People Counter with Area Occupancy Control for Covid-19

E. Khoumeri\textsuperscript{1(✉)}, H. Fraoucene\textsuperscript{2}, El Hadi Khoumeri\textsuperscript{1,2},
C. Hamouda\textsuperscript{1,2}, and R. Cheggou\textsuperscript{1}

\textsuperscript{1} Laboratoire des Technologies Innovantes, GEII Department,
Ecole Nationale Supérieure de Technologie, Algiers, Algeria
{elhadi.khoumeri, rabea.cheggou}@enst.dz
\textsuperscript{2} Laboratory of Advanced Technologies of Genie Electrics, Faculty of Electrical and Computer Engineering, Mouloud Mammeri University, Tizi-Ouzou, Algeria
henia.fraoucene@ummtodz

Abstract. Governments have imposed social distancing regulations to counter the coronavirus pandemic. An automated occupancy control system to provide a more cost-effective and efficient way to abide with these safety regulations. With the generalization of the use of digital images, the analysis of movement in video sequences has proved to be an essential tool for various applications such as video surveillance, robotics etc. The advance in video processing algorithms and the fast computational capability, give a possibility to use a video tracking and counting people in real time. Estimating the number of people in real time is useful information for several applications such as security and health management. With the COVID-19 pandemic, the counting of people present in a region of interest is important to control the area occupancy in order to minimize human virus transmission. In this paper we present a finished solution for counting people present in the same area. The system is based on Raspberry Pi and a common camera. The method called BLOB (Binary Large Object) analysis is used. The performance of the system, achieving an average count rate between 95% and 98%.

Keywords: People counting · COVID-19 · OpenCv · Occupancy control · Raspberry Pi · BLOB

1 Introduction

Today, a lot of research has been published to try to solve the problem of counting people using a video camera. This is not a simple task, there are some difficult issues to solve. One of the problems with using the video camera for people counting is the occlusion between people, when a group enters or leaves the field of view of the camera. It is very difficult to distinguish between people in the same group. Existing solutions offering video cameras based on people counting are efficient and reliable but very expensive. This work proposes an efficient and inexpensive solution for counting people and discusses the problem of detecting overlapping people shapes in the video. In this project, we will propose the design and an implementation of a counting
algorithm based on image processing using the Raspberry-Pi in real time. The extraction of the characteristics of an image containing the objects to be detected is given using a method called BLOB (Binary Large Object) analysis [1]. The number of people counted is displayed on the screen directly. The Raspberry Pi, running with the Linux operating system installed on an SD memory card, a video camera used as an image acquisition device to capture it continuously. The coronavirus transmission mechanisms identified in [2] this prompted us to find a way to limit close face-to-face contact in closed area. Our system accurately counts the number of people entering and leaving the closed area, giving the number of customers inside in real time. A tablet installed at the entrance shows people if they can enter or if the maximum occupancy has been reached and they must wait. When other people leave, the screen updates accordingly to the occupancy situation. This allows an effective occupancy control for essential businesses such as supermarkets, pharmacies, banks and hospitals as well as for government institutions.

1.1 Internet of Things

The term Internet of Things (IoT) was first introduced in 1999 [3], IoT is an essential part of modern embedded systems that describes a network of physical objects connected to the internet in order to communicate and share data between them. The goal of IoT is to make use of these large amounts of data to make systems more efficient and automated. There are several examples of IoT solutions for various problems, such as using light beams and infrared sensors to analyze the occupancy level of a house and to adjust the airflow of the ventilation system [4], using video and location data to enhance counting the number of people in an area [5], and for fire safety in buildings [6]. The Internet of Things is also used for remote patient health management with the sending of vital information directly to the attending physician [7].

1.2 Why People Counting?

Due to the coronavirus pandemic, travel is limited and people are advised to stay at home. At the time of deconfinement, the medical authorities described distancing protocols. The idea of making a person counter at the entrance of a building welcoming people was born from the limitation of the number of people present at the same time in a closed area. People have to do their shopping, go to a doctor or to the bank. To monitor occupancy with the number of people present we need an automated system, our application will display the number of people entering and leaving so, the person managing security can be sure that they meet the requested health standards.

1.3 Literature Survey

The great difficulty of counting people by camera is knowing how to count a person hidden behind another person. Different research has been proposed, as in [8] and [9] Uses method for facial recognition to detect people. with a single camera, the system counts the number of people. Counting is done by analyzing the image to detect faces. Ching-Tang and All [10] offered a bi-directional counting system for the flow of people
with Kinect. They put the Kinect camera above the door to capture the situation of the pedestrian flow. Then, this system detects people in the coverage area using the depth image information from the Kinect system. Borislav [11] presents an efficient and reliable method of automatic segmentation, tracking and counting of people, designed for a system with an aerial (overhead) camera. [12] This work presents an implementation of a people counting system based on a modified gradient histogram (HoG), a support vector machine (SVM) and a novel counting technique using an FPGA implementation. Another automatic bidirectional method [13] for counting passers-by based on image processing. To reduce the overall overlap, a camera set with a straight down view is adopted. And an experimental formula based on the area is adopted to derive the rough estimate of pedestrians with simple and fast tracking. An algorithm is used to track pedestrians. Using this, track information, merge and split, cases can be easily judged. Some companies [14, 15] and [16] sell products similar to our application, we differentiate ourselves from these products, by their prohibitive prices which sometimes exceed 1000 dollars for one system [17].

