the precision of counting the number of humans within a picture. Additionally, the Haar Cascade refinement approach is used, which significantly improves detection accuracy—particularly in busy locations. This work's approach makes great use of Yolo's capabilities to count and identify people. YOLO can identify human counts quickly and correctly because it scans the entire image in a single forward pass, which makes it a good option for real-time applications. As an extra technique, the algorithm's efficiency is improved, and more precise counting is made possible by incorporating the use of Haar Cascade refinement, especially when dealing with densely packed crowds. Real inputs are employed in addition to datasets to assess the suggested system, and the results show promise for the proposed system.*

**Keywords:** -Haar cascade, YOLO

## I. Introduction

Human count detection is a significant technology with several uses, including crowd surveillance, public safety and security, and area capacity

control. It primarily focuses on tracking and evaluating people's movements in crowded areas. Safety systems, necessary in places like emergency exits and locations with maximum occupancy, are still another crucial use. Conventional counting techniques suffer from inaccuracies and inefficiencies when trying to precisely count the number of people in congested and complicated locations.

The regrettable bridge collapse in Gujarat that was a result of crowding served as the impetus for our research on human count detection. Addressing concerns about public safety in crowded areas was crucial. After witnessing the disastrous consequences of such circumstances, it created an application to act as a preventative mechanism for human count identification within a specific time frame. Our goal is to use cutting-edge computer vision techniques, particularly the YOLO algorithm modified for human count identification and the Haar Cascade refining approach, to help prevent problems associated with overpopulation.

## II. Literature survey

To determine the present state of the art, a thorough survey in the relevant domain has been conducted. Here, a portion of the significant work has been discussed.

The authors of [1] demonstrate how the normalized cross-correlation algorithm can identify people based on their facial traits as well as their skin tone, with an average high-count accuracy of 80% in congested areas. It also lists the mistakes made when people cover up their faces with facial hair or masks. The method used in the paper [2] is

centered on counting the number of pills arranged in a tray and is based on deep learning-based counting. They made use of a unique dataset that included numerous pictures of medications. The authors reported 90% accuracy for a cluttered tray. The authors also stated that the error only happened when there was medication overlap on the tray. In [3], the deep learning-based technique for human detection is described. The authors reported an 80% accuracy rate; however, as radar was utilized for detection, errors could have resulted from low brightness or the presence of several barriers. The authors in their paper [4, 5] identify and follow people using the YOLO method. It accurately detects people in a given frame with a high accuracy rate of 92%, albeit with some overlap anomalies. In their publication, the authors use the SOM algorithm [6]. The technique can be applied to low- to medium-density crowds for tracking and counting people. This approach can also be applied with an accuracy of 86% to 95%. The model in the paper [7] identifies an activity in a single frame after a predetermined amount of time, assisting the authors in identifying and detecting humans in real-world scenarios using surveillance cameras. To get better performance, the authors of the paper [8] use linear SVM-based human detection as a test case for human recognition, utilizing the current collection of human attributes. The authors have introduced PairNet, an efficient, robust, and generalized object detection system based on deep learning that was reported in [13]. PairNet may also be expanded for human detection. In [14], a comparison between YOLO V2 and V3 is conducted. The authors also suggest revised architectures for YOLO V3. They have demonstrated the model's effectiveness on low-processing-power devices. YOLO V8 is utilized in the proposed work for human detection.

### III. Proposed System

Figure 1 displays the proposed system's flowchart. The method combines YOLO object detection for a more thorough identification of the objects in each frame with Haar cascade detection for the first recognition of the human body. Real-time interaction with the video stream is enabled, and identified objects, including bounding boxes and masks, are visually shown.

For the front end of this paper, we used Flutter. The user's attendance record is stored locally using the shared\_preferences package. State management is done via the Riverpod package, which is an improved version of the popular state management tool Provider. This also uses firebase\_auth login credentials for verification.

This paper presents a system that uses the Yolov8 algorithm—specifically, YOU ONLY LOOK ONCE—as the primary detection tool and the Haar Cascade Classifier for human posture detection to

recognize and count humans. Python has been the main backend language utilized in our Visual Studio (VS Code) implementation of the backend. Here, YOLO V8 and the Haar Cascade Classifier are combined.

