Smart Sensing Supporting Energy-Efficient Buildings: On Comparing Prototypes for People Counting

Lorenzo Monti
University of Bologna, Italy
Bologna, Italy
lorenzo.monti20@unibo.it

Silvia Mirri
University of Bologna, Italy
Bologna, Italy
silvia.mirri@unibo.it

Catia Prandi
University of Bologna, Italy
Bologna, Italy
catia.prandi2@unibo.it

Paola Salomoni
University of Bologna, Italy
Bologna, Italy
paola.salomoni@unibo.it

ABSTRACT

The wide diffusion of smart objects and Internet of Things (IoT) is making available sensing applications that can support citizens' smart moving in outdoor contexts, as well as indoor ones. Detecting people presence and monitoring their flows could be strategic in public buildings, including shopping centers, administration offices, schools and universities. Having information about the actual occupancy of rooms in specific hours could provide useful insights for smart buildings management, that could be exploited in adequately setting the Heat, Ventilation and Air Conditioning (HVAC), the alarm, the lighting systems, and also other management issues (such as classrooms or labs assignment for different didactic activities, on the basis of the students frequency, in a smart campus). In this context, different approaches can be adopted, different technologies and sensors equipment can be installed, implying different requirements (in terms of budget), with different accuracy.

In this paper, we present a preliminary experiment aiming to detect and count people in small indoor crowded environments (such as students in a classroom). The paper describes a prototype we have designed and developed, by exploiting two different low-budget cameras. The results of an evaluation assessment are reported, by comparing the two cameras outcomes and discussing the obtained accuracy.

CCS CONCEPTS

• **Human-centered computing** → *Ambient intelligence*; • **Applied computing** → *Environmental sciences*; • **Hardware** → *Sensor devices and platforms*;

KEYWORDS

Smart sensing, Indoor Environmental Quality (IEQ), Internet of Things (IoT), Smart Environment, Occupancy Sensing

Goodtechs '19, September 25-27, 2019, Valencia, Spain

ACM ISBN 978-1-4503-6261-0/19/09.
https://doi.org/10.1145/3342428.3342691

ACM Reference Format:
Lorenzo Monti, Silvia Mirri, Catia Prandi, and Paola Salomoni. 2019. Smart Sensing Supporting Energy-Efficient Buildings: On Comparing Prototypes for People Counting. In *International Conference on Smart Objects and Technologies for Social Good (Goodtechs '19), September 25-27, 2019, Valencia, Spain.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3342428.3342691

1 INTRODUCTION

The wide diffusion of smart objects [6] and Internet of Things (IoT) opens interesting opportunities in several contexts related to humans' daily life activities [5, 13]: from smart cities [24] to smart buildings [27], from the health field ([23, 29]) to the museum/cultural heritage context [4, 7, 33]. In all these fields, sensing applications can support people's smart moving, not only in outdoor contexts [32], but also in those indoor ones [26]. In particular, in public buildings (such as public administration offices, municipalities, schools, universities, museums, etc.) detecting people's presence or monitoring their flows can be strategic for many purposes, including those ones devoted to enhance indoor staying quality [31]. Hence, in smart buildings, having information about the occupancy of rooms can provide insights for their management, so as to adequately set specific configurations, such as the Heat, Ventilation and Air Conditioning (HVAC), the alarm, the lighting, and the building security systems, as well as other management issues. In a smart campus context, integrating sensing applications with official and public open data (such as lectures time tables, enrolment data, number of students attending lectures in classrooms and activities in labs, etc.), so as to count people who are in a room, can be effective in improving management activities. In order to get sufficiently precise information in this context, using a movement sensor or a presence sensor is not enough, because they are not adequate to count and return the number of occupants, since they are not sufficiently precise and accurate [25]. To face this issue, solutions providing more accuracy are those ones based on using pictures and video-camera [25]. In this context, identifying low-budget technological solutions (in terms of hardware and software) is necessary, so as to limit installation costs in all the smart campus environments (in particular, in classrooms and in labs). Moreover, solutions compliant with privacy acts and regulations are needed.

