PEDESTRIAN COUNTING SYSTEM IN REAL TIME IMAGE BASED ON HAAR-CASCADE CLASSIFIER

[1]DONGHWAN KU, [2]HAE-WON CHOI, [3]HYUNSUNG KIM

[1]Dept. of Cyber Security, Kyungil University, Korea
[2]Dept. of Computer Engineering, KyungwoonUniversity, Korea
[3]Dept. of Mathematical Sciences, University of Malawi, Malawi
E-mail: [1]imcman5@naver.com, [2]kim@kiu.ac.kr

Abstract - Effective personnel management is requiredfor the success of modern business. Pedestrian counting is one of key factors in the personnel management. When it is performed manually, there are possibility of counting errors and even this could lead high labor costs. This paper proposes an automated pedestrian counting system in real time image based on Haar-Cascade classifier. The proposed system not only provides the counted number of pedestrians but also ensures better performance compared with the other systems. All processes in the proposed system are performed based on images only over Raspberry Pi with a camera module. Thereby, it is possible to implement a high-performance pedestrian counting system at lower cost.

Index Terms - Cascade classifier, Haar-like features, Image processing, Pedestrian counting

I. INTRODUCTION

To know how many customers visit the shop is critical for making the business decisions. It may be used as an index of shopping popularity in department stores or hypermarkets, as well as helping managers position the deployment of personnel [1]–[4]. In addition, the number of customers of a year is also used to predict the number of customers at peak times. In the exhibition or stadium where entrances are set, security personnel count the number of visitors through the entrance. Therefore, the information of the number of people through the entrance is very important and has a high reference value [5]. Although counting manually at the entrance has high accuracy, the human resource department should pay high costs to hire labors. Also, if the labors are counting for a long time, they may be tired and miscount the number of people. To solve this problem, several automatic people counting devices were presented, such as mechanical shaft device. It is not only able to save the cost of human resource, but also assure the competent accuracy of counting. However, it has a disadvantage that only one person can pass through the device at one time. Therefore, this device is not suitable for the pedestrian counting.

In the automatic passenger counting, technologies using infrared sensors have also emerged [6]–[8]. It detects the passage of a person by using the interruption of the infrared signal to count the number of people. The movement direction of the passenger can be determinedby comparing the difference in interrupting time between two infrared sensors. However, when many people pass the entrance at the same time, its accuracy is not good. Furthermore, the system cannot obtain any other information if only infrared sensors are used. Recently, the computation speed of a computer has become very fast [9]–[11]. Camera attached to the computer allows us to exchange each other's images in real time, and the surveillance cameras for the security automatically send its image information to the security control room [12]–[14]. As a result, more and more researchers are interested in the area of computer image processing.

The purpose of this paper is to propose an automated pedestrian counting system in real time camera image based on Haar-Cascade classifier. There are fouradvantages of this research.

1) It can provide high quality images, high performance, high accuracy and high reliability that can monitor and control the completion of the work compared to the infrared sensing devices.
2) Installing a camera is economical and convenient. Addition to this,it is easy to maintain.
3) The information about crowed people are obtained from the camera and they are transferred to the computers over Internet. This can achieve the realtime remote management
4) It includes real time, reliable, and security requirements.

II. PRELIMINARIES

In this section, methods for human detection are provided. Detection process of our proposed system is based on Haar-like feature and cascade classifier proposed by Viola and Jones [15]–[16].

A. Haar-like Feature

In an area of human face detection, Viola and Jones first proposed a way of detecting objects using features to treat the faces of a wide variety of people[16]. In their algorithm, the simple Haar-like features are used

to detect human face. There are two motivations for adoption of the Haar-like features rather than pixel values. The first reason is that the Haar-like features can encode ad-hoc domain knowledge, which is difficult to describe using a finite quantity of training data. Compared with raw pixels, the Haar-like features can efficiently reduce or increase the in-/out-of-class variability and thus making classification easier. And there is also the other reason: the operation of feature-based system is much faster than that of pixel-based system. These reasons cannot be ignored in terms of the overhead of the operation and that's why the features are used in object detection where limited time is required.

