Transactions on Energy Systems and Engineering Applications

Article

A Computer vision based system for human detection and automatic people counting

Gabriela Curiel[1], Kevin Guerrero[1], Diego Gómez[1] and Daniela Charris[1,*]

[1] Department of Electrical and Electronics Engineering, Universidad del Norte, Puerto Colombia, Colombia.
[*] Correspondence: dmcharris@uninorte.edu.co

Received: 21 January 2024; Accepted: 14 September 2024; Published: 25 October 2024

Abstract: Occupancy control is a fundamental aspect of managing spaces and services effectively. It aims to ensure safety, compliance with regulations, emergency preparedness, and overall satisfaction for individuals and businesses. To align with the described need, this paper presents a computer vision based system for automatic people counting in gates. The system is divided in five stages: video capture, motion analysis, human detection, human tracking and people counting. A camera captures the top-view image of the gate and analyze the change or movement in the objects in scene. When motion is detected, the frame is sent to the object detector, which is a convolutional neural network. Then, a tracking algorithm analyzes the movement patterns of people. According to the route, it is determined whether the person arrives or leaves and the count is updated. Two test scenarios are analyzed: the entry of a public bus and a building gate. The people detection module is tested, showing a mean average precision of 95.2%. Also, the counting is tested showing an average precision of 96.8%, a recall of 92% and an F1-Score of 94.3%. Finally, the system performance is evaluated, showing an average processing time of 34.2 ms.

Keywords: Computer vision, deep learning, human detection, convolutional neural network, object tracking, people counting.

© 2024 by the authors. Published by Universidad Tecnológica de Bolívar under the terms of the Creative Commons Attribution 4.0 License. Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI. https://doi.org/10.32397/tesea.vol5.n2.624

1. Introduction

The growing population and degree of urbanization has significantly increased the demand for products, services and public spaces. Scenarios such as roads, public transportation, shopping centers, museums, markets, sports facilities and buildings, where a large number of people gather, must be appropriately designed and managed so that people can enjoy them safely [1].

In commercial buildings, excessive crowding results in time delays and has an impact on the user experience of costumers. Managers and security personnel implement systems that allow them to know the flow of people and make decisions about available personnel and resources. In addition, regulations must be followed to ensure the safety of customers, visitors and employees [2].

How to cite this article: Curiel, Daniela; Guerrero, Kevin; Gómez, Diego; Charris, Daniela. A Computer vision based system for human detection and automatic people counting. *Transactions on Energy Systems and Engineering Applications*, 5(2): 624, 2024. DOI:10.32397/tesea.vol5.n2.624

Although in public scenarios such as stadiums, cinemas and museums the ticket offices control the occupancy, the public transportation system is more unpredictable and delicate. It is exposed to traffic jams, road maintenance, changes in user demand, among other factors. Thus, passenger counting plays a vital role in the effective management of routes and buses. In Colombia, most of user complaints in public transportation systems are associated with excess passengers on buses or delays in some of the services [3–6]. Therefore, it is necessary to implement effective people counting systems to manage demand and improve route planning.

Technologies for people counting range from traditional mechanical devices such as tally counters and turnstiles, to modern electronic solutions such as infrared (IR) sensors, Wi-Fi sensors, radars and cameras [7]. Each technology responds to specific requirements, however, their efficiency levels vary, resulting in more cost-effective solutions.

Mechanical devices are considered the oldest approaches. At entrances to shopping centers and stores, building guards keep track of people arriving and leaving by using a tally counter. It is also used to make inventories of objects and animals [8, 9]. Despite being very cheap, the main disadvantage is that it requires a person recording the number of people in attendance, therefore, it includes high human error. On the other hand, mechanical turnstiles are uncomfortable, lead to congestion during peak people flow, and their effectiveness is compromised because they can be bypassed.

Infrared people counters use two-way infrared rays to count the number of people passing through a gate [10]. It is easy to install and suitable for narrow accesses. Another alternative is a mat attached to the floor. The floor surface at the gate is equipped with sensors that detect when someone steps in. Both systems cause false detections when an obstacle different than a human passes through the gate. On the other hand, WiFi based-systems detect smartphones connected to count people [11]. They require a WiFi access point for the users to connect. The data collected is imprecise because it counts only the devices.

Several researchers have expressed an interest in enhancing people counting systems through innovative technologies including computer vision. Thermal or thermographic cameras detect and capture the infrared radiation emitted by objects. Unlike traditional cameras that capture visible light, they operate in the infrared spectrum, allowing them to record temperature variations in the scene [12]. The different temperatures represented by varying intensities of infrared radiation and translates it into a thermal image. Thermal cameras can be utilized to detect the presence of individuals based on their body heat signatures, offering a method for counting passengers in various environments [13]. The authors in [14] propose an occupancy estimation system based on a low-resolution thermal camera. A similar approach is presented in [15], where the authors present a pedestrian tracking system using the body temperature of an infrared camera.

Depth cameras provide information about the distance of objects from the camera. They can be used to determine the number of people by recognizing the spatial arrangement of individuals and their distances from the camera [16]. In [17], the authors use a depth camera to count passengers in a bus from a top-view image. Also, in [18], the authors present an automatic people counter in a gate with this camera.