This paper presents an experiment we have conducted with the aim of detecting and counting people who are occupying a classroom by using a sensor prototype based on two different cameras: a

Microsoft Kinect and an Intel RealSense cameras. Such a prototype has been installed in a classroom with the capacity of 100 persons, where we have collected more than 1,000 pictures during planned sessions, by involving 48 volunteers. The main purposes were to build a dataset, to test and to compare the accuracy of the cameras, with two different architectural systems: one is based on a client-server architecture, while the other one is standalone, supported by a Raspberry Pi2 model B. The results show that the Intel RealSense D415 camera reported a better accuracy, with both the architectural systems and, in particular, with the client-server based one.

The remainder of the paper is structured as follows. Section 2 briefly presents some main related work, introducing and comparing approaches that can be adopted with the aim of facing similar goals. Section 3 describes the design of the prototype, providing details about our case study, the client, the prediction and the presentation layers of the system. Section 4 illustrates the evaluation we have performed and discusses the obtained results, by comparing the two different cameras and the two different architectural approaches. Finally, Section 5 concludes the paper, disclosing some future works.

2 RELATED WORKS

Counting people is an important task used in many contexts [14] that could be related to statistical, logistics, marketing, security, and energy efficiency purposes. In literature, we can find many works aimed at tracking and counting people, that exploit different approaches, such as the ones based on the use of passive infrared sensors (PIR), which are electronic sensors that measure infrared (IR) light radiating from objects in their field of view. The combination of multiple PIR sensors (as shown in [28, 36]) has also been used to count the number of persons passing through doorways.

Another approach to estimate the number of people in specific contexts is based on the use of the Radio Frequency Identification (RFID) technology, that typically consists of three components: readers, tags, and the middleware software. RFID readers with antennas are devices used to read or write data from/to RFID tags. Most of the solutions based on such an approach exploits probabilistic estimators that achieve the required accuracy and confidence level [1, 18].

A different kind of methods is based on Wi-Fi probe-request-frame; smartphones are designed to periodically transmit these frame to determine when a known access point is within a specific distance and by capitalizing this Wi-Fi behavior. In this sense, the goal of crowd and people counting can be done thanks to monitoring and counting these Wi-Fi frames [19, 35].

Another approach to estimate the number of people within a room, is based on the use of a single carbon dioxide sensor (CO_2), starting from the assumption that an indoor environment is affected by human activities and the influence can be measured by various sensors from different aspects, to infer the density of the crowd, as described in [20, 21], or with the use of hybrid techniques, by combining different sensors such as video camera and CO_2 sensors, as proposed in [34].

Finally, the last set of methods we have taken into account is based on cameras: their main target is the design and implementation of algorithms devoted to automatically count people in two different ways. The first way is based on the recognition of video segments stream, where the counting process can be divided in two main steps: (i) the first step consists in detecting moving blob on the basis of classic algorithms for motion detection, such as *background subtraction*, and a segmentation strategy, such as *K-means*; (ii) in the second step, the detected blobs are monitored to determinate the direction of motion or to determinate the number of people present in case of a single frame shot from a camera. Two main categories of methods can be used in this context, dealing with the problem of pedestrian's counting:

- The *Line of Interest (LOI)* counting methods are designed to evaluate the number of people crossing a virtual Line of Interest within the scene studied [10, 16, 22].
- The *Region Of Interest (ROI)* counting methods are designed for crowd estimation to evaluate the number of people present within a Region Of Interest (ROI) in the studied scene, at a given time [3, 12, 38].

The second item (the Region of Interest counting methods) is based on image(s) analysis, most of these existing people counting methods can be divided into three main categories:

- *pixel-based analysis*: these methods are mostly focused on density estimation rather than the count, because they make an extensive use of local features, such as edge information or individual pixel analysis to obtain counts as described in [8, 9, 37].
- *Texture-based analysis*: it is an active topic in image processing, which plays an important role in many applications, such as image retrieval and face recognition. Rely on texture modelling through the analysis of image patches [2, 9], some texture analysis methods used in literature are grey-level co-occurrence matrix, Fourier analysis and fractal dimension as cited in [11].
- *Object-level analysis*: try to locate different type of objects in a scene [15, 17, 37].