Haar-like features are digital image features for recognize similar objects in an image.Each feature is composed of two or three "black" and "white" rectangles and they are horizontally or vertically adjacent each other where these rectangles can be located by 45 degrees. The value of Haar-like feature is calculated as a weight that is the difference between the sums of the pixel intensities within black and white rectangles. The weight of components are the opposite and their absolute values are inversely proportional to the areas. Fig.1 shows the prototypes of Haar-like features.

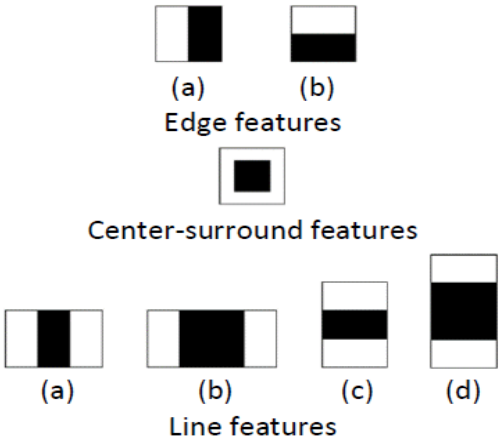


Fig.1.Aset of Haar-like features[16]

A number of features are used in the classifier for use in detection. However, direct computation of pixel sums through multiple rectangles would make the process very slow and is not appropriate for the real-time process. Viola and Jonesproposed an efficient way to compute the sums rapidly. The integral image at location x,y can be solved as the equation (1), where ii(x,y) is the integral image and I(x',y') is the original image.

$$ii(x,y) = \sum_{x' \le x, y' \le y} I(x', y') \quad (1)$$

B. Cascade Classifier

A cascade classifier consisting of stages can improve detection performance while radically reducing computation time[15]. Each stage of the classifier is made up of weak learners, and the higher the level, more learners are used to derive specific object. The system detects objects by moving a window over the whole image. Each stage of the classifier labels the location of current window as either positive or negative. The positive means that a specified object was found and the negative means that it wasnot found in the image. If the labelling gives a negative result, then the classification of this window is completed, and the detection window is moved to the next location. If the labelling represents a positive result, then the classification moves to the next level. The classifier finally yields a verdict of positive if all stages, including the last one, yield a result and the object is found in the image.

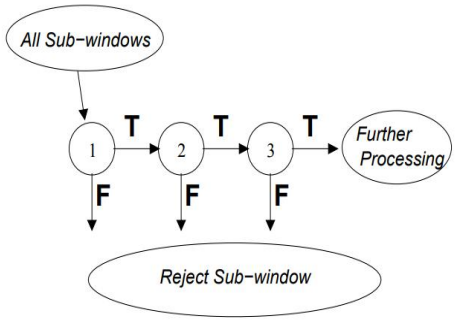


Fig.2. Stages of the cascade classifier[15]

A true positive means that the object is included in the image and the classifier labels it exactly. A false positive means that the object is misjudged as being in the image, although it is not.A true negative means that a non-object was classified as not being the specified object and a false negative occurs when the classifier could not detect the specified object in the image. There is no way to correct the mistakes if the classification is wrong during the execution of the process. Therefore,a low false negative rate should be guaranteed at each stage of the cascade in order to work well.However, each stage can have a relatively high false positive rate, because even if the n-th stage classifies the non-object as actually being the object, then this mistake can be fixed in n+1-th and subsequent stages of the classifier[15].

III. PROPOSED PEDESTRIAN COUNTING SYSTEM

In this section, we propose a pedestrian counting system in real time camera image. The proposed system is based on Haar-like feature and cascade classifier[15]–[16].To count pedestrians in real time image, the proposed system consists of three phases: pedestrian detection phase, pedestrian counting phase and data transmission phase. The flow chart of our proposed system is shown in Fig.3.