RGB cameras capture and record images in the visible spectrum of light with the combination of three primary colors: red, green, and blue. RGB cameras are the most common type of cameras used in everyday devices such as smartphones, digital cameras, and webcams. RGB cameras are versatile and widely used due to their ability to capture detailed and realistic color information, making them suitable for a broad range of visual tasks. This technology has been used for counting both in buildings and in public transportation. The system presented by [19] is designed to track the movement of people in shopping centers analyze peak hours, while authors in [20] count passengers in a bus using the frontal view of the bus. Although all these camera technologies can be combined to obtain better results, the ones presented first (thermal and depth) are more expensive than RGB cameras, therefore, they are less used in counting applications.

Deep learning has proven to be particularly effective in image analysis, making it better suited for certain tasks compared to traditional methods. These models learn to identify complex patterns and features at different levels of abstraction, making them well-suited for tasks like object detection, segmentation, and classification. The authors in [21] perform crowd surveillance in streets, while the authors in [20, 22–25] detect passengers in buses using deep neural networks. It is also possible to use deep learning techniques to track the detected objects, by complementing the operation with algorithms such as Multiple Object Tracking (MOT), Simple Online And Realtime Tracking (SORT) [26], FairMOT [27] and TransMOT [28].

Visual Language Models integrate both visual and textual data and provide more powerful tools to analyze videos. They have been designed to adapt to different contexts. However, they are considered computationally expensive, since they combine convolutional and recurrent neural networks with transformers, requiring significant GPU resources and memory [29, 30]. For tasks that involve processing and analyzing only visual data without any textual input, such as image classification, object detection, and semantic segmentation, CNNs are typically more efficient and effective, especially for tasks where text understanding is not necessary. In scenarios where only visual information needs to be processed, CNN can lead to faster inference times and lower resource requirements.

Although many computer vision technologies have been used to address this problem, factors such as camera location, angle, and resolution may cause improper operation of the counting system. Furthermore, continuous operation must be guaranteed and available resources must be optimized. As a result, is important to choose a suitable embedded platform and monitor its performance metrics. To align with the described need, this paper presents a computer vision based system for automatic people counting in gates. The authors complemented the Automatic Passenger Counter presented in [31] for the bus transportation system in the city of Pereira, Colombia. In addition to modifying the architecture to reduce processing costs, the execution time of each of the modules is analyzed to determine possible bottlenecks and showing a continuous operation in the selected embedded platform. The document is structured as follows: Section 2 details the system and the modules that compose it. Then, Section 3 shows the experimental results of the tests in the detection module and overall operation. Section 4 discusses the system performance and the conclusions are shown in Section 5.

2. Methods and Materials

After reviewing the different technologies and studies carried out in automatic people counting, as well as their advantages and limitations, the authors propose a computer vision based system. To improve the counting performance in an embedded platform, the architecture presented in [31] is modified by adding a motion detection algorithm. As a result, the system is composed of five modules: capture, motion detection, object detection, counting, and tracking. The modules are organized into two processing threads (T1 and T2), as presented in Figure 1.

In T1, the system connects to the camera or video, extracts the frames and a performs the motion detection algorithm. If a significant change is detected in the image pixels, the frame is saved in the buffer so that it can be processed by T2. On the other hand, the buffer is constantly revised in T2. If a new frame is stored, the system activates the object detection, tracking and counting operations. A detailed description of each module is presented as follows.

2.1. Video Capture

Due to its advantages over other camera technologies, an RGB camera is used to analyze the flow of people from a top-view of the entrance. Taking into account works such as [17, 20, 22, 24, 31], the camera is placed at the upper part of the portal with the lens pointing towards the floor, as shown in Figure 2 for

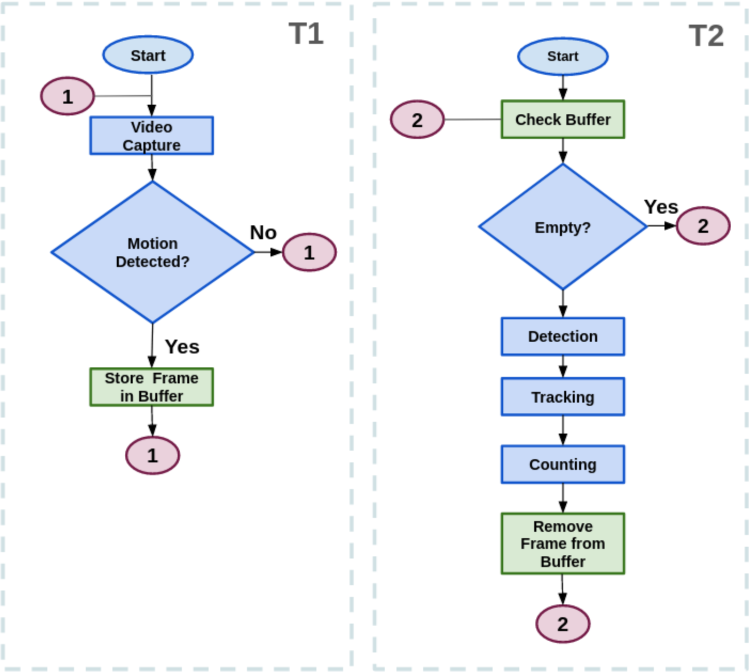

Figure 1. Code Flowchart.

