People Identification and Counting System Using Raspberry Pi

(AU-PiCC: raspberry Pi Customer Counter)

Tussanai Parthornratt[1] , Natchaphon Burapanonte[2], Wisarute Gunjarueg[3]
Assumption University,
Vincent Mary School of Engineering (VME),
Department of Telecommunications and Electronics,
Bangsaothong, Samutprakarn, Thailand
tussanaiprt@au.edu[1], natchaphon@masstron.com[2], gungun8799@gmail.com[3]

Abstract — **The work in this paper focuses on an implementation of OpenCV in an embedded system like raspberry Pi to create a mini-standalone station for counting people. The key feature of AU-PiCC (Assumption University's raspberry Pi Customer Counter) is to count a number of interested people on target product in a pre-defined area along with a simple face identification to avoid counting duplicates. The experimental results show that this raspberry Pi-based system can be used as a simple people counter station.**

Keywords—people counter, customer counter, human detection, face detection, people counting, identification, raspberry Pi, openCV, eigenfaces, image processing, computer vision, image processing

I. Introduction

The problem arises when there is a need to determine and quantify customer's interest. One of the most important factors to achieve high number of sales is to realize the level of customer's interest in the product so that a marketing scheme can be revised and adjusted to be well-suited for target customers. There are two main engineering aspects on this issue that needs to be described and they are customer's interest and people counting. An embedded system is the main target of this paper to allow portability and simplicity in terms of implementation and installation.

The paper is organized as follows. Related works are classified and reviewed in Section II. Section III explains system implementation and section IV presents experimental results, which will then be followed by conclusion in section V.

II. Related Works

Regarding research on analyzing people from an engineering viewpoint, two main categories of related works are customer's interest and people counting. The main objective on analyzing customer's interest is to be able to determine whether a customer has an interest in the product based on their behaviors in neighborhood of product location whereas the main objective on analyzing people counting is to be able to count how many people are residing in one particular region or area of interest. It is worth noting that the methodologies used in both of these two research areas are computer-vision based. For the work in this paper, it is the merging of these two categories. Namely, the use of people counting techniques is explored within field of view of a raspberry Pi to imply customer's interest in the product.

A. Customer's Interest

Customer's interest raises the issue of subjectivity and ambiguity in terms of description and quantification from business viewpoint. This will reduce the risk of product failure [1]. Even though a number of sales can be used to imply customer's interest, it will take place after a certain period of time that the products have been in store. It is quite pointless to determine customer's interest based on number of sales as customer's interest must be maximized to increase number of sales rather than implying from it. Namely, higher number of sales that imply higher level of customer's interest in the products might not be very useful as it is not instantly available. To understand customer's interest, a real-time analysis on customer's behaviors is, therefore, conducted using Finite state machine [2]. Different statuses of customer can be classified accordingly and they reflect only the presence of interest in the product not the quantity. Further research proposes the study on different levels of customer's interest implying different sets of customers' actions or shopping behaviors using Hidden Markov Model (HMM) and this can be more resourceful for marketing team [3].

Ultimately, the system that monitors the interest of customers and correlates it with number of sales to allow proactive marketing would ensure that its direction is appropriate. Such information is deemed complementary to the marketing strategy at point of sales and distribution channels. Further research is essential if high level of customer's interest does not correlate well with high number of sales as it indicates various aspects and possible problems of the products or marketing.

B. People counting

The methodology used for people counting is also based on computer vision. Different purposes to conduct people counting can differ vastly depending on the research problems being initially set. In terms of scale, the only variation is the scale of captured images or video footage being used for processing people counting algorithms, namely large-scale and small-scale people counting. Large-scale people counting serves a purpose of estimating the size of crowd and flow of people from afar within an area [4]. High density of people in an area creates an increasingly challenging situation to deal with including dynamic crowd motion [5] and background interference [6].