In this work, we have adopted the last approach described in this section, as detailed in the case study presented in the following.

3 SYSTEM DESIGN

In this Section, we present the system design of our platform by detailing the entire architecture composed of different layers, as shown in figure 1, and as detailed in the following:

- *Client layer*: we have two different low-cost cameras (Intel RealSense Depth Camera D415 and Microsoft Kinect), both connected to a Raspberry Pi 2 model B each; in this case, we can capture the images and initially stored in the device. Moreover, the Raspberry Pi in our prototype has been enabled on the internal LAN of the campus network so at to upload data (over HTTPS) to the server and at the same time to externally monitor the devices (via an SSH connection).
- *Server layer*: it is divided in two distinct subsections:
  - *Prediction layer*: first of all, it is applied the prediction through the object detection system YOLOv3[1] for the counting of the people in the classroom, and after that

[1] https://pjreddie.com/darknet/yolo/

the number of predicted people is stored in a relational MySQL database by means of a script.
- *Presentation layer*: it takes care of visualizing the predicted images and of presenting the predicted number of people in the room to the users. To accomplish this task, NGINX[2] has been used as web server and reverse-proxy, and Flask[3] has been used as a Python micro-framework to implement the Web application.

In the following subsections, we will present each level in detail, after having described our experiment.

3.1 Case Study

This SubSection briefly presents the experiment we have conducted. Our case study was set up in one of the classrooms of the Campus of Cesena (University of Bologna), with a maximum capacity of 100 seats. The classroom it has been equipped with two low-budget cameras, as shown in Figure 2, which have been installed on the ceiling, near the projector, with the aim of capturing images of the people in the classroom. In particular, we have gathered the pictures, building our dataset, during special sessions, by involving 48 volunteers, properly informing them about the project.

The volunteers were invited to occupy the classroom as if they were attending a lecture, sitting in different rows and in different ways, so as to collect photos reproducing different patterns in terms of classroom occupation. The photo collection sessions involved different groups of people, so not all the pictures have the same number of people. This was done with the aim of testing the accuracy of our prototype in different contexts (in terms of room occupancy patterns and in terms of the number of occupants).

3.2 Client Layer

This layer is devoted to the data acquisition, focusing on monitoring the actual occupation of classrooms and laboratories, compared with their capacity. In order to count the number of people in an indoor area, we took advantage of two different kinds of low-budget cameras (depicted in Figure 2):

- *an Intel RealSense D415 Depth camera:* this camera is USB-powered and consists of an infrared projector, a pair of depth sensors and with an RGB Sensor, with a resolution and a frame rate equal to 1920 x 1080-pixel at 30 fps. The RealSense technologies provide a suite of depth and tracking technologies, that makes possible to count the number of people in a given area. We plugged-in (via USB) the camera to a Raspberry Pi 2 model B and we acquired 640 x 480-pixel frame rate images every five minute (this interval was set to better support the storing operations).
- *a Microsoft Kinect camera:* initially, the Kinect was developed as a gaming tool for Xbox 360. It contains three main components that work together to detect user's motions and to create her/his physical image on the screen: an RGB color VGA video camera, a depth sensor, and a multi-array microphone. As for the cameras, both the video and depth sensor have a 640 x 480-pixel resolution and run at 30 fps. We plugged-in (via USB) the camera to a Raspberry Pi 2 model B and we acquired 640 x 480-pixel frame rate images every five minutes.