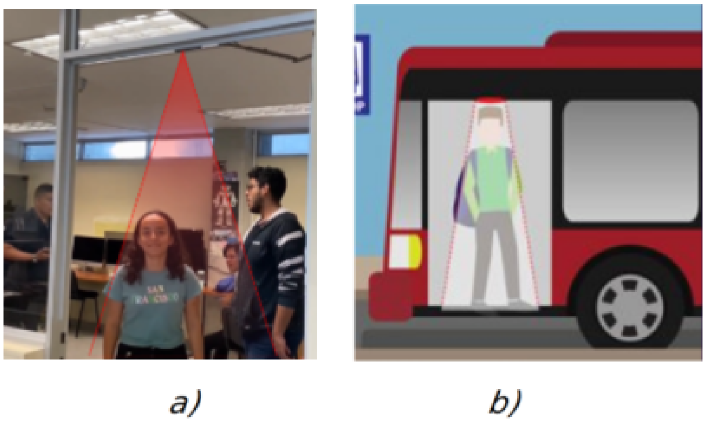

Figure 2. Location and angle of camera in a) building gate and b) bus gate.

the selected test scenarios. The image obtained from the camera with this location and angle is evident in Figure 3.

The camera provides a resolution of 1280x720 pixels and 30 frames per second (FPS). Then, the video capture module connects to it to obtain the frames by using OpenCV libraries. This module also receives pre-recorded videofiles and image sequences as input.

For the bus gate, the dataset of videos was collected in public buses in the city of Pereira under the execution of projects with the Ministry of Science, Technology, and Innovation of the Colombian Government and Universidad del Norte. On the other hand, in the case of the building gate videos, the people used for this study were volunteers from the Engineering College. The people involved in this study received informed consent about the use of their visual information delivered through the cameras.

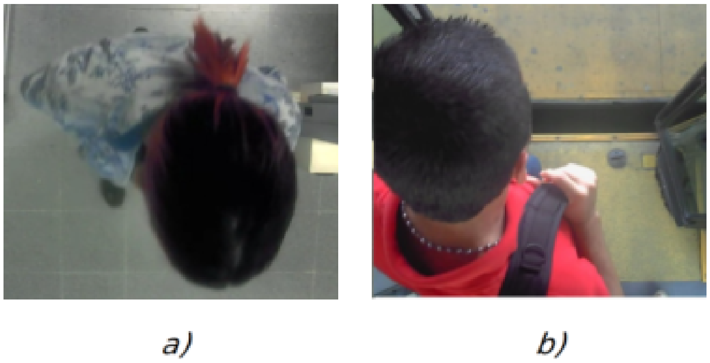

Figure 3. Camera view in a) building gate and b) bus gate.

2.2. Motion Detection

One of the main bottlenecks of computer vision technologies is the high computational cost involved in image operations. If algorithms are complex, the load on the platform and the processing time will be greater. In the case of a modular system, such as the one proposed, the failure or delay in a module has a direct impact on the performance of the entire system.

In the test scenarios, it is possible to find times of the day when there is no flow of people and the image obtained by the camera is completely still. However, the system proposed in [31] performs complete and uninterrupted operation as the system is constantly searching for people in the image. Decreasing the FPS of the camera is not feasible because when a person passes the portal there will be fewer frames to analyze. Thus, the authors propose to implement a motion detection algorithm, which does run uninterrupted, but requires a lower computational cost. It sends frames to be analyzed only when necessary.

Motion detection is widely used in security systems, traffic analysis on highways, object tracking, people counting, among others. There is a variety of methods to perform motion detection, however, the most common is background extraction. It assumes that the background does not change over between consecutive frames and elements that are not static can be detected in the video because the pixels move through the frame. In order to counteract lighting changes that affect the appearance of the background, probabilistic methods are implemented to analyze the distribution of pixels in the image and distinguish between actual motion and shadowing. For its implementation, the Gaussian Mixture Model-based foreground and background segmentation is performed [32–34].

Despite more complex methods are very powerful to perform background extraction, are considered computationally expensive. Considering that in this approach the camera is mounted and fixed, and the end goal is to deploy the system in an embedded platform, the numerical difference between the pixels of two consecutive frames is used to detect motion. With this, the percentage of change in the image is calculated and if this exceeds a threshold, the current frame is sent to analyze.

2.3. Human Detection

Object detection methodologies has evolved from conventional approaches such as Viola-Jones [35] and HOG detectors [36] to deep learning methods such as Single Shot Detector (SSD) [37] and You Only Look Once (YOLO) [23]. Following an exhaustive research of techniques for human detection, a neural network-based object detector is chosen. Taking into account compatibility and resources with C++ programming language, the Darknet framework is used to fine-tune different YOLO versions. YOLO distinguishes itself by its proficiency in efficiently detecting individuals through the processing of entire images using a singular neural network.

Since the system is expected to be executed in an embedded platform, several versions of YOLO-tiny are compared, which have the same updates between versions but with fewer convolutional layers. For this, the test is to fine-tune the versions YOLOv2-tiny, YOLOv3-tiny, YOLOv4-tiny and YOLOv7-tiny with the same test dataset and compare metrics such as Mean Average Precision (mAP) and mean Intersection over Union (IoU). The dataset, as described in [31], comprises videos sourced from the public transport system in the city of Pereira. Images with different operational conditions are extracted from these videos and annotated to indicate passenger locations. Situations such as people with hats, bags, umbrellas and accessories in general, as well as different heights and hair types were addressed in the fine-tuning process of the object detection model. Creating a diverse dataset was crucial to ensure the ability to generalize well to new and unseen data. Thus, we included a wide variety of examples that cover different scenarios, angles, lighting conditions, backgrounds, etc. Considering that our approach is based on a deep learning model, it was necessary to tackle this problem from the object detection model. Thus, in the testing process, 70% of the dataset was randomly chosen for training, 15% for validation and 15% for testing. Then, the versions are fine-tuned in a NVIDIA RTX2080. The comparison is shown in Table 1.