For small-scale people counting, it is mainly for counting a number of people in a group within a store or an area of interest using various technologies. These technologies vary from a sensor-based system to a vision-based system. According to S. Velipasalar, T. Ying-Li, and A. Hampapur, a small-scale people counting can be classified into three broad categories: Systems using contact-type counters; systems using sensors; and vision-based systems using cameras [7]. Obviously, sensor-based systems rely on various types of sensors (e.g. vision sensor [8]) that are particularly installed across area of interest. The drawback is an inability to differentiate people and their direction of motion. For vision-based systems, more features can be implemented in addition to the sensor-based systems. Namely, people can be tracked down [9], studied on their pattern of motion [10], studied on their interaction [7], etc. using a variety of machine learning techniques. Multiple cameras are also possible to extend field of view and introduce different viewing angles. This paper aims at counting potential customers who approach the aisle of a target product by usage of a single camera. An ability to count and classify if the face are of duplicates is a challenge for this project.

C. Small-scale People Counting Techniques

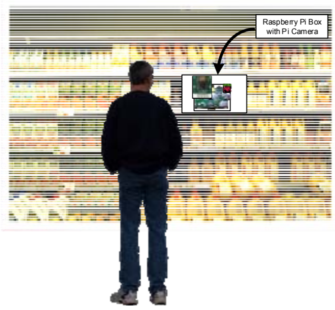
With reference to an OpenCV (Open Source Computer Vision), there are two available processes considered useful for people counting, namely, face detection and face recognition. The former is to determine if the captured images are classified as "face" then the latter will examine that face and determine whose face they belong to. The former mainly extracts characteristics of the images, derives the face descriptors, checks if it is the face. The resultant face can be used for training purposes in addition to the existing face databases like AT&T Facedatabase, Yale Facedatabase A, extended Yale Facedatabase B [11]. To perform a face detection operation, object in an image needs to be examined and classified as "face" by one of the two cascade classifiers; Haar-like features and Local Binary Pattern (LBP) features [12].

Once the face is detected, face recognition is important to avoid duplicates and face descriptors are the essence to enable efficient and reliable process. Face descriptors are an image representation obtained with high dimensionality that contain essential features to describe the face. Common problems arise when high-dimensional dataset needs to be reduced in complexity whilst keeping maximum information on an individual people. PCA (Principal Component Analysis) and LDA (Linear Discriminant Analysis) are two main methods to handle this high-dimensional dataset for eigenfaces and fisherfaces respectively. PCA simplifies the dataset by eliminating components based on covariance whereas LDA simplifies the dataset based on object classes. This effectively makes fisherfaces less prone to lighting conditions. However, due to the fact that these two methods are directly applied to a dataset, a number of same people faces with equivalent quality are required to ensure accuracy in face recognition. Having more images is ideal at best, therefore, LBPH (Local Binary Pattern Histogram) can help tackle this issue by defining only local features around each and every pixel of images to reduce high dimensionality. LBP is claimed as the most efficient real-time face detection [13].

The work in this paper is only dedicated to an implementation of a people counting system in an embedded platform like Raspberry Pi. In particular, OpenCV is the working horse in this project and eigenfaces method is used for this initial study.

III. SYSTEM IMPLEMENTATION

The system implemented is targeted at using raspberry Pi due to being low-cost and self-contained. An only accessory required for this project is Pi Camera, which is a 5M sensor resolution. The model used in this project is Raspberry Pi 1 Model B with 16GB Class 10 micro SD card, which is powered by a 700 MHz single-core ARM11 microprocessor and 512-MB RAM alongside 2 USB ports, a 15-pin PI camera (CSI) and HDMI connector.



a) Customer's point of view

b) Raspberry Pi's point of view

Fig. 1. System Overview

Raspberry Pi and Pi camera module are used with a software written to detect face of a person who comes and check the product then count total number of them. However, another problem has arisen, if the same person come check he product and then go back for second time the Pi would have counted for second time. We then proceed to the next level, which will be able to distinguish each people by using the so called "Face Recognition" process that is available in OpenCV so we can get the exact number of customer without repeatedly counted. Two main operations in this project consist of face analysis and customer counting as shown in Fig. 2.

