CCTV-Based Occupancy Estimation on Edge Devices

Aman Prakash Munjewar
Department of Electronics and Communications
National Institute of Technology
Rourkela, 769008

Saurabh Singh
Department of Electronics and Communications
National Institute of Technology
Rourkela, 769008

Katta Tejas
Department of Electronics and Communications
National Institute of Technology
Rourkela, 769008

Nihith Venkata Madhuri
Department of Electronics and Communications
National Institute of Technology
Rourkela, 769008

Pulipati Rishikesh
Department of Electronics and Communications
National Institute of Technology
Rourkela, 769008

***Abstract*—Video-based occupancy estimation is becoming increasingly important in smart buildings, transportation hubs, and surveillance systems. Traditional sensor-based solutions (e.g., PIR, ultrasonic, and CO-based methods) often lack precision and responsiveness, especially in complex indoor environments. In contrast, deep learning computer vision approaches provide high accuracy in person detection and counting but typically require significant computational resources, making them challenging for edge deployment.**

This work introduces a lightweight CCTV-based occupancy estimation framework for resource-constrained edge devices. The system employs YOLOv5m for person detection and incorporates temporal smoothing, bounding box filtering, and region-of-interest (ROI) constraints to improve robustness in challenging indoor scenes. The framework is optimized and deployed on platforms including NVIDIA Jetson Nano, Jetson Xavier NX, and Raspberry Pi 4/5.Experiments using the Mall Dataset confirm that the proposed solution can achieve near real-time performance with practical accuracy, maintaining errors within ±1–2 occupants under normal conditions. Overall, the system demonstrates an effective balance between performance and computational efficiency, making it suitable for real-world Internet of Video Things (IoVT) applications.

***Index Terms*—Occupancy estimation, crowd counting, IoVT, edge AI, YOLOv5, Jetson Nano, embedded vision systems, smart building automation, person detection.**

I. Introduction

Human counting in internal environments has become a key competitive factor for emergent intelligent infrastructure. Continued growth of automation in Smart Buildings and Smart Cities has resulted in the creation of perpetual occupancy monitoring systems, to which several types of applications now rely upon, including:

Energy Efficiency – HVAC systems can adjust to actual occupancy;

Safety and Compliance – Prevents overcrowding in shopping malls, office buildings, theatres, and public spaces;

Security – Can help determine anomalies in crowd or unauthorized visitor gathering at specific locations;

Commercial Analytics – Assists in forecasting staffing levels, space usage, layout design for retail stores.

Additionally, with the rise of EDGE computing technology, using computer vision to make initial determinations of the "vision" used to extract people from an image can occur adjacent to (or near) the camera sensor. This reduces the need for on-the-go data transmission over networks while also protecting the confidentiality of individuals captured by each camera. This is congruent with Internet of Video Things (IoVT) where cameras not only record video but also act as smart sensors for the ultimate understanding of human behaviours.

While traditional sensing methods, including PIR, ultrasound, and Motion sensors, can accurately count individuals, there are various factors present (stationary people) that provide no accurate results or measurements when people are currently not moving. The estimation based on CO2 has a very slow response time and is influenced by air flow conditions. CO2-based estimation methods for determining occupancy are therefore indirect and not visual; however, computer vision-based methods fulfill all of the above qualification aspects.

Most of the more precise "crowd counting" computer vision models (for example, transformer-based detection architecture, encoder-decoder architecture and density map estimation networks) need to be executed on high-performance systems, thus making it impossible to deploy them at the relatively low cost EDGE devices like the Raspberry Pi or Jetson Nano difficult.

The research focuses on creating a lightweight person-counting model pipeline that employs optimized inference and efficiency strategies and includes filtering and processing mechanisms that are compatible with edge devices to address this challenge. These components, when combined, will ensure accurate counting and near real-time performance on limited-resource hardware platforms.

II. Related Work

Occupancy estimation using vision-based systems has progressed rapidly with the advancement of deep learning. Existing research can be broadly grouped into three key areas: detection-based methods, density estimation techniques, and optimization approaches targeting real-time edge deployment.

A. Traditional Occupancy Counting Sensors

Conventional systems that depend on PIR, ultrasonic, or CO sensors are adept at detecting if someone is present in a space, but do not accurately identify how many people are present. As a result of being based on methods not adapted for counting, these conventional systems also have challenges associated with things like stationary people, reflections off of walls, or the ability to respond quickly enough. The proposed method will help address these limitations by employing a vision-based approach that uses YOLOv5m technology to detect people using existing video surveillance footage. The use of this technology for real-time people counting makes it better-suited for today's advanced smart building environments, as it is much more accurate than the current methods available today.