Table 1. Results of different fine-tuned YOLO-tiny models.

Model	mAP	mIoU	# of conv. layers
YoloV2-tiny	89.7%	55.9%	9
YoloV3-tiny	92.1%	58.5%	13
YoloV4-tiny	**95.2%**	**57.9%**	**21**
Yolov7-tiny	98.4%	66.1%	58

Although the most advanced version of YOLO has a better performance, the number of convolutional layers makes its computational cost very high. Considering the metrics, YOLOv4-tiny is selected. With the trained model, the detection stage is executed in a second processing thread. It runs through the frame vector that is being filled by the video capture stage. Necessary files include the trained weights of the YOLO model, the list of classes (only "person"), and the configuration file. The output of the network consists of the detection (object location and probability). By setting a probability threshold of 80% person, favorable detections are selected, as shown in Figure 4.

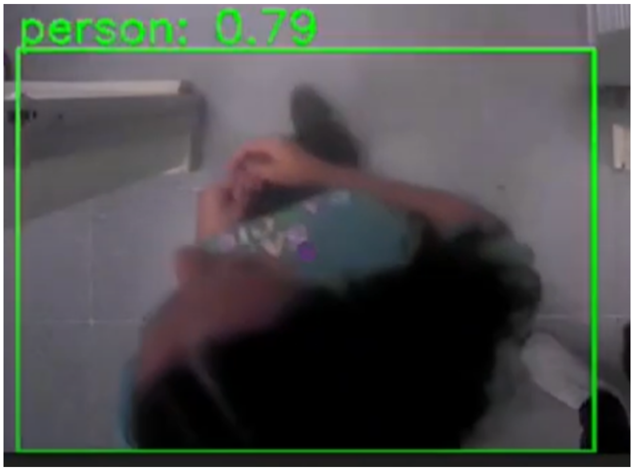

Figure 4. Visualization of object location and probability

2.4. Object Tracking

Following the detection stage, this phase utilizes the Optical Flow by Lucas Kanade as the tracking algorithm. This choice is made due to its capability to provide comprehensive motion information for two successive frames [38]. From the detection stage, bounding boxes are extracted used to calculate the centroid point of the detection. Then, it is assigned to a tracker, which saves the route. Both, route and current centroid position are shown in Figure 5.

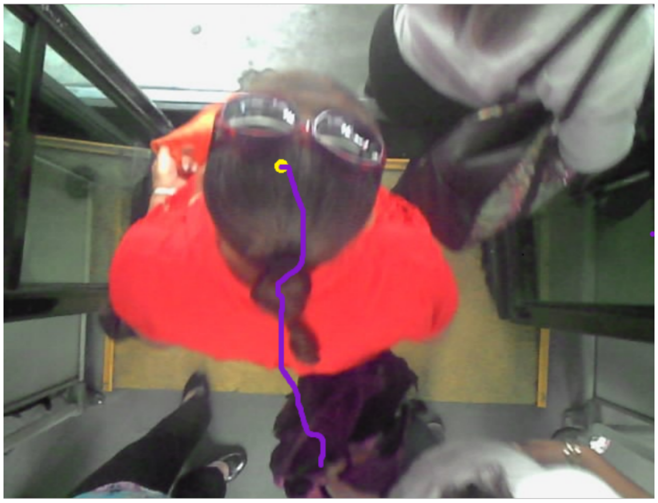

Figure 5. Visualization of bounding box centroid and tracker route.

Although within the scenarios it is expected that only one person passes at a time, it is possible that we have high attendance or people entering at the same time. Tracking algorithms are designed to handle multiple detections by distinguishing between different objects, maintaining identity over time, and dealing with occlusions and interactions. Our approach matches detections to existing trackers based on a distance metric between the centroids of the detections and a route. Thus, if two or more people pass through the frame at the same time, the tracking algorithm can identify and assign the closest detection to the closest tracker, allowing a number greater than one of objects to exist within the frame.

2.5. People Counting

The trajectory of each tracker is evaluated across two consecutive frames, considering three possibilities: the route goes from the upper zone to the lower zone, the route goes from the lower zone to the upper zone, or exhibits no positional changes. For that, specific zones and thresholds within the image are established and presented in Figure 6. This logic can be applied to any type of entry and exit portal, and under any configuration.

Coordinates for both a starting point (P0) and an ending point (P1) are derived from the trajectory across two frames. If P0 falls within the upper zone and P1 surpasses the threshold into the lower zone, leads to an increment in the counter from upper to lower. Conversely, if P0 is in the lower zone, and P1 surpasses the threshold into the upper zone, there is a counter increase between lower to upper zone. Instances where the individual does not transition between zones, i.e., doesn't surpass the threshold, are not counted by the system. Figure 7 illustrates examples of the counting system.

The thresholds for selecting when a person left or entered are based on the boundary of the access door seen from the camera angle. Considering that the tracking module evaluates the centroids of the detections, that is, the centroids of the people seen from above, these thresholds were calibrated manually considering the location of the camera and the route followed by the people entering and exiting the gate.