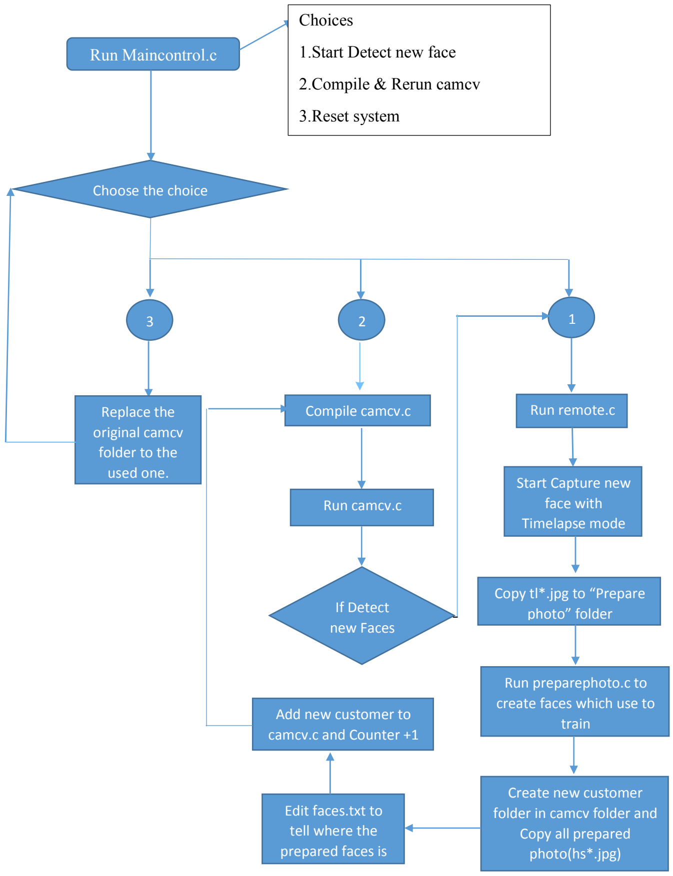


Fig. 2. Overall People Counting Program

A. *Face Analysis*

Once a Pi camera is properly installed and enabled, raspberry Pi will then be ready for running the C-program once all necessary tools have been compiled and installed (e.g. raspistill, raspivid, raspicam tools, etc.) along with Cmakelist.txt file modified to allow an original Linux-version OpenCV to run inside Raspberry Pi environment successfully. Next, OpenCV library named "libopencv-dev" and "python-opencv" need to be compiled and installed. Once that is done, a face recognition API called "libfacerec-0.04" needs to be compiled, installed and also added to your CMakeList.txt file. It is compulsory that a set of grey scale images of persons with equal size are available to maximize the result of face recognition (In this project, 100x100 pixels image is used to optimize recognition quality). The steps for "PreparePhoto" are can be found at https://thinkrpi.wordpress.com/opencv-and-pi-camera-board/ [14]. Once the photos have been prepared accordingly, they will then be used for training and it serves the purpose to detect face duplicates. Next, a file named "camcv.cpp" is written and this file includes a "raspivid.c" from official raspberry pi's camera source file to enable face recognition functionality. Level of confidence can be chosen too but very high value will increase processing time on Raspberry Pi. Additionally, faces used for training have to be defined and linked in the main program properly (e.g. camcv.cpp).

B. *Customer Counting*

Once the training is done, the process of customer counting can be started. Customer counting algorithm is implemented in such a way that a program will automatically capture customer's face and compare it with existing eigenfaces. This will trigger an action to either match that eignefaces with one in the customer's list (trained faces) or detect new customer as "other". If the match is hit, the number of customers will not be increased. But if the match is not found, new captured faces of this customer's will then be retrieved via "timelapse" function, stored in the new folder specifically created for this customer and added to the list of customers. Effectively, next time this person is detected by AU-PiCC, his/her ID will be shown as "Customer [number of customer]". To make this flow of execution possible, file input/output operation is key. Please refer to Fig. 2 for a summary of all operations.