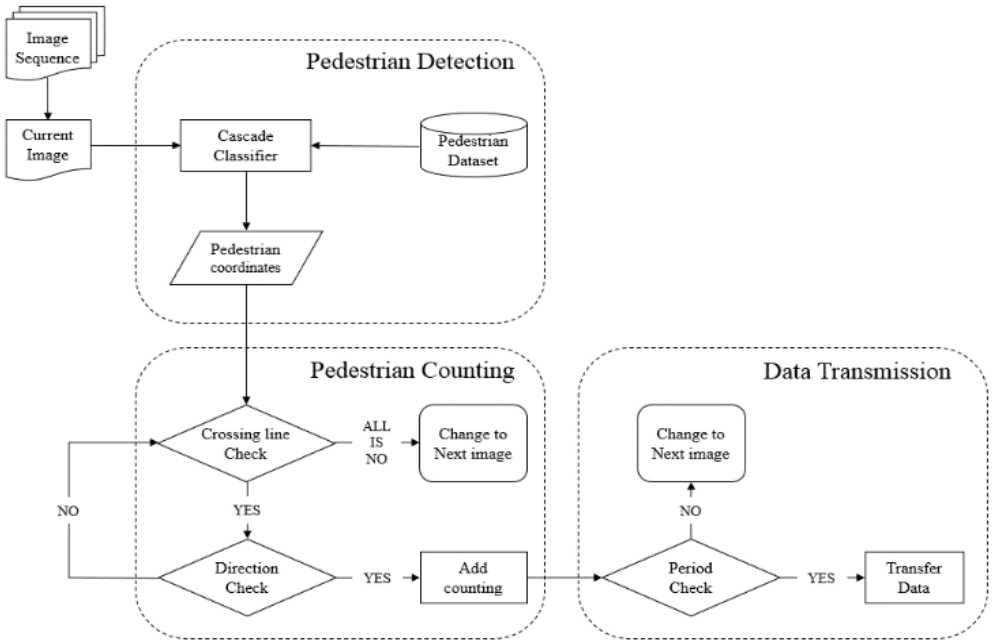


Fig.3. Flow chart of the proposed system

The first step is to detect a region of pedestrian from the input image. In the detection phase of pedestrian, we use a classifier named cascade proposed by Viola and Jones in [15]. This classifier will detect whether a pedestrian is in the regionof detection window or not based on the dataset learned with Haar-like feature from [16]. Then,pedestrian counting phase starts when the region is detected as a pedestrian passes the "crossing line". The crossing lineshould bepreset for the system. Checking if the pedestrian is passing the crossing line or not can be figured out by comparing coordinate of the line and that of the pedestrian. The problem to be solved in this process is how to detect any pedestrian who has only one-way moving direction.Sincethere are already a number of detection algorithms, weonly focus on our research to the pedestrian counting phase.When this phase is used several times, the accumulated counting data will be transferred to the central serverperiodically.

C. Pedestrian Detection

Before detectingthe region of pedestrian for counting, suppose that we have a dataset with Haar-like features which present the features of pedestrian. This dataset is necessary for detecting pedestrians by using cascade classifier and is a set of data learned in advance with learning algorithms such as AdaBoost, which selects asmall number of critical visual features from a larger set and yields extremely efficient classifiers[17].

After getting an image from input image sequence, the system needs to convert it to grayscale. The region of pedestrian will be detected by using cascade classifier. This classifier detects the region of pedestrian with a few prototypes of Haar-like features and as the classifier processes higher level, the number of prototypes used in classifier increases gradually.Through these procedures, the region selected as a pedestrian that matches Haar-like features can be found. With this region, the system calculates the center coordinate of itto check whether it is passing the crossing line or not.

Our final goal of the proposed system is to count pedestrians who enters to a specific area. This means thateven though a pedestrian appears in the image, the system should not count him(or her) if he does not enter predefined area, such as an entrance of a building. To achieve this goal, we present a criterion namedcrossing line, which is a straight line perpendicular to the path of the pedestrian in the image. Be aware of that the crossingline lies on the entrance and is not a physical line, but a virtual linefor counting people. If the path of pedestrian is vertical, it is horizontal and can be a constant for x, if it is horizontal, it is vertical and can be a constant for y. Therefore, according to the direction of the path and entrance, 4 cases can be occurred: (1) horizontal path and left direction, (2) horizontal path and right direction, (3) vertical path and upward direction, and(4) vertical path and downward direction. Each case is individually handled by each process. With the notion of pedestrians coordinates in previous and current images, we will provide the process of determining whether he (or she) can be counted or not as provided in the following categories.