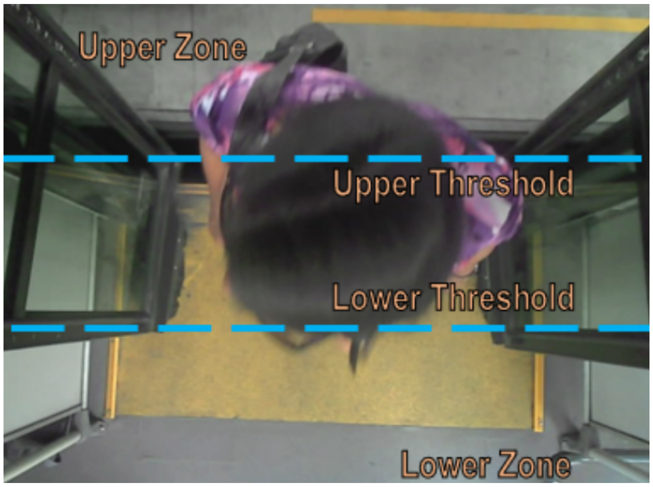

Figure 6. Zone delimitation.

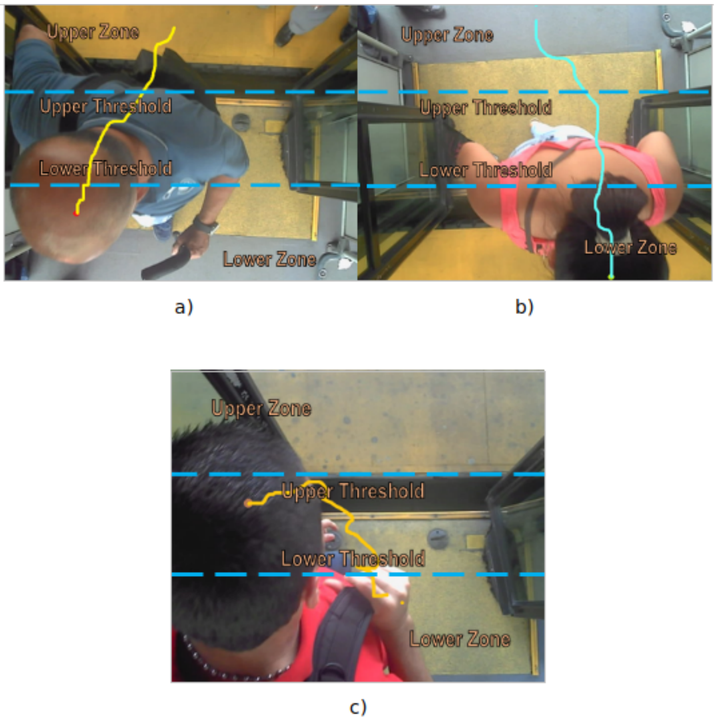

Figure 7. a) Exit b) Entry c) Not counted.

An inappropriate calibration of these thresholds would cause a person to be counted entering or leaving without having completely passed the gate.

Since the system is expected to be uninterrupted in the test scenarios: building doors and buses, it is necessary that the processing core supports this continuous operation. Embedded systems are essential in due to their real-time processing capabilities, integration capabilities, reliability, compact size, cost efficiency, customization options, dedicated functionality, and durability in challenging environments. They provide a solid foundation for building efficient and robust automation solutions tailored to the needs of industrial processes. Therefore, it is proposed to use an Axiomtek industrial computer. The selected computer has an Intel Core i7 processor and 16GB of RAM.

Despite devices such as the Raspberry Pi or the NVIDIA Jetson Nano are excellent for prototyping, educational purposes, and low-cost projects, the scenarios proposed require enhanced reliability, performance, and longevity for the embedded platform. Industrial computers are built with high-quality, industrial-grade components designed to withstand harsh environments, including extreme temperatures, humidity, dust, and vibrations. They are designed for continuous operation and have a longer life-cycle.

Figure 8 shows all the components of the proposed system. For the fine tuning stage of the object detection module (Figure 8.a.), a computer with the graphics card is used, which receives the images for training as input and produces the deep learning model files. These files are transferred to the embedded platform and used to analyze the videos or camera stream, as presented in Figure 8.b.

Finally, sending the files over the Internet was not considered either, since this implied having an Internet connection and additional costs of transmitting the information and hosting the object detection model on a web server in the cloud.