2 Methodology

2.1 The Characteristics of the Human Body

We can recognize a human by using different information. We see the silhouette of a person with our eyes, we will collect other types of information that will confirm that it is a person. The person will undoubtedly breathe, blink, so to identify a person, we can use technologies that provide information that will identify the person [18]. Although each individual is different and even unique, we all have in common our morphology. A person normally made up of two arms, two legs, a head, etc. which implies that a human silhouette can be differentiated from the silhouette of an animal.

2.2 The People Counting Algorithm

Design is the process of representing the various functions of the system, so one or more programs quickly perform those functions. This is the most important part of our job. In the following we will describe an algorithm for counting people in a video sequence, first we describe the design of our system then we implement it, then we present the results. Our goal is to design a system for counting people entering and leaving a building or an area. The general scheme of the design of our system is represented by the following flowchart:

Referring to (Fig. 1), the system detects pedestrians passing in a visual field, the video stream is captured with a camera placed on the ceiling with a top view, this video is split into several frames (images), one image per 30 ms, each image is processed separately so that people can be detected. For the people tracking part, the system follows people in the video sequences by memorizing their coordinates and their movement (entering or leaving). In the part counting people we will count the number of people entering and leaving.
2.3 Detection Algorithm

The people detection is summarized in the algorithm (Fig. 4), in this algorithm we find the following functions:

**Video (Picture):** stream playback by a webcam located in the ceiling with top view, as shown in Fig. 3.

**Transformation:** transform images to grayscale,

**Filtering:** in this step we will apply a Gaussian filter to eliminate the noise in the image in order to have better detection,

**Detection of the Background:** memorization of the first image that we will consider it as the background,

**Background Subtraction:** the subtraction of each frame from the recorded background,

**Threshold:** binarize the image to have a better result, transforming the gray level image into a black and white image such that the parts in white are moving as shown in (Figs. 2 and 3).
Find Contours: detected all contours of the borders of moving objects,

For Each Contour C in the Set of Contours: the area of each contour is calculated if it is outside the detection interval (if less than min or greater than the max as indicated in the algorithm, it will be eliminated). Then, we find the contour characteristics (person) (x coordinate, y coordinate, width and height),

Calculate the Center of Gravity Cg: for each contour, we memorize its coordinates x (Cx) and y(Cy), see (Fig. 5).
2.4 Tracking Algorithm

It is proposed as showing in Fig. 6:

After having determined the coordinates Cx and Cy, we store them in two tables cx-current and Cy-current. In the following image, if the distance between the center of gravity found Cg (cx and cy) and the points stored in Cx-current and Cy-current is less than a threshold (threshold which determines the step of a normal person who is about 45 cm, which was found by tests), then We consider that the person detected in this
image is the same as that detected in the previous image; otherwise, if the distance is great, we consider that this person is a new person who appears in the video and we memorize these coordinates.

2.5 Counting Algorithm

The algorithm is presented in Fig. 7

This algorithm uses two virtual lines (the entry and exit direction). We take H as the height of the image and L as the width (see Fig. 7). For each contour (a person), which will be detected, has a center of gravity $C_g$ $(C_x, C_y)$. To know if the detected person will enter or will leave, we proceed as follows: if $C_y$ is greater than $H/2$, the person is consigned as outgoing, however, if $C_y$ is less than $H/2$, the person will enter (Fig. 8).
To be able to count exactly the number of people, after each disappearance of a contour which had already existed in the previous image, the coordinates of Cg are compared with two thresholds Min and Max, these two thresholds determine the precision of the entry or of the exit of persons, if Cy is less than the Min threshold, then the person is considered to be outgoing, otherwise if Cy is greater than the Max threshold, then the person is considered as entering.

3 Experimental Results

Our system is developed under the Raspberry Pi 2 environment which meets the needs of our application, its characteristics: CPU Quad cortex A7 900 MHz, 1G RAM, GPU 250 MHz Video Core IV, for a price of 40 dollars. For the implementation of algorithms we have chosen the Python language and for image processing we have used functions from the OpenCV library. We used a webcam for a price of $10. Note that we can use any camera.

A tablet is connected to the Raspberry-Pi via Wi-Fi to display the authorization to enter the store, the number of people present in the store in this case has been set at 10. Once this threshold is reached, the tablet displays STOP, if a person goes out, so there will be 9 people, the tablet displays GO, it gives permission for a person to enter inside. The system calculates the number of people entering minus the number of people leaving to define the number of people present in the store. Once the number of people inside is 10, it displays STOP again.