IV. Experimental Results

In our experiment we conduct a test by storing one face, which considered as "staff" of the company so that when this person comes by, it will not be detected as a customer. That is, a person labeled as staff must not be regarded as a customer and the number of customer will not be increased. In contrast, an appearance of new customers within the field of view of Pi camera will trigger a program to initiate a facial detection and facial recognition, then followed by capturing new sets of faces upon mismatch of new faces with existing ones in the database by initiating *timelapse* mode of Pi camera. This new set of faces will be predicted as *other* rather than *staff* and will then be stored in the database. The program really shows that the counting with re-appearance of the staff named "Ice" and customer 1 do not increase a number of total customers. In our experiment, one staff named "Ice" is used as a pre-trained person and one more customer is set to emerge within field of view of our AU-PiCC.

After a program execution, the result shows that face detection is successful, staff's face is recognized and the person is labelled *Ice* as illustrated in Fig. 3. That is, a person named "Ice" is not regarded as a customer. Hence, the label "Customer [number of customer]" is not displayed on the screen as expected. This shows that AU-PiCC has the ability not to increase the number of customer. Now when the new person emerges into the scene, the result in Fig. 4. finally shows up with label "Customer 1" to signify the new customer.

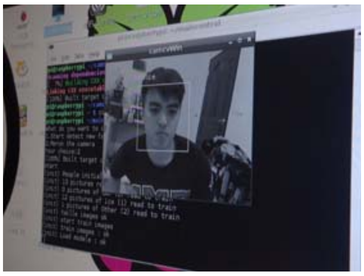


Fig. 3. Face Detection and Recognition of staff named *Ice*

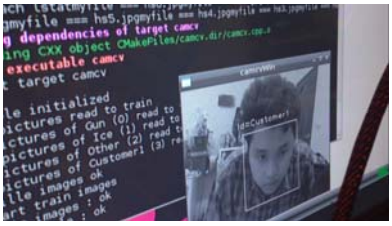


Fig. 4. Face Detection and Recognition of Customer 1

Regarding the issue of accuracy, this brings up the question of an ability of AU-PiCC to process on captured images, compute eigenfaces based on those images and compare them to its database so as to identify the person successfully. Table I show that the distance between one particular customer and raspberry Pi camera directly affects an accuracy of recognition as the distance increases. The data are based on simple experiments conducted for 3 sets of distances, namely 90 cm, 60 cm and 30 cm between a customer and raspberry Pi camera. In addition, processing time also decreases when the distance is deemed appropriate to allow AU-PiCC to capture enough fine-quality iamges containing the face of customer for facial recognition.

TABLE I. ACCURACY OF RECOGNITION AT DIFFERENT DISTANCES BETWEEN ONE CUSTOMER AND PI CAMERA

SET	*Distance*	*Captured Images*	*Accuracy of Recognition*	*Percentage of Accuracy**
1	90 cm		Poor	≤30%
2	60 cm		Medium	≤50%
3	30 cm		Best	≤90%

*Percentage of accuracy means how many times PiCC can identify a person correctly out of 5 experiments for each set.

V. CONCLUSION

The counter of AU-PiCC will start when the person who comes to check the product is neither staff nor the same old customer in the database. According to Table I, the minimum distance of 30 cm that will guarantee 90% accuracy becomes a limit to our system in terms of proximity of customer to AU-PiCC module that is well-suited for the task of counting customer because it can ensure genuine interest in the product instead of a normal passing-by occurrence. With such capability, any company will be able to observe the number of people interested in the product and use this information in order to develop the packaging or advertisement to make people interested in their product even more. Yet with the hardware that are not quite powerful, PICC still have some limitations which are shown in Table II.