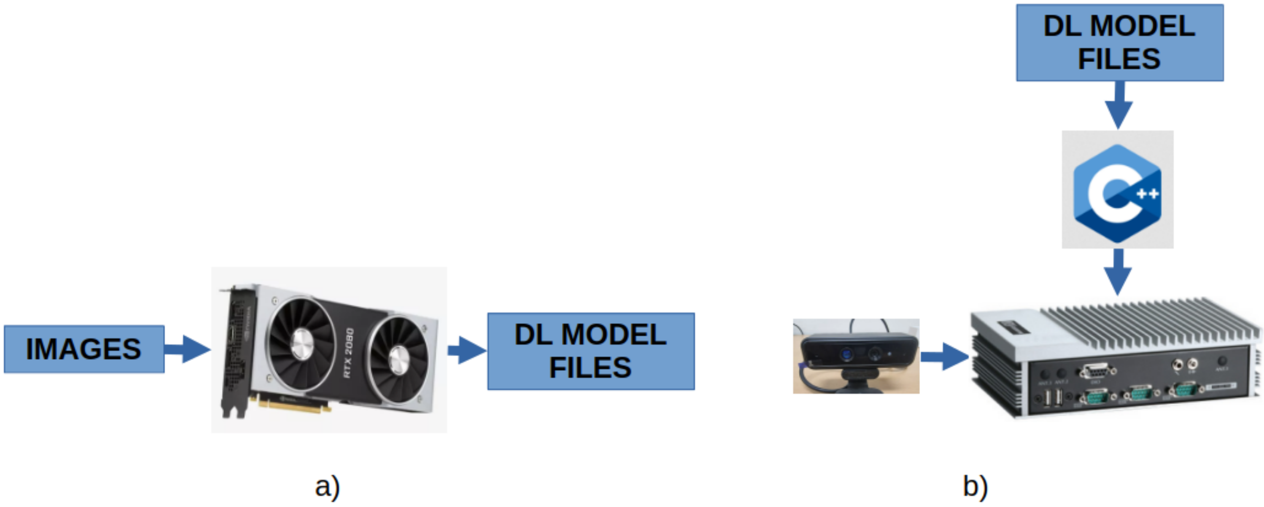

Figure 8. Block diagram for a) fine-tuning process b) production in embedded platform.

3. Experimental Tests and Results

To evaluate the counting system and perform comparisons with the approach presented in [31], an experimental design is implemented to assess performance across various metrics [39]. The public transportation system in the city of Pereira is used for testing. 47 videos of 30 seconds recorded at 25 FPS are analyzed. Thus, a camera installed in a gate of a building from Universidad del Norte is used to test the system in real time.

The following concepts are defined:

- True positives (TP): correct entry or exit reported
- False Positives (FP): incorrect entry or exit reported
- False negatives (FN): no input or output is reported (incorrectly)

Additionally, precision, recall, and F1-score metrics are defined as

$$\text{precision} = \frac{T_P}{T_P + F_P}, , \quad (1)$$

$$\text{recall} = \frac{T_P}{T_P + F_N}, , \quad (2)$$

$$\text{F}_{\text{score}} = \frac{2 * \text{precision} * \text{recall}}{\text{precision} + \text{recall}}. \quad (3)$$

In order to facilitate a comprehensive comparison, it was imperative to manually determine the exact count of individuals in each video. The resulting data is tabulated, with each row corresponding to a video frame. This information is compared with the values obtained from the proposed approach. The table included entries for the actual count of people entering and exiting, the system-generated count, TP, FP and FN for each video segment, along with corresponding calculations of precision, recall, and F-value.

Percentage values of the validation metrics were obtained, as well as their respective confidence intervals. They are displayed in Figure 9.

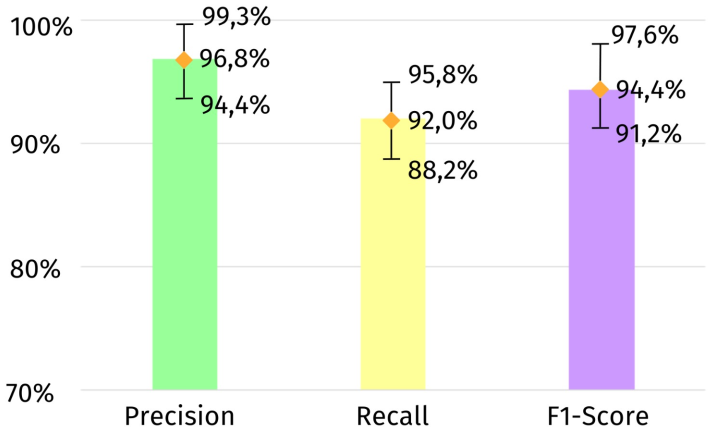

Figure 9. Confidence intervals for precision, recall and F1-score.

The metrics presented correspond to the evaluation of the system in the pre-recorded videos of the public bus transportation system in the city of Pereira. A proof of concept of the system was carried out on a building gate, however, no formal metrics were taken from it. Also, Table 2 show the average system performance metrics taken from the tests.

The metrics given in Table 2 were obtained directly on the embedded device using C++ timing libraries. In the code, the execution time of each block of instructions corresponding to each module was measured and saved through the system logs. At the end the average of the metrics in all the videos was computed.

Table 2. Performance metrics of the system.

Metrics	Average Value
FPS	29.2
Capture time	4.6 ms
Motion time	0.7 ms
Detection time	27 ms
Tracking time	1.3 ms
Counting time	0.6 ms
Total time	34.2 ms

4. Discussion

In contrast to the approach presented in [31], notable distinctions arise. With the selection of the Darknet framework for object detection and the comprehensive evaluation of different YOLO versions, YOLOv4-tiny is selected for the implementation. Despite YOLOv7-tiny shows higher mAP (98.4%) and mIoU (66.1%) in the testing images, the amount of convolutional layers stands out as a disadvantage. This number increases the amount of mathematical operations in the images and, consequently, the processing time per frame. Considering that the system is designed with continuous operation with a camera of 30FPS, a proper relationship between performance and inference time is found with YOLOv4-tiny.

Then, when the entire system is evaluated, the average values of precision (96.8%), recall (92%) and F1-score (94.4%) show a proper operation of the counting with a confidence level of 95%. The precision remains over 95%, that is, the number of false positives is low. On the other hand, the lowest metric is recall, reflecting failures in the system by not counting people. Analyzing the videos and specific cases, this situation is mainly evidenced in the simultaneous passage of several people in opposite directions and children. Therefore, the F1-score that combines both precision and recall is calculated.