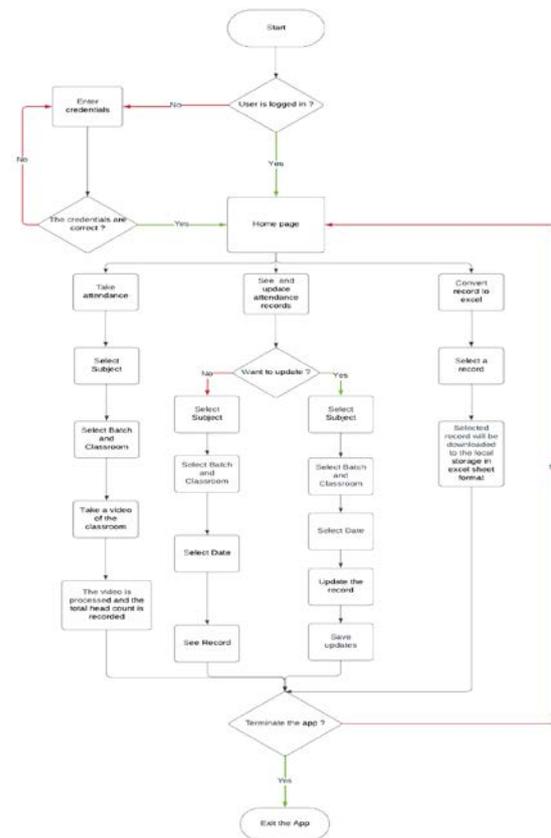


Fig 1: Flowchart of the proposed system

The blocks that make up the system are as follows: Login Page: To confirm the user's identity, the system processes the information.

Invalid Credential: The system detects if the credentials are invalid.

If the user's credentials are invalid, they will likely receive a reminder to update or retrieve their login details. The user gets taken to the home page if the credentials are genuine.

Home Page: This is the primary dashboard or menu where users may access all of the application's features.

There are three primary options available from the home page:

Take Attendance: The user may upload a video of the attendees to record attendance. After watching the film, the system counts the participants and compares that number to a list or projected number. See and Update Records: Using this option, the user can view every attendance record in addition to updating them.

Transform the record into an Excel spreadsheet: The user can utilize this function to transform

attendance records into an Excel spreadsheet format.

The most recent iteration of the ultralytics algorithm, known as YOLOv8, lacks pre-trained models and setups. As a post-processing phase, we are refining the count of false positives using the Haar cascade refinement technique. This method improves the count's accuracy in densely populated areas. With a focus on precisely counting people in densely populated areas, the application seeks to accelerate the development of sophisticated systems.

We have evaluated several models, such as YOLOv3, YOLOv4, and YOLOv7, to determine which has the best accuracy rate for recognizing and counting the number of people. After a comparative study of the data, we determined that the YOLO approach had the highest accuracy rate and was more useful for achieving our objectives; therefore, we decided to use it as our project's primary methodology. With a mean absolute error (MAE) of 1.03 and a root mean square (RMS) of 1.20, our method reliably locates and counts people even in densely populated areas. Following the YOLO procedure, we also used the Haar cascade technique to increase the outcome's precision. This methodology facilitates the enhancement of the data acquired after the video processing.

**Frontend:** This program is an intuitively designed attendance management solution. If a user is not logged in, they start by going to the login page. The system reminds users to retrieve their login credentials if it detects invalid credentials. After their authentication is successful, users are taken to the homepage. The main page acts as the focal point for every feature. From this point on, the user can record their attendance by just snapping a video of the classroom and adding the date, subject, and class. After processing the video, each person's attendance will be noted. The user also has the option to see and edit the records. With the help of this feature, the user can examine and amend attendance data in real-time. Moreover, the program has an Excel sheet conversion feature. It makes it simple for faculty members to exchange attendance information with higher authorities by enabling the user to convert the records into an Excel spreadsheet format.

**Backend:** Using a supplied dataset, we have employed the openCV package to detect and count humans in real-time. Using a pre-trained YOLOv8 and a Haar cascade classifier for full-body recognition, it applies frames to the human-like structure and processes it from a video file. It then counts the number of frames, detecting the number of humans in each frame at a time. Both the overall number of frames processed and the number of

frames in which at least one person is identified are tracked by the system.