[2]https://www.nginx.com/
[3]http://flask.pocoo.org

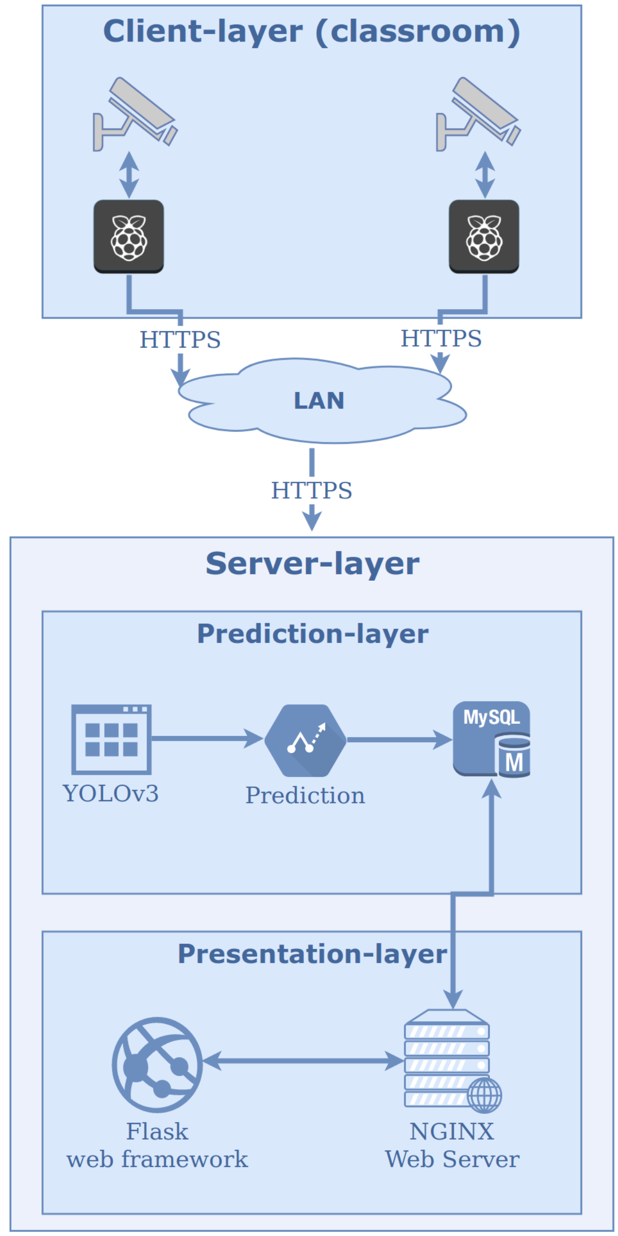

Figure 1: Our Prototype Client-Server Architecture

Each camera forwards the images to the web server, which is devoted to the prediction phase.

3.3 Prediction Layer

The prediction layer retrieves data from the cameras, on the client side, and exploits YOLOv3 with the aim of predicting the number of people in a precise moment. This tool applies a single neural network to the full image. In particular, this network divides the image into regions and it predicts bounding boxes and probabilities

Figure 2: Our video-cameras installed in the classroom

for each region. The bounding boxes are weighted by the predicted probabilities. The model reports advantages over classifier-based systems. It considers the whole image at the test time, with the aim of letting the predictions be informed about the whole context depicted in the picture. This library can perform predictions with a single network evaluation, unlike systems just like R-CNN which requires thousands for a single image. This makes it extremely fast, approximately more than 1000x faster than R-CNN and 100x faster than Fast R-CNN [30]. Once the prediction is done, we store in a MySQL database the number of people predicted and the timestamp related to when the image was taken (see Figure 1).

3.4 Presentation Layer

The presentation layer supports the data access and interaction. We implemented a rich-web base application by using standard web technologies, such as HTML5, CSS3, Javascript, etc. The back-end system has been developed as a web application, by using a Python micro-framework, called *Flask*. Lastly, we have used *NGINX* as a web server and reverse proxy to make pages available on port 80.

4 EVALUATION

In order to evaluate the overall performance of our prototype, we have conducted some test sessions by involving some volunteers. It is worth noting that such tests have been conducted in accordance with privacy and data protection regulation, and all the volunteers were adequately informed about such a research project and about the use of the images we were taking, by signing an informed consent form, specially prepared for those collection sessions.