B. Computer Vision–Based Crowd Counting

With advances in Deep Learning, Computer Vision (CV) techniques have been developed that are capable of accurately counting all people in images at a relatively high level of accuracy. Deep learning techniques can usually be classified into three types of methods: detection-based, regression-based, or hybrid models. Detection-based models (e.g., YOLO) identify individuals in an image by a bounding box, thus enabling real-time processing. Regression methods or hybrid models are used to count people in highly dense areas, but they require greater computational resources. Our aim is to deploy our model at the edge and therefore we select the YOLO model due to its combination of speed, accuracy and lightweight design.

C. Lightweight Architectures and Edge Deployment

Over the past few years, many researchers have investigated how deep learning-based systems can be optimized for resource-constrained environments by developing efficient architectures such as MobileNet, ShuffleNet, and EfficientNet-Lite. Many of these architectures are designed to create highly efficient models that will require minimal computation and memory resources, allowing them to operate on low-power devices. Additionally, common optimization techniques such as quantization, pruning, and knowledge distillation enable further reductions in computational complexity.

Although YOLOv5m is not the most compact version of a convolutional neural network (CNN) in the YOLO family, it provides a strong trade-off between speed and accuracy. This CNN can also benefit substantially from TensorRT optimizations on platforms such as the Jetson Nano and Jetson Xavier NX, which makes it a well-suited candidate for applications that rely heavily on deployments at the edge of the network.

D. Tracking and Temporal Smoothing

Tracking objects through a series of connected frames is one approach that provides temporal consistency; however, in our implementation we stabilize occupancy estimates (i.e., the number of people in a given area) through kalman-based temporal smoothing so that any changes between frames due to partial occlusion, missing detections, or rapid motion will not impact overall counting accuracy; thus giving an improved and reliable count along with enhanced accuracy and robustness of our real-time occupancy estimation through the addition of a kalman filter.

III. Problem Definition And Objectives

Occupancy estimation from CCTV footage can be quite challenging. Indoor environments, for example, often contain multiple people that can overlap or cluster with one another, or be partially visible depending on the type of furniture, the layout of the room, or where the camera is placed. The presence of shadows, changing illumination, and perspective issues, all create challenges to distinguishing individuals reliably. While the accuracy of detection using cloud-based deep learning approaches can be very high, these models are generally very resource intensive and require constant access to an Internet connection, which adds an element of latency when using the detected results, creates bandwidth strain, and raises privacy issues. Therefore, the approach to estimating occupancy in smart buildings and IoVT will be to use local edge devices (typically low-power and resource-constrained) to perform the task of estimating occupancy locally via the edge devices (using their optimal configurations for efficiency, instead of using the model's raw compute power).

The major challenge is to build a function that is both highly accurate and efficient in terms of the computational requirements to perform in real time on low-power devices.

A. Problem Statement

An effective computer vision system is needed for counting individuals captured by CCTV video in real time while also functioning on edge computing devices with little processing resources. Current methods are inefficient or provide poor accuracy. Traditional sensors have low levels of accuracy;

density estimation networks are computationally intensive; and current approaches cannot accurately detect individuals in changing indoor environments, indicating a need for high-performance detection systems that work within the limitations imposed by the hardware used for embedded applications.

B. System Goals

To address this problem, the proposed solution is designed around the following primary goals:

- To produce as accurate an estimate of the number of occupants as possible – no more than ±1-2 persons error, even when persons are partially occluded/overlapping.
- To provide as near to real time as possible performance (minimum 5 frames per second may be achievable, with more powerful deployment hardware).
- To function as efficiently/optimally on lightweight platforms (e.g. Jetson Nano, Jetson Xavier NX, Raspberry Pi) with no requirement for cloud off-loading.
- To provide an accurate estimate of the number and location of occupants in Real Time, therefore the Temporal Stability of the Detection Output can be optimised using Region of Interest filter and Kalman filter smoothing post-processing techniques.

C. Motivation for YOLO-Based Approach

Due to these requirements, a system built around YOLOv5m as the backbone detection structure was selected. YOLO, unlike other regressive or hybrid/deep learning methods, provides a more interpretable representation of the detected objects in the form of bounding boxes. Therefore, it is easier to interpret and optimize for deployment to edge-based platforms. It has excellent accuracy-to-computation efficiency characteristics when combined with inference-time optimizations (examples: TensorRT) for performance standards suitable for real-time workloads within the Internet of Things environment.

IV. System Operation and Workflow

Occupancy estimates are produced in an automated manner via the proposed system's utilization of live video streams from CCTV. This process will include the detection of persons within each frame through the use of YOLOv5m deep learning model.The entire workflow consists of several sequential components that have been designed to allow for accurate person detection as well as the completion of all necessary tasks in real-time while meeting any edge computing requirements.