Regarding the embedded platform, the processing time for each stage is measured and analyzed. Among all the stages, the object detection part shows the highest processing time, due to the convolutional neural network. Despite the motion detection algorithm uses the entire frame, the mathematical operations are simple and the processing time is low. The tracking and counting modules does not use the image as source of information, They analyze the routes and centroids of detections. Thus, the processing time is low. With all the stages, the system takes 34.2ms to process a frame, resulting in 29.2 FPS of continuous operation. However, since the motion detection stage filters some frames, the difference between the camera and system FPS is compensated. In the worst cases, a delay of a milliseconds is be generated in the counting, which would stabilize when the frame becomes static again and is not visible for human.

5. Conclusions

This work presents a system for human detection and automatic people counting in gates through the utilization of cameras as sensors. An RGB camera is selected for video capture due to its ability to capture comprehensive visual data. The investigation involves the exploration and testing of various deep learning and computer algorithms. To achieve accurate recognition of individuals in public transportation, the YOLOv4-tiny convolutional neural network is fine-tuned for detection using top-view imagery. Additionally, a Visual Optical Flow tracking algorithm is implemented to identify the position and movement of people, facilitating precise passenger counting.

The performance of the system is evaluated, showing an average precision of 96.8%, a recall of 92%, and an F1-score of 94.4% in the counting process. These metrics, with a 95% confidence level, validate the chosen technologies. Also, performance metrics in the selected platform show an average processing time of 34.2ms and 29.2 FPS. As future work, mathematical operations on images and neural network inference can be optimized with CPU libraries such as AVX, GStreamer and OpenCL.

Acknowledgments

The authors thank Universidad del Norte for providing the videos collected in public buses in the city of Pereira under the execution of projects with the Ministry of Science, Technology, and Innovation of the Colombian Government.

Funding: This research received no external funding.

Author contributions: Conceptualization, G.C., K.G., D.G. and D.C.; Methodology, D.G. and D.C.; Software, G.C. and K.G.; Validation, G.C., K.G. and D.C.; Formal Analysis, D.G. and D.C.; Investigation, D.G. and D.C.; Resources, D.G. and D.C.; Data Curation, G.C. and K.G.; Writing – Original Draft Preparation, G.C.; K.G.; D.G. and D.C.; Writing – Review & Editing, G.C.; K.G.; D.G. and D.C.; Visualization, G.C. and D.C.; Supervision, D.G. and D.C.; Project Administration, D.G. and D.C.; Funding Acquisition, D.G.

Disclosure statement: The authors declare no conflict of interest.

References

[1] Xianzhi Li, Qiao Yu, Bander Alzahrani, Ahmed Barnawi, Ahmed Alhindi, Daniyal Alghazzawi, and Yiming Miao. Data fusion for intelligent crowd monitoring and management systems: A survey. *IEEE Access*, 9:47069–47083, 2021.

[2] Ali M. Al-Shaery, Shroug S. Alshehri, Norah S. Farooqi, and Mohamed O. Khozium. In-depth survey to detect, monitor and manage crowd. *IEEE Access*, 8:209008–209019, 2020.

[3] Periódico El Heraldo. ¿a qué se deben los retrasos en las rutas de transmetro?, 2022.

[4] Periódico El Universal. ¿por qué se demoran tanto en pasar los buses de transcaribe?, 2021.

[5] Revista Semana. Aumentan quejas por falta de buses alimentadores en el portal de suba de transmilenio, 2022.

[6] Periódico El País. La increíble cifra de caleños regañados en su trabajo por culpa de los retrasos del mÍo, 2023.

[7] Chris Mccarthy, Irene Moser, Prem Prakash Jayaraman, Hadi Ghaderi, Adin Ming Tan, Ali Yavari, Ubaid Mehmood, Matthew Simmons, Yehuda Weizman, Dimitrios Georgakopoulos, Franz Konstantin Fuss, and Hussein Dia. A field study of internet of things-based solutions for automatic passenger counting. *IEEE Open Journal of Intelligent Transportation Systems*, 2:384–401, 2021.

[8] Charlie Kim and Joseph Derisi. Versacount: Customizable manual tally software for cell counting. *Source code for biology and medicine*, 5:1, 01 2010.

[9] Julismah Jani. Using tally counter to count heart rate in physical fitness activity: An innovation approach. *IJAEDU-International E-Journal of Advances in Education*, 1:112, 08 2015.

[10] Sylvia T. Kouyoumdjieva, Peter Danielis, and Gunnar Karlsson. Survey of non-image-based approaches for counting people. *IEEE Communications Surveys and Tutorials*, 22(2):1305–1336, 2020.

[11] Navod Suraweera, Alycia Winter, Julian Sorensen, Shenghong Li, Mark Johnson, Iain B. Collings, Stephen V. Hanly, Wei Ni, and Mark Hedley. Passive through-wall counting of people walking using wifi beamforming reports. *IEEE Systems Journal*, 15(4):5476–5482, 2021.

[12] V-COUNT. People counting technologies: A comprehensive guide, 2021.

[13] Kamaleswari P and Krishnaraj N. An assessment of object detection in thermal (infrared) image processing. In *2023 Third International Conference on Artificial Intelligence and Smart Energy (ICAIS)*, pages 1498–1503, 2023.

[14] Mertcan Cokbas, Prakash Ishwar, and Janusz Konrad. Low-resolution overhead thermal tripwire for occupancy estimation. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 398–406, 2020.

[15] Peng-Rong Tsou, Cheng-En Wu, Yen-Ru Chen, Yun-Ting Ho, Jun-Kai Chang, and Hsiao-Ping Tsai. Counting people by using convolutional neural network and a pir array. In *2020 21st IEEE International Conference on Mobile Data Management (MDM)*, pages 342–347, 2020.