1. **Video Capture Initialization:** To collect a series of frames, the script uses the OpenCV 'VideoCapture' class to access a video file ('in.avi').
2. **Loading a Haar Cascade Classifier:** OpenCV's 'CascadeClassifier' class is used to load a Haar cascade classifier specifically made for full-body detection ('haarcascade\_fullbody.xml'). The initial detection of human bodies is done with this classifier.
3. **Yolo Model Loading:** The script uses the Ultralytics library to load a pre-trained Yolo model ('yolov8x.pt'). An object recognition technique called YOLO is able to recognize and locate several object classes in an image.
4. **Frame Processing Loop:** To process every video frame, the script goes into a loop. It uses the 'cap.read()' function to read the following frame.
5. **OpenCV to Numpy Conversion:** Using 'np.array()' and the appropriate data type of 'np.uint8,' the frame is transformed into a NumPy array.
6. **YOLO Object Detections:** 'model(img, stream=True)' is used to apply the YOLO model to the frame. This yields a set of results with masks and bounding box coordinates for items that have been detected.
7. **Processing of the Results:** The script uses "result.plot()" to plot the results on the frames and extracts bounding box coordinates for each result it receives from the YOLO model.
8. **Visualization:** The script displays the processed frame with the YOLO detection results. "cv2.imshow("result", res\_plotted)" is being used.
9. **User Interaction:** The script exits the loop and stops processing the video if the user hits the "q" key.
10. **Cleanup:** "cv2.destroyAllWindows()" closes all OpenCV windows, and "cap.release()" releases the video capture object.

#### IV. Result

##### Application Screenshots:-

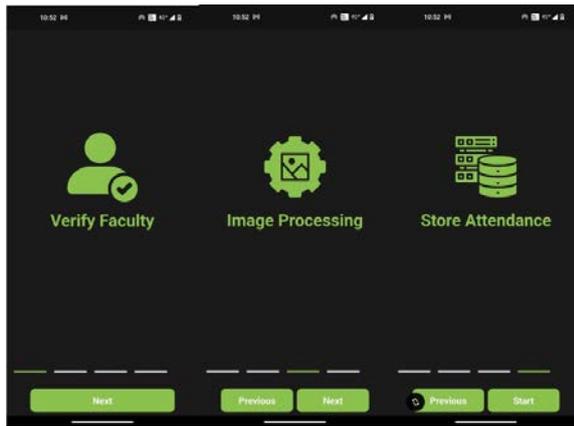


Fig2.

Fig3.

Fig4.

Figure: The preprocessing and user input procedures are depicted in Figures 2, 3, and 4 above.



Fig 9.

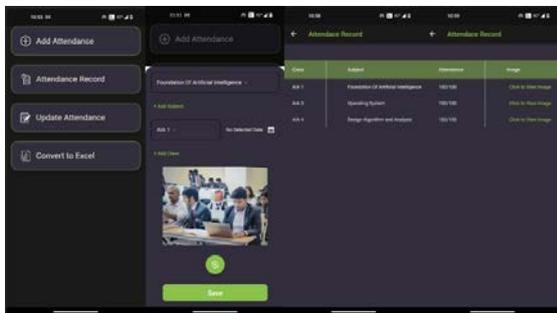


Fig 5

Fig 6

Fig 7

Fig 8

Figure: The locations and methods of data storage following the processing of the uploaded video or image are depicted in Figures 5, 6, 7, and 8.

Backend Snapshots:



Fig 10.

Figures 9 and 10 above show how the detection and count functions are based on our actual inputs.



Fig 11.

Figure: The output screen and count of a random video are displayed in Figure 11 above.

#### V. Conclusion

In conclusion, the suggested work has demonstrated the effectiveness of YOLO V8 in conjunction with Haar cascade algorithms for human detection. The application of the YOLO algorithm not only shows that it can precisely and effectively detect the human count in various settings, but it also does so accurately and efficiently. The system's low root mean square error (RMSE) and mean absolute error (MAE) across a variety of datasets highlight its applicability in many real-world scenarios. Additionally, the system design provides practical solutions for the corporate and educational sectors looking to improve their operations by identifying human counts. More accurate counting is required in sectors including urban planning, security, and retail.