To test our system, we placed the webcam at the entrance of a building as in Fig. 9. It shows the principle of detecting a moving person by keeping its center of gravity with the calculation of the difference between the images (current and old image).
Figure 9. Detection of a person going out.

Figure 10 shown, when a person goes out, the OUT counter is incremented by +1.

Fig. 10. Counter increase.
Another case presented in Fig. 11a and b, where two people enter at the same time.

![Fig. 11. a. Two persons detected. b. Counter increased +2.](image)

We put the tablet at the store entrance, to regulate the entrance. If the number of people is less than ten in our case, the system displays the GO message on the tablet in green, as in Fig. 12a. If the number of people reaches exactly ten, the system displays the STOP message in red on the tablet as in Fig. 12b.

![Fig. 12. a. Occupancy less than ten. b. Occupancy reached ten people.](image)

4 Conclusion

This article presents a simple and automatic method based on bi-directional processing to count people. It offers an inexpensive solution to manage the flow of visitors to shops and buildings. To reduce the overall overlap, a single camera set to a downward view is adopted. An experimental formula based on area was used to obtain a pedestrian body estimate by determining the center of gravity Cg with the fast tracking algorithm to track pedestrians. Using this information, merging and splitting cases can be handled quickly. This system has already been applied in a shopping center. Based on these experimental results, he showed that the proposed method works very reliably and achieves high accuracy rates. Different approaches can be used to improve the system; like designing a database to display input/output statistics, adding other cameras to improve counting.
References

1. Cazasnoves, A., et al.: Statistical content-adapted sampling (SCAS) for 3D computed tomography. Comput. Biol. Med. 92, 9–21 (2018)
2. Chamola, V., et al.: A comprehensive review of the COVID-19 pandemic and the role of IoT, drones, AI, blockchain, and 5G in managing its impact. IEEE Access 8, 90225–90265 (2020)
3. Suresh, P., et al.: A state of the art review on the Internet of Things (IoT) history, technology and fields of deployment. In: International Conference on Science Engineering and Management Research (ICSEMR), Chennai, pp. 1–8 (2014)
4. Cheggou, R., et al.: Energy-saving through smart home concept. In: International Conference in Artificial Intelligence in Renewable Energetic Systems, pp. 50–58. Springer, Cham, March 2017
5. Stenbrunn, A.: Hosting a building management system on a smart network camera: on the development of an IoT system. Bachelor thesis, Malmo University, November 2015
6. Khoumeri, E., et al.: An IoT system for smart building. In: Future of Information and Communication Conference, pp. 522–533. Springer, Cham, December 2018
7. Khoumeri, E., et al.: An IoT based e-health platform using Raspberry Pi. In: Future of Information and Communication Conference, pp. 257–271. Springer, Cham (2020)
8. Chen, T.Y., et al.: A people counting system based on face-detection. In: Proceedings of the Fourth International Conference on Genetic and Evolutionary Computing, pp. 699–702, December 2010
9. Surbhi, S., Dalpat, S.: Design of people counting system using MATLAB. In: Proceedings of Tenth International Conference on Contemporary Computing (IC3), August 2017
10. Heish, C.-T., et al.: A kinect-based people-flow counting system. In: IEEE International Symposium on Intelligent Signal Processing and Communication Systems (ISPACS 2012), pp. 146–150, November 2012
11. Antić, B., Letić, D., Ćulibrk, D., Crnojević, V.: K-means based segmentation for real-time zenithal people counting. In: 2009 16th IEEE International Conference on Image Processing (ICIP), Cairo, pp. 2565–2568 (2009)
12. Ahmad, I., et al.: An FPGA based approach for people counting using image processing techniques. In: 2019 11th International Conference on Knowledge and Smart Technology (KST), Phuket, Thailand, pp. 148–152 (2019)
13. Cao, J., Sun, L., Odoom, M.G., Luan, F., Song, X.: Counting people by using a single camera without calibration. In: 2016 Chinese Control and Decision Conference (CCDC), Yinchuan, pp. 2048–2051 (2016)
14. Covid-19 Automated Occupancy Control System. FootfallCam. https://www.footfallcam.com/Industries/Covid-19-Automated-Occupancy-Control-System. Accessed 28 July 2020
15. Digital versus COVID 19. Digital versus COVID 19 - People counting: Alphabird GmbH. https://www.alphabird.at/en/. Accessed 28 July 2020
16. People counter project for Covid-19. https://www.rs-online.com/designspark/people-counter-project-for-covid-19. Accessed 28 July 2020
17. People Counting Camera: 3D SCOPE II. StoreTraffic, 06 July 2020. https://storetraffic.com/product/3d-scope-ii-people-counting-camera/. Accessed 28 July 2020
18. Belconde, A.: Modélisation de la détection de présence humaine. PhD thesis, University of Orleans, France, September 2010