D. Pedestrian Counting

[Horizontal Path, Left Direction] There are two conditions for determining a valid pedestrian. One is whether a pedestrian is involved with the crossing line and the other is whether his (or her) moving direction

corresponds to the predefined entrance direction. Only a pedestrian that satisfies both conditions can be counted as valid one. The system needs to compare x-coordinate of the pedestrian with that of vertical crossing line. If they match, it means the pedestrian is crossing the entrance and we should focus on it. If not, it means that he (or she) is not involved with theline and we donot need to conduct the next procedures for checking the other conditions. Instead, the system conducts this procedure again to find other valid pedestrians.Once we know the detected pedestrian is passing the cross line, also we should know the other condition - whetherhis (or her)moving direction is right-to-left. For checking this condition, the system gets pedestrian's previous coordinate and compares it with his (or her) current one. Ifhis (or her) currentx-coordinate is smaller than the previous one, it means that he (or she) has ideal moving direction for this case and he (or she) can be counted as a valid pedestrian. The system repeats the procedures mentioned in this section as many times as the number of the regions detected as a pedestrian.If the whole procedures are performed normally, the number of valid pedestrians increases when more than one person crosses the entrance at the same time.The system needs to add this image's counting data of valid pedestrians to a total counting data and savesit.

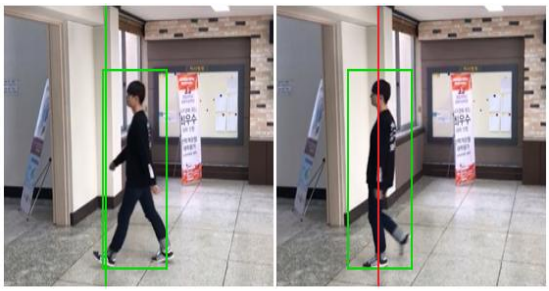


Fig. 4.Pedestrian passing crossing line

[Horizontal Path, Right Direction] In this section, we will handle the case that the pedestrian's path is horizontal, and the entering direction is left-to-right in the image. The overall procedures in determiningfeatures are similar with that of the first case. However, there are some important differences. The procedure is the same until checking line-crossing, which arecomparison between x-coordinate of the pedestrian and that of the crossing line, but the direction checking procedure is different. In this case, as the pedestrian should have left-to-right moving direction, only a pedestrian whose currentx-coordinate is bigger than his (or her) previous one is in condition. A pedestrian who meets two conditions can be counted as a valid pedestrian in this case. The system repeats this section's procedures as many times as the number of regions detected as a pedestrian.After that, the system adds counting data from repeated process in this image to atotal counting data and saves the total.

[Vertical Path, Upward Direction] The object moving away from the nearest point not only has a feature that the size of it gradually decreases in the image sequence but also has the direction to move upward from the bottom of the image. In the proposed system, as in the two cases mentioned before, we will find out the moving direction of the pedestrian, upward or downward,in the image and check whether his (or her) moving in desired direction.Having differences with mentioned two cases where the pedestrian's path is horizontal in the image, the crossing line is horizontal in the image and comparisons and calculations on y-coordinates are mainly used rather than x-coordinates in this case.First, the system checks whether the y-coordinate of the pedestrian and that of the horizontal line in the current image. If they are in the condition, the pedestrian is crossing the entrance and we should find out if he (or she) is moving upwards.If they are not, the system checks again to find the other pedestrians who are crossing the entrance. System compares the pedestrian's currenty-coordinateand his (or her)previous one so that we could know his (or her) moving direction. If he (or she) moves upwards, the current y-coordinate will be larger than the previous one.So only those cases are counted as valid one. Also the system repeats this section's procedures as many times as the number of regions detected as a pedestrian and when the iteration is over, the system adds the counting data of valid pedestrian to atotal and saves it.