We have conducted a preliminary evaluation by taking into account a subset of snapshots with different patterns in terms of people distribution captured in the classroom at the university campus and we have tried different scenarios, by considering different amount of people in the classroom at the time of the shots.

The accuracy and results of the proposed people counting technology are based on blob detection, by exploiting the YOLOv3 tool in two different pre-trained models: a lighter one (tiny weights), with which it's possible to compute it inside the Raspberry Pi 2, and another, more complete, one (full weights), which can be launched only by the server. The accuracy obtained with the full pre-trained model is shown in Table 1 (obtained with the Kinect camera) and 2 (obtained with the Intel camera); while the accuracy obtained with the tiny pre-trained model is shown in Table 3 (with the Kinect camera) and 4 (with the Intel camera).

Real number (**R.N.**) represents the exact number of people which were present at the time of shooting (counted by a human operator). False Counting Number (**F.C.N.**) represents the errors of the system, defining those situations when the face of a person has been counted twice, as a result of a person's movement, or by counting the upper side of the T-shirt as a face. Predicted Number (**P.N.**) represents the number of person predicted by YOLOv3 tool. To assess the accuracy of the proposed people counting system, we have exploited the following formula:

$$Accuracy(\%) = \begin{cases} \frac{\textbf{P.N.}-\textbf{F.C.N.}}{\textbf{R.N.}} * 100, & \forall\ \textbf{P.N.} \leq \textbf{R.N.} \\ \\ \frac{\textbf{R.N.}-\textbf{F.C.N.}}{\textbf{R.N.}} * 100, & \forall\ \textbf{P.N.} > \textbf{R.N.} \end{cases}$$

As shown in tables 2 in reference to 1 and 4 in reference to 3, for the same tests, the best results, with the same image resolution, were obtained by Intel RealSense D415 camera both for full weights and tiny weights. In particular, the average accuracy of Intel Realsense D415 camera is 92.2% with full pre-trained model and 40.2% with the tiny pre-trained model, and the average accuracy of Microsoft Kinect camera is 85.7% with full pre-trained model and 21.7% with the tiny pre-trained model.

Table 5 divides into three threshold by density of persons (**R.N.**) the data of the four previous tables (each of which is the representation of one of the columns), where we have considered the tests conducted with 3/4/5 people as low density, those with 10/11 people at medium density and finally those with 47/48 high-density people. In almost all cases the YOLOv3 network (both with full and tiny weights) works better in medium density (the only case is that of Intel Realsense D415 with full pre-trained mode).

From our tests we observed that YOLOv3 tends to return errors when there are people sitting close each other, turning back or

R.N.	F.C.N.	P.N.	% accuracy
4	0	4	100
3	0	2	66.6
4	0	3	75
4	0	4	100
48	1	41	83,3
47	1	36	74.5
48	0	34	70.8
10	0	10	100
10	2	12	80
5	0	4	80
11	1	12	90.1
10	0	9	90
10	0	10	100
5	0	4	90
AVERAGE			**85.7**

Table 1: Microsoft Kinect camera accuracy with full pre-trained model

R.N.	F.C.N.	P.N.	% accuracy
4	0	4	100
3	0	2	66.6
4	0	4	100
4	0	4	100
48	0	48	100
47	0	47	100
48	0	45	93.7
10	0	10	100
10	1	11	90
5	2	7	60
11	0	11	100
10	2	10	100
10	2	12	80
5	0	5	100
AVERAGE			**92.2**

Table 2: Intel Realsense D415 camera accuracy with full pre-trained model

bending over. Moreover, regarding the cameras position (in perspective), this method slightly suffers from partial occlusion of the blob determined by the overlap in perspective of people in a row in the desks.

5 CONCLUSION AND FUTURE WORKS

In this paper, we present a prototype for counting the number of people in a specific context: a University Campus. The proposed strategy is based on exploiting pictures obtained from two different sensors, a Microsoft Kinect camera and an Intel Realsense D415. Each of them has been positioned in perspective to cover all the classroom desks.