The workflow of the proposed system begins with an ongoing acquisition of video frames captured by a surveillance camera. After capturing each video frame, the frame will be pre-processed, resized, and normalized so it meets the required input specifications as provided by the YOLOv5m deep learning model. Once the frame has been processed, the YOLOv5m model performs object detection by identifying characteristics within the video frame and using the visual characteristics to identify bounding boxes associated with human beings. Due to the model being trained on a vast amount of annotated datasets including those provided by COCO, it provides a high degree of reliability when identifying humans within video frames and will disregard any other items that are not humans.

Once the detection is complete, the detection results will be filtered by both the application of a set confidence threshold, as well as by applying Region of Interest (ROI) restrictions to further eliminate from consideration anything (including false detections) occurring in non-human physical locations within the video frame, such as on the ceiling, through reflections on objects, or on digital signs. The filtered results will be counted and allow for an accurate occupancy count of persons within the video frame.A Kalman filter combines the current frame's detection count with prior predictions, producing a stable and continuous output over time rather than abrupt numerical jumps. This improves reliability in situations where individuals overlap or move unpredictably.

V. System Architecture

The intended framework for estimating occupancy is a lightweight solution that allows for efficient operation on edge devices through the use of lightweight detection, optimized processing, and temporal smoothing. The architecture consists of four main components: Video Acquisition, Person Detection, Counting and Filtering, and Output Visualisation.

1. Input and Pre-Processing: Video frames are received from a CCTV stream then resized to a resolution that will support real-time inference on edge devices. Only the area designated as the defined Region of Interest (ROI) is processed and the ROI ignores irrelevant areas (e.g., Ceilings and Walls) to minimize the computation requirements and reduce false detection.

2. Person Detection (YOLOv5m Model): Each video frame with a person will have the people detected using the YOLOv5m object detection model. The YOLOv5m detection model is a good compromise between accuracy and resource efficiency, which makes it suitable for embedded device platforms such as Jetson Nano, Jetson Xavier NX and Raspberry Pi 4 and 5. The detection outputs of the YOLOv5m model will include bounding box coordinates and confidence scores assigned to each detected person.

3. Occupancy Count and Temporal Smoothing: The bounding boxes for each detected person in the video frame are processed to produce an estimate of the actual number of people present in each video frame. The estimation of occupancy count is smoothed using a Kalman Filter to minimize the count fluctuation that may occur from missing detections, motion or occlusion and will enhance the consistency and reliability of occupancy estimate over time.

4. Output Generation and Visualisation: The final occupancy count will be overlaid on the processed video frame and the resulting occupancy value can additionally be logged or transmitted to building management or IoT systems

```
CCTV Video Stream
        |
        v
Preprocessing (Resize + ROI)
        |
        v
YOLOv5m Person Detection
        |
        v
Count Estimation
        |
        v
Kalman Filter Smoothing
        |
        v
Final Output (Display + Logging)
```

VI. Results

The performance of the suggested CCTV-based occupancy estimation system was tested by using the Mall Surveillance Dataset. The CCTV Occupancy Estimation System was tested for its accuracy of detection, precision of counting and temporal stability. Qualitative and quantitative analyses were carried out to evaluate these parameters.

1. Qualitative Detection:

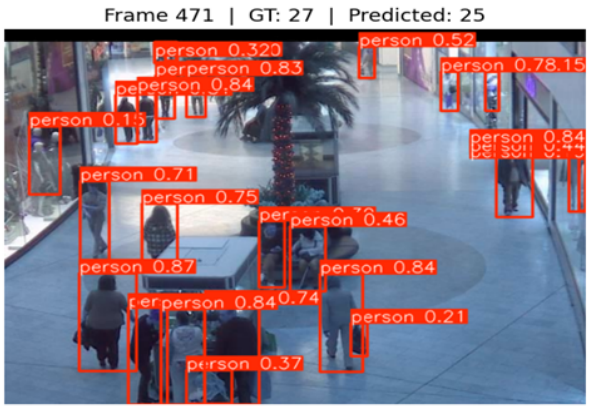

Fig. 1. Detection result for frame example

An output from Frame 471 of the YOLO model used to detect persons is shown in Figure 1: The network detects persons and places bounding boxes with a confidence value for each detected person.

Count from Ground Truth: 27 Persons

Count from Prediction: 25 Persons

Count Deviation: -2 Persons

Most of the persons detected in this example are visible in the scene; the persons who are partially occluded and in the seating area were detected correctly. The unaccounted predicted count is due to missed detections of persons that had heavy occlusions and/or who were in the background. In summary, this means that the system is capable of reliably detecting persons in moderately crowded indoor spaces, but will experience difficulties detecting those that are heavily occluded or compressed due to perspective reactions.