TABLE II. AU-PICC LIMITATIONS

No.	*Issues*	*Descriptions*
1	Lighting condition	The lighting condition must be reasonable. (100 lux or above)
2	Headgear	Customers with hat or turban that cover all of their forehead may not be detected precisely.
3	Processing time	Delay in detecting/training new faces. In our experiment, we observe the delay of around 45 seconds.
4	Camera shutter speed	Pi Camera can only take one picture per second. This results in restrictions to have a person in front of the camera long enough to get sufficient amount of his/her photos (minimum # of photos is four photos assuming that they are of relatively fine conditions)
5	Number of customers	Due to not being very powerful, the more faces of the customers in the database slows down the counting process. It is found out that when there are more than five customers, AU-PiCC will almost come to a halt.

To overcome these limitations, further project can be built to forward image processing and data storing on eigenfaces to the cloud provided that connectivity is guaranteed uninterruptible. This will not only enhance AU-PiCC in terms of supporting a counting of larger number of customers but will also reduce processing time and show real-time counting with minimum delay.

REFERENCES

[1] E. M. Tauber, "Reduce New Product Failures: Measure Needs As Well As Purchase Interest," *Journal of Marketing,* vol. 37, pp. 61-64, 1973.

[2] R. Sicre and H. Nicolas, "Human Behaviour Analysis and Event Recognition at a Point of Sale," in *Image and Video Technology (PSIVT), 2010 Fourth Pacific-Rim Symposium on*, 2010, pp. 127-132.

[3] M. C. Popa, L. J. M. Rothkrantz, C. Shan, and P. Wiggers, "Assessment of customers' level of interest," in *Image Processing (ICIP), 2012 19th IEEE International Conference on*, 2012, pp. 41-44.

[4] A. B. Chan and N. Vasconcelos, "Counting People With Low-Level Features and Bayesian Regression," *Image Processing, IEEE Transactions on,* vol. 21, pp. 2160-2177, 2012.

[5] A. Albiol, V. Naranjo, and I. Mora, "Real-time high density people counter using morphological tools," in *Pattern Recognition, 2000. Proceedings. 15th International Conference on*, 2000, pp. 652-655 vol.4.

[6] H. Ya-Li and G. K. H. Pang, "People Counting and Human Detection in a Challenging Situation," *Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on,* vol. 41, pp. 24-33, 2011.

[7] S. Velipasalar, T. Ying-Li, and A. Hampapur, "Automatic Counting of Interacting People by using a Single Uncalibrated Camera," in *Multimedia and Expo, 2006 IEEE International Conference on*, 2006, pp. 1265-1268.

[8] L. Rizzon, N. Massari, M. Gottardi, and L. Gasparini, "A low-power people counting system based on a," in *Circuits and Systems, 2009. ISCAS 2009. IEEE International Symposium on*, 2009, pp. 786-786.

[9] J. Garcia, A. Gardel, I. Bravo, J. L. Lazaro, M. Martinez, and D. Rodriguez, "Directional People Counter Based on Head Tracking," *Industrial Electronics, IEEE Transactions on,* vol. 60, pp. 3991-4000, 2013.

[10] H. Jyun-Ciang, H. Mao-Cheng, H. Gwo-Jiun, and J. Gwo-Jia, "The Application of Face and People-Flow Recognition for Kanban Broadcast System," in *Innovative Computing, Information and Control, 2007. ICICIC '07. Second International Conference on*, 2007, pp. 146-146.

[11] (2015). *Face Recognition with OpenCV*. Available: http://docs.opencv.org/2.4/modules/contrib/doc/facerec/facerec_tutorial.html#face-recognition

[12] (2015). *Cascade Calssifier*. Available: http://docs.opencv.org/modules/objdetect/doc/cascade_classification.html?highlight=cascadeclassifier#cascadeclassifier

[13] K. Kadir, M. K. Kamaruddin, H. Nasir, S. I. Safie, and Z. A. K. Bakti, "A comparative study between LBP and Haar-like features for Face Detection using OpenCV," in *Engineering Technology and Technopreneuship (ICE2T), 2014 4th International Conference on*, 2014, pp. 335-339.

[14] P. Raufast. (2013). *OpenCV and Pi Camera Board*. Available: https://thinkrpi.wordpress.com/opencv-and-pi-camera-board/