The proposed method is based on the design of two main modules, the client-layer in which images were taken, stored and sent to the server-layer in which the number of people are first predicted using the YOLOv3 tool and then stored the predicted number on a MySQL database. As future work, we intend to try a different kind of camera positions, such as change the position of the cameras placing them at the bottom of the classroom and take pictures of the person's backs, or the triangulation of the cameras from three

R.N.	F.C.N.	P.N.	% accuracy
4	0	0	0
3	0	0	0
4	0	1	25
4	0	1	25
48	0	4	8.3
47	0	4	8.5
48	1	7	12.5
10	0	5	50
10	0	5	50
5	0	0	0
11	0	6	54.5
10	0	5	50
10	0	2	20
5	0	0	0
AVERAGE			**21.7**

Table 3: Microsoft Kinect camera accuracy with tiny pre-trained model

R.N.	F.C.N.	P.N.	% accuracy
4	0	0	0
3	0	2	66.6
4	0	3	75
4	0	0	0
48	2	12	20.8
47	3	13	21.3
48	2	8	12.5
10	0	7	70
10	1	6	50
5	0	3	60
11	1	5	36.4
10	3	11	70
10	0	4	40
5	0	2	40
AVERAGE			**40.2**

Table 4: Intel RealSense D415 camera accuracy with tiny pre-trained model

	Kinect/Full	Kinect/Tiny	Intel/Full	Intel/Tiny
low	85,2	8,3	87,7	40,3
medium	**92,02**	**44,9**	94	**53,3**
high	76,2	9,7	**97,9**	18,2

Table 5: Accuracy per different density rates

different points of the classroom in order to minimize false positives and false negatives. Another issue is about of using depth images that each camera already has a sensor for this purpose, and exploit also a thermal camera (we are currently testing a thermal FLIR Lepton model) in order to solve the privacy problems. We are also working on re-train the model with the dataset taken from the special session with the people as already described in Case Study section, focused on a specific class to improve the performance of the model with tiny weights in order to implement the prediction directly on Raspberry Pi.

ACKNOWLEDGMENTS

The authors would like to thank Massimo Donini and Luca Rispoli for their precious support and valuable contribution to the implementation. Many thanks also to Chiara Ceccarini, Davide Aguiari, and Prof. Giovanni Pau.

REFERENCES

[1] Rebecca Angeles. 2005. RFID technologies: supply-chain applications and implementation issues. *Information systems management* 22, 1 (2005), 51–65.

[2] Ankan Bansal and KS Venkatesh. 2015. People counting in high density crowds from still images. *arXiv preprint arXiv:1507.08445* (2015).

[3] Enrico Bondi, Lorenzo Seidenari, Andrew D Bagdanov, and Alberto Del Bimbo. 2014. Real-time people counting from depth imagery of crowded environments. In *2014 11th IEEE International Conference on Advanced Video and Signal Based Surveillance (AVSS)*. IEEE, 337–342.

[4] Armir Bujari, Matteo Ciman, Ombretta Gaggi, and Claudio E Palazzi. 2017. Using gamification to discover cultural heritage locations from geo-tagged photos. *Personal and Ubiquitous Computing* 21, 2 (2017), 235–252.

[5] Armir Bujari, Marco Furini, Federica Mandreoli, Riccardo Martoglia, Manuela Montangero, and Daniele Ronzani. 2018. Standards, Security and Business Models: Key Challenges for the IoT Scenario. *Mobile Networks and Applications* 23, 1 (01 Feb 2018), 147–154. https://doi.org/10.1007/s11036-017-0835-8

[6] Armir Bujari, Claudio E Palazzi, and Daniele Ronzani. 2017. FANET application scenarios and mobility models. In *Proceedings of the 3rd Workshop on Micro Aerial Vehicle Networks, Systems, and Applications*. ACM, 43–46.

[7] Juan-Carlos Cano, Pietro Manzoni, and C-K Toh. 2006. Ubiqmuseum: A bluetooth and java based context-aware system for ubiquitous computing. *Wireless Personal Communications* 38, 2 (2006), 187–202.