[16] Shijie Sun, Naveed Akhtar, Huansheng Song, Chaoyang Zhang, Jianxin Li, and Ajmal Mian. Benchmark data and method for real-time people counting in cluttered scenes using depth sensors. *IEEE Transactions on Intelligent Transportation Systems*, 20(10):3599–3612, 2019.

[17] José Rodríguez Cadena. Contador automático de pasajeros. B.s. thesis, Universidad del Norte, 2017.

[18] Seyed Ali Hosseini Shamoushaki, Mohammad Mostafa Talebi, Amineh Mazandarani, and S. Jafar Hosseini. A high-accuracy, cost-effective people counting solution based on visual depth data. In *2020 International Conference on Machine Vision and Image Processing (MVIP)*, pages 1–6, 2020.

[19] Andes Herviana, Dodi Wisaksono Sudiharto, and Fazmah Arif Yulianto. The prototype of in-store visitor and people passing counters using single shot detector performed by opencv. In *2020 1st International Conference on Information Technology, Advanced Mechanical and Electrical Engineering (ICITAMEE)*, pages 169–174, 2020.

[20] Hayato Nakashima, Ismail Arai, and Kazutoshi Fujikawa. Passenger counter based on random forest regressor using drive recorder and sensors in buses. In *2019 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops)*, pages 561–566, 2019.

[21] Muhammad Haris Kaka Khel, Kushsairy Abdul Kadir, Sheroz Khan, Mnmm Noor, Haidawati Nasir, Nawaf Waqas, and Akbar Khan. Realtime crowd monitoring—estimating count, speed and direction of people using hybridized yolov4. *IEEE Access*, 11:56368–56379, 2023.

[22] Immanuel Jose C. Valencia, Marielet A. Guillermo, Elmer P. Dadios, Alexis M. Fillone, Edwin Sybingco, and Robert Kerwin C. Billones. Overhead view bus passenger detection and counter using deepsort and tiny-yolo v4. In *2022 IEEE 14th International Conference on Humanoid, Nanotechnology, Information Technology, Communication and Control, Environment, and Management (HNICEM)*, pages 1–6, 2022.

[23] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. You only look once: Unified, real-time object detection. pages 779–788, 06 2016.

[24] Farzan Erlik Nowruzi, Wassim A. El Ahmar, Robert Laganiere, and Amir H. Ghods. In-vehicle occupancy detection with convolutional networks on thermal images. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pages 941–948, 2019.

[25] Robert Seidel, Nico Jahn, Sambu Seo, Thomas Goerttler, and Klaus Obermayer. Napc: A neural algorithm for automated passenger counting in public transport on a privacy-friendly dataset. *IEEE Open Journal of Intelligent Transportation Systems*, 3:33–44, 2022.

[26] Nicolai Wojke, Alex Bewley, and Dietrich Paulus. Simple online and realtime tracking with a deep association metric. In *2017 IEEE International Conference on Image Processing (ICIP)*, pages 3645–3649, 2017.

[27] Yifu Zhang, Chunyu Wang, Xinggang Wang, Wenjun Zeng, and Wenyu Liu. Fairmot: On the fairness of detection and re-identification in multiple object tracking. *International Journal of Computer Vision*, 129(11):3069–3087, sep 2021.

[28] Peng Chu, Jiang Wang, Quanzeng You, Haibin Ling, and Zicheng Liu. Transmot: Spatial-temporal graph transformer for multiple object tracking, 2021.

[29] Huitong Chen. Comparison of large language and vision models on representative downstream tasks. In *2023 International Conference on Image Processing, Computer Vision and Machine Learning (ICICML)*, pages 307–311, 2023.

[30] Yueting Yang, Xintong Zhang, Jinan Xu, and Wenjuan Han. Empowering vision-language models for reasoning ability through large language models. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 10056–10060, 2024.

[31] Gabriela Curiel, Kevin Guerrero, Diego Gómez, and Daniela Charris. An improved architecture for automatic people counting in public transport using deep learning. In *2023 1st IEEE Colombian Caribbean Conference (C3)*, 2023.

[32] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, volume 2, pages 28–31 Vol.2, 2004.

[33] Zoran Zivkovic and Ferdinand van der Heijden. Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters*, 27(7):773–780, 2006.

[34] P. KaewTraKulPong and R. Bowden. *An Improved Adaptive Background Mixture Model for Real-time Tracking with Shadow Detection*, pages 135–144. Springer US, Boston, MA, 2002.

[35] Ahmed Elngar, Mohamed Arafa, Abd Naeem, Ahmed Essa, and Zahra shaaban. The viola-jones face detection algorithm analysis: A survey. *Journal of Cybersecurity and Information Management*, pages 85–95, 01 2021.

[36] Seemanthini .K and Manjunath S.S. Human detection and tracking using hog for action recognition. *Procedia Computer Science*, 132:1317–1326, 01 2018.

[37] Songmin Jia, Chentao Diao, Guoliang Zhang, Ao Dun, Yanjun Sun, Xiuzhi Li, and Xiangyin Zhang. Object detection based on the improved single shot multibox detector. *Journal of Physics: Conference Series*, 1187:042041, 04 2019.

[38] Bruce Lucas and Takeo Kanade. An iterative image registration technique with an application to stereo vision (ijcai). volume 81, 04 1981.

[39] Humberto Gutiérrez Pulido and Román Salazar. *Análisis y Diseño de Experimentos*. 05 2012.