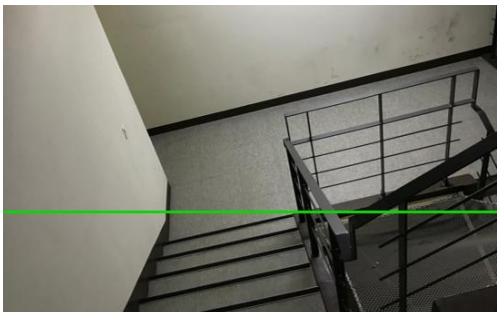


Fig. 5. Horizontal crossing line

[Vertical Path, Downward Direction] As the path of pedestrians is vertical as the same as third case, the procedure that checking whether the pedestrian is passing the line, which are comparison between x-coordinate of the pedestrian and that of the crossing line, is similar with the previous case. The Procedure of determining the pedestrian's direction of movement is performed only for a pedestrian who crosses the line, and an approaching pedestrian has a direction of moving from top to bottom in the image sequence. Therefore, when comparing the y-coordinate of the pedestrian of the current image with that of the previous image, only the pedestrian corresponding to the current one smaller than the previous one can be counted as a valid pedestrian.The system repeats the

procedures as many times as the number of regions detected as a pedestrian and adds the number of valid pedestrians in the image to a total data and saves it.

E. Data Transmission

If a predetermined process, which is one of the four cases, is repeatedly performed in each image from the image sequence, the total count would be correct when a pedestrian is entering to an entrance. This system can selectively transmit what it sends and how long the period of transmission for the various purposes. For example, suppose that we want to get the number of pedestrians who enter the entrance in each hour. When an hour has elapsed since the system was running, the total counting data will be transmitted to the server, and the time and total counting data in the system should be reset to zero to accumulate for new every in an hour.



Fig. 6. Raspberry Pi and a camera module



(a) Image of Raspberry Pi and server side (HOG-based)



(b) Image of Raspberry Pi and server side (Proposed)

Fig. 7. Simulation results comparison

IV. SIMULATION RESULT AND ANALYSIS

In this section, we provide the simulation results and analysis of the proposed system. The analysis is based on a comparison with a counting system using histogram of oriented gradients (HOG) features [18]. We used open computer vision library (OpenCV) for the processing of image [19]. The simulation was performed on Raspberry Pi Bwith a camera module as shown in Fig.6 and the environment of Raspberry Pi is 1.2GHz ARM Cortex-A53 under Linux, considering low costs and resource constraints.INRIA person dataset in [18] was used, which was trained with 2,716 human images and 1,218 non-human images.

Server is set up with a personal computer of 64-bits Windows 7 operating system, 3.2 GHz processor and 4GB memory where Raspberry Pi transmits the data to the server every minute. Database keeps current time, the number of pedestrian and feature's name as data. Fig. 7 shows the results of two systems used either HOG or Haar-like.

HOG-based system uses gradient orientation in each cell of an image and the process was performed in every cell.So, even if the system detects human well, the processing speed of it was actuallyvery slow.If the detection is slow, the system cannot provide real-time process.On the other hand, the proposed system uses the integral image and cascade classifier, which detects pedestrian in fast. This means that the proposed system can providereal-time processing. The simulation results show that the proposed system is fast in the processing and there are no frame-drops. Therefore, the proposed system is suitable for the real-time counting and ensures pleasant performance and lower cost as shown in Table I.

Table I. Comparison of related systems

Name	HOG-based counting system	Proposed counting system
Detection Method	HOG features	Haar-like features
Basic Feature of Detection	Oriented gradient	Brightness and contrast
Performance Speed	Slow	Fast

CONCLUSION

In this paper, we proposed a pedestrian counting system in real time image based on Haar-like feature and cascade classifier. In the proposed system, the pedestrian counting is achieved based on three phases, which are pedestrian detection phase, pedestrian counting phase and data transmission phase. The proposed system not only provides the counted number of pedestrians but also ensures better performance compared with the other systems. All processes in the proposed system are performed based on images only over Raspberry Pi with a camera module. Thereby, it is possible to implement a

high-performance pedestrian counting system at lower cost.