[8] Antoni B Chan, Zhang-Sheng John Liang, and Nuno Vasconcelos. 2008. Privacy preserving crowd monitoring: Counting people without people models or tracking. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 1–7.

[9] Antoni B Chan and Nuno Vasconcelos. 2012. Counting people with low-level features and Bayesian regression. *IEEE Transactions on Image Processing* 21, 4 (2012), 2160–2177.

[10] Luca Del Pizzo, Pasquale Foggia, Antonio Greco, Gennaro Percannella, and Mario Vento. 2016. Counting people by RGB or depth overhead cameras. *Pattern Recognition Letters* 81 (2016), 41–50.

[11] Shervan Fekri-Ershad. 2018. A Review on Image Texture Analysis Methods. *arXiv preprint arXiv:1804.00494* (2018).

[12] Alvaro Fernandez-Rincon, David Fuentes-Jimenez, Cristina Losada-Gutierrez, Marta Marrón Romera, Carlos A Luna, Javier Macías Guarasa, and Manuel Mazo. 2017. Robust People Detection and Tracking from an Overhead Time-of-Flight Camera.. In *VISIGRAPP (4: VISAPP)*. 556–564.

[13] Marco Furini, Federica Mandreoli, Riccardo Martoglia, and Manuela Montangero. 2017. *IoT: Science Fiction or Real Revolution?* Springer International Publishing, 96–105. https://doi.org/10.1007/978-3-319-61949-1_11

[14] Jungong Han, Eric J Pauwels, Paul M de Zeeuw, and Peter HN de With. 2012. Employing a RGB-D sensor for real-time tracking of humans across multiple re-entries in a smart environment. *IEEE Transactions on Consumer Electronics* 58, 2 (2012), 255–263.

[15] Haroon Idrees, Imran Saleemi, Cody Seibert, and Mubarak Shah. 2013. Multi-source multi-scale counting in extremely dense crowd images. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2547–2554.

[16] Rabah Iguernaissi, Djamal Merad, and Pierre Drap. 2018. People Counting based on Kinect Depth Data.. In *ICPRAM*. 364–370.

[17] Minjin Kim, Wonjun Kim, and Changick Kim. 2011. Estimating the number of people in crowded scenes. In *Visual Information Processing and Communication II*, Vol. 7882. International Society for Optics and Photonics, 78820L.

[18] Murali Kodialam and Thyaga Nandagopal. 2006. Fast and reliable estimation schemes in RFID systems. In *Proceedings of the 12th annual international conference on Mobile computing and networking*. ACM, 322–333.

[19] Kai Li, Chau Yuen, Salii S Kanhere, Kun Hu, Wei Zhang, Fan Jiang, and Xiang Liu. 2018. Understanding crowd density with a smartphone sensing system. In *2018 IEEE 4th World Forum on Internet of Things (WF-IoT)*. IEEE, 517–522.

[20] Tengyue Li and Simon Fong. 2018. Counting Passengers in Public Buses by Sensing Carbon Dioxide Concentration: System Design and Implementation. In *Proceedings of the 2018 2nd International Conference on Big Data and Internet of Things*. ACM, 218–221.

[21] Salvatore Longo and Bin Cheng. 2015. Privacy preserving crowd estimation for safer cities. In *Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers - UbiComp '15*. ACM Press. https://doi.org/10.1145/2800835.2801631

[22] Carlos A Luna, Cristina Losada-Gutierrez, David Fuentes-Jimenez, Alvaro Fernandez-Rincon, Manuel Mazo, and Javier Macias-Guarasa. 2017. Robust people detection using depth information from an overhead Time-of-Flight camera. *Expert Systems with Applications* 71 (2017), 240–256.

[23] Simone Mangano, Hassan Saidinejad, Fabio Veronese, Sara Comai, Matteo Matteucci, and Fabio Salice. 2015. Bridge: Mutual reassurance for autonomous and independent living. *IEEE Intelligent Systems* 30, 4 (2015), 31–38.