2. Counting Accuracy:

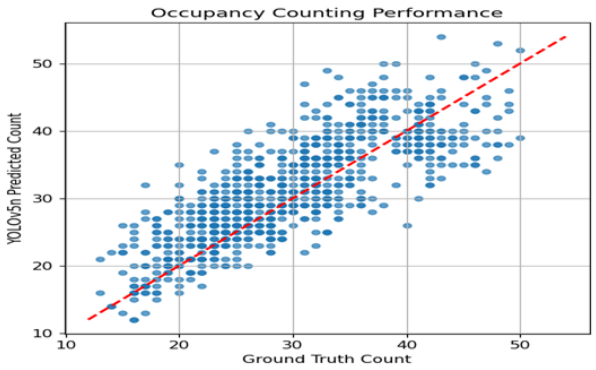

Fig. 2. Detection result for frame example

According to Figure 2, there is a close correlation between the actual and estimated persons by the YOLOv5n network, as most of the points in this figure fall along the theoretical accuracy line. Overall, the performance of YOLOv5n is favourably aligned with the trends, but when YOLOv5n predicts crowded scenes, it indicates fewer persons than were counted, likely due to occlusion of the subjects in the images. The strong positive correlation (Pearson correlation coefficient = 0.784) supports this notion, indicating a strong positive relationship between true and predicted counts. However, the relatively low accuracy measurement of 20.6

3. Temporal Stability and Kalman Filtering:

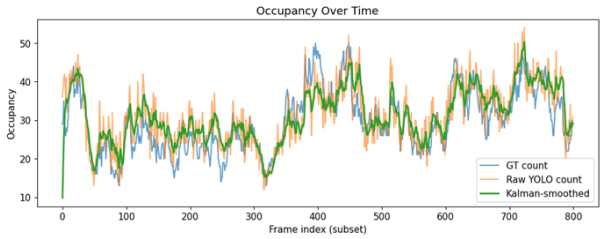

Fig. 3. Detection result for frame example

Figure 3 compares the ground-truth occupancy with the raw YOLO predictions and the Kalman-filtered results over time. While the YOLO predictions follow the true trend, they fluctuate due to missed or false detections. The Kalman filter smooths these fluctuations, producing a more stable and reliable estimate. This makes it more suitable for real-world applications where consistent occupancy tracking is preferred over frame-by-frame precision.

VII. Conclusion

In this project, we developed an occupancy counting system using CCTV footage and the YOLOv5m model. The system successfully detected and counted people in real time and performed well on edge devices such as the Jetson Nano, Jetson

Xavier NX, and Raspberry Pi. Unlike traditional sensors, this method offers higher accuracy and handles movement and partial occlusions effectively. Although the performance may decline in very crowded or low-light environments, it still provides reliable occupancy estimates. With further optimization, this approach can be widely applied in smart buildings, automation systems, and security monitoring.

REFERENCES

[1] J. Yi, F. Chen, Z. Shen, Y. Xiang, S. Xiao, and W. Zhou, "An effective lightweight crowd counting method based on an encoder–decoder network for Internet of Video Things," *IEEE Internet Things J.*, vol. 11, no. 2, pp. 3082–3093, Jan. 2024.

[2] S. Zhang, G. Wu, J. P. Costeira, and J. M. F. Moura, "Understanding traffic density from large-scale Web camera data," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, 2017, pp. 4264–4273.

[3] H. Xu, Z. Cai, R. Li, and W. Li, "Efficient CityCam-to-edge cooperative learning for vehicle counting in ITS," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 9, pp. 16600–16611, Sep. 2022.

[4] T. Li, H. Chang, M. Wang, B. Ni, R. Hong, and S. Yan, "Crowded scene analysis: A survey," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 25, no. 3, pp. 367–386, Mar. 2015.

[5] D. Kang, Z. Ma, and A. B. Chan, "Beyond counting: Comparisons of density maps for crowd analysis tasks—Counting, detection, and tracking," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 29, no. 5, pp. 1408–1422, May 2019.

[6] X. Yu, Y. Liang, X. Lin, J. Wan, T. Wang, and H.-N. Dai, "Frequency feature pyramid network with global-local consistency loss for crowd-and-vehicle counting in congested scenes," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 7, pp. 9654–9664, Jul. 2022.

[7] J. Liu, S. Zheng, G. Xu, and M. Lin, "Cross-domain sentiment aware word embeddings for review sentiment analysis," *Int. J. Mach. Learn. Cybern.*, vol. 12, no. 2, pp. 343–354, 2021.

[8] J. Liu *et al.*, "Aliasing black box adversarial attack with joint self-attention distribution and confidence probability," *Expert Syst. Appl.*, vol. 214, Mar. 2023, Art. no. 119110.

[9] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," arXiv:1409.1556, 2014.