ACKNOWLEGDEMENTS

Hyunsung Kim is the corresponding author. This work was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2017R1D1A1B04032598).

REFERENCES

[1] N. Soni, N. Soni, "Benefits of pedestrianization and warrants to pedestrianize an area," Land Use Policy, vol. 57, pp. 139-150, 2016.

[2] W. K. M. Alhajyaseen, M. Iryo-Asano, "Studying critical pedestrian behavioral changes for the safety assessment at signalized crosswalks," Safety Science, vol. 91, pp. 351-360, 2017.

[3] A. A. Lantseva, S. V. Ivanov, "Assessment of pedestrian flow volumes through public transport modelling," Proceedia Computer Science, vol. 136, pp. 463-471, 2018.

[4] C. Sutheerakul, N. Kronprasert, M. Kaewmoracharoen, P. Pichayapan, "Application of Unmanned Aerial Vehicles to Pedestrian Traffic Monitoring and Management for Shopping Streets," Transportation Research Procedia, vol. 25, pp. 1717-1734, 2017.

[5] K. J. Clifton, P. A. Singleton, C. D. Muhs, R. J. Schneider, "Representing pedestrian activity in travel demand models: Framework and application," Journal of Transport Geography, vol. 52, pp. 111-122, 2016.

[6] X. Zhao, Z. He, S. Zhang, D. Liang, "Robust pedestrian detection in thermal infrared imagery using a shape distribution histogram feature and modified sparse representation classification," Pattern Recognition, vol. 18, no. 6, pp. 1947-1960, 2015.

[7] H. Sun, C. Wang, B. Wang, N. El-Sheimy, "Pyramid binary pattern features for real-time pedestrian detection from infrared videos," Neurocomputing, vol. 74, no. 5, pp. 797-804, 2011.

[8] C. Dai, Y. Zheng, X. Li, "Pedestrian detection and tracking in infrared imagery using shape and appearance," Computer Vision and Image Understanding, vol. 106, no. 2-3, pp. 288-299, 2007.

[9] X. Ren, T. Wang, S. Sun, M. Deng, K. Schweizer, "Speeded testing in the assessment of intelligence gives rise to a speed factor," Intelligence, vol. 66, pp. 64-71, 2018.

[10] K. Li, "Energy and time constrained task scheduling on multiprocessor computers with discrete speed levels," Journal of Parallel and Distributed Computing, vol. 95, pp. 15-28, 2016.

[11] W. Ahmed, M. M. Muthaher, J. M. Basheer, "Introducing high performance computing (HPC) concepts in nstitutions with an absence of HPC culture," Proc. of 2013 Sixth International Conference on Contemporary Computing, pp. 274-277, 2013.

[12] C. Feng, V. R. Kamat, H. Cai, "Camera marker networks for articulated machine pose estimation," Automation in Construction, vol. 96, pp. 148-160, 2018.

[13] A. Q. Susko, F. Gilbertson, D. J. Heuschele, K. Smith, P. Marchetto, "An automatable, field camera track system for phenotyping crop lodging and crop movement," HardwareX, vol. 5, e00029, 2018.

[14] S. D. Pizzo, S. Gaglione, A. Angrisano, G. Salvi, S. Troisi, "Reliable vessel attitude estimation by wide angle camera," Measurement, vol. 127, pp. 314-324, 2018.

[15] P. Viola, M. Jones, "Rapid object detection using a boosted cascade of simple features," Proc. of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 1, pp. 511-518, 2001.

[16] P. Viola, M. J. Jones, "Robust Real-Time Face Detection," International Journal of Computer Vision, vol. 57, no. 2, pp. 137-154, 2004.

[17] J. Matas, J. Sochman, "AdaBoost," Centre for Machine Perception, Czech Technical University, Prague.

[18] N. Dalal, B. Triggs, "Histograms of Oriented Gradients for Human Detection," in proc. IEEE international conference onComputer Vision and Pattern Recognition, vol. 1, pp. 886-893, 2005.

[19] OpenCV Library, https://opencv.org/

★★★