[24] Andrea Melis, Silvia Mirri, Catia Prandi, Marco Prandini, Paola Salomoni, and Franco Callegati. 2016. Crowdsensing for smart mobility through a service-oriented architecture. In *2016 IEEE International Smart Cities Conference (ISC2)*. IEEE, 1–2.

[25] Hessam Mohammadmoradi, Sirajum Munir, Omprakash Gnawali, and Charles Shelton. 2017. Measuring People-Flow Through Doorways using Easy-to-Install IR Array Sensors. In *2017 13th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE, 35–43.

[26] Lorenzo Monti, Catia Prandi, and Silvia Mirri. 2018. Iot and data visualization to enhance hyperlocal data in a smart campus context. In *Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good*. ACM, 1–6.

[27] Catia Prandi, Lorenzo Monti, Chiara Ceccarini, and Paola Salomoni. 2019. Smart Campus: Fostering the Community Awareness Through an Intelligent Environment. *Mobile Networks and Applications* (2019), 1–8.

[28] Yordan P. Raykov, Emre Ozer, Ganesh Dasika, Alexis Boukouvalas, and Max A. Little. 2016. Predicting room occupancy with a single passive infrared (PIR) sensor through behavior extraction. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing - UbiComp '16*. ACM Press. https://doi.org/10.1145/2971648.2971746

[29] Roberto Reda, Filippo Piccinini, and Antonella Carbonaro. 2018. Towards consistent data representation in the IoT healthcare landscape. In *Proceedings of the 2018 International Conference on Digital Health*. ACM, 5–10.

[30] Joseph Redmon and Ali Farhadi. 2018. YOLOv3: An Incremental Improvement. *arXiv* (2018). https://pjreddie.com/media/files/papers/YOLOv3.pdf

[31] Marco Roccetti, Stefano Ferretti, Claudio E Palazzi, Paola Salomoni, and Marco Furini. 2008. Riding the web evolution: from egoism to altruism. In *2008 5th IEEE Consumer Communications and Networking Conference*. IEEE, 1123–1127.

[32] Julio Sanguesa, Manuel Fogue, Piedad Garrido, Francisco Martinez, Juan-Carlos Cano, Carlos Calafate, and Pietro Manzoni. 2013. An infrastructureless approach to estimate vehicular density in urban environments. *Sensors* 13, 2 (2013), 2399–2418.

[33] Rita Tse, Davide Aguiari, Ka-Seng Chou, Su-Kit Tang, Daniele Giusto, and Giovanni Pau. 2018. Monitoring cultural heritage buildings via low-cost edge computing/sensing platforms: the Biblioteca Joanina de Coimbra case study. In *Proceedings of the 4th EAI International Conference on Smart Objects and Technologies for Social Good*. ACM, 148–152.

[34] Fulin Wang, Qingqing Feng, Zheliang Chen, Qianchuan Zhao, Zhijing Cheng, Jianhong Zou, Yufeng Zhang, Jinbo Mai, Yun Li, and Hayden Reeve. 2017. Predictive control of indoor environment using occupant number detected by video data and CO 2 concentration. *Energy and Buildings* 145 (June 2017), 155–162. https://doi.org/10.1016/j.enbuild.2017.04.014

[35] Ooi Boon Yaik, Kong Zan Wai, Ian KT Tan, and Ooi Boon Sheng. 2016. Measuring the accuracy of crowd counting using Wi-Fi probe-request-frame counting technique. *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)* 8, 2 (2016), 79–81.

[36] Jaeseok Yun and Sang-Shin Lee. 2014. Human Movement Detection and Identification Using Pyroelectric Infrared Sensors. *Sensors* 14, 5 (May 2014), 8057–8081. https://doi.org/10.3390/s140508057

[37] Tao Zhao, Ram Nevatia, and Bo Wu. 2008. Segmentation and tracking of multiple humans in crowded environments. *IEEE transactions on pattern analysis and machine intelligence* 30, 7 (2008), 1198–1211.

[38] Lei Zhu and Kin-Hong Wong. 2013. Human tracking and counting using the kinect range sensor based on adaboost and kalman filter. In *International Symposium on Visual Computing*. Springer, 582–591.