There has been a significant growth in tourism in recent years, necessitating the use of apps like this one to aid in crowd control. Our project is currently limited to static data, meaning that videos must be uploaded. It is not yet a real-time application, but we plan to integrate a surveillance camera soon to continuously count the number of people in a given area at any given time. We intend to enhance the program and render it more intuitive for users by optimizing the functionality offered within.

This creative method of calculating the population of humans has a lot of promise for the future. The YOLO V8 algorithm may be further refined as technology advances, which would improve its capacity to precisely identify and count people in a variety of scenarios. This model could be improved over time and made more useful to a larger range of disciplines, such as social analytics, crowd control, and security monitoring, through current machine learning technologies and ongoing computer vision research. The suggested method's

adaptability provides a useful tool with potential applications in solving changing challenges related to human count identification in dynamic contexts.

#### VI. References

- [1] Tsong-Yi Chen, Chao-Ho Chen, Da-Jin Wang, & Yi-Li Kuo. (2010a). A people counting system based on face-detection. 2010 Fourth International Conference on Genetic and Evolutionary Computing. <https://doi.org/10.1109/icgec.2010.178>
- [2] Moon, J., Lim, S., Lee, H., Yu, S., & Lee, K.-B. (2022). Smart count system based on object detection using Deep Learning. Remote Sensing, 14(15), 3761. <https://doi.org/10.3390/rs14153761>
- [3] Qiu, C., Zhang, D., Hu, Y., Li, H., Sun, Q., & Chen, Y. (2023). Radio-assisted human detection. IEEE Transactions on Multimedia, 25, 2613–2623. <https://doi.org/10.1109/tmm.2022.3149129>
- [4] Sun Jiahui, Ge Huayong and Zhang Zhehao 2021 IEEE Advanced Information Technology, Electronic and Automation Control Conference (IAEAC) AS-YOLO: An Improved YOLOv4 based on Attention Mechanism and Squeeze Net for Person Detection
- [5] Kannadaguli Prasanth 2020 YOLO v4 Based Human Detection System Using Aerial Thermal Imaging for UAV Based Surveillance Applications IEEE Decision Aid Sciences and Application (DASA)
- [6] Pervaiz, M., Ghadi, Y. Y., Gochoo, M., Jalal, A., Kamal, S., & Kim, D.-S. (2021). A smart surveillance system for people counting and tracking using particle flow and modified som. Sustainability, 13(10), 5367. <https://doi.org/10.3390/su13105367>
- [7] Motwani, N. P., & S, S. (2023). Human activities detection using deep learning techniques-yolov8. ITM Web of Conferences, 56, 03003. <https://doi.org/10.1051/itmconf/20235603003>
- [8] Bharti R., Bhadane K., Bhadane P. and Gadhe A. 2019 Object Detection and Recognition for Blind Assistance International Research Journal of Engineering and Technology (IRJET) 06 e-ISSN : 2395-0056
- [9] Nguyen Huy Hoang, Ta Thi Nhung, Nguyen Ngoc Cuong, Bui Van Truong, Pham Hung Manh and Nguyen Duc Minh 2020 IEEE International Conference on Communications and

Electronics, ICCE YOLO Based Real-Time Human Detection for Smart Video surveillance at the Edge

[10] Zuraimi Muhammad Azhad Bin and Zaman Fadhlán Hafiz Helmi Kamaru Computer Applications & Industrial Electronics (ISCAIE), IEEE Symposium on 2021 Vehicle Detection and Tracking using YOLO and Deep SORT

[11] Ultralytics. (2024, May 8). YOLOv8. Ultralytics YOLOv8 Docs.  
<https://docs.ultralytics.com/models/yolov8/>

[12] Geeta Pandey (2024, Apr 16). Morbi bridge collapse: How India tourist spot became a bridge of death. <https://www.bbc.com/news/world-asia-india-63477292>

[13] P Kakade, A Kale, I Jawade, R Jadhav, N Kulkarni, "Optic Disc Detection using Deep Learning Techniques", International Journal of Computing and Digital Systems 11 (1), 189-197

[14] Ameya Kale, Ishan Jawade, Pratik Kakade, Rushikesh Jadhav, Nilima Kulkarni, "PairNet: a deep learning-based object detection and segmentation system", Machine Intelligence and Smart Systems: Proceedings of MISS 2021, 423-436.