2008 Congress on Image and Signal Processing

Object Detection and Counting System

Chomtip Pornpanomchai cccpp@mahidol.ac.th
Fuangchat Stheitsthienchai lin_eang@hotmail.com
Sorawat Rattanachuen u4688090@mahidol.ac.th

Department of Computer Science, Faculty of Science, Mahidol University
Rama 6 Road, Rajchatavee, Bangkok 10400, Thailand

Abstract.

The research intends to develop the object detection and counting system using image processing. Overall works are software development of a system that requires a video stream or single image. They consist of the following components: background without any moving objects and the scene with moving objects. The system is designed to find the differentiation which is the moving objects and find the number of moving objects from the video scene. The object detection and counting system consists of six major components: 1) Image Acquisition 2) Image Enhancement 3) Image Segmentation 4) Image Analysis 5) Object Counting, and 6) Reports.

The experiment has been conducted in order to access the following qualities: 1) Usability, to prove that the system can determine object detection and counting under the specific condition lay out. 2) Efficiency, to show that the system can work with high accuracy.

Key-Words: *object detection, object counting, image processing*

1. Introduction

The idea of using the video camera to count the number of objects has been proposed as a new way of detection and counting approach. The current detection and counting approaches are based too much on the sensor equipment. The use of sensor equipment to count the number of objects has been widely spread into different kinds of industries. Even though the equipment itself provides a benefit to the user, there are also some disadvantages. The sensor equipment is only used for real time - the user cannot insert other kinds of input such as video file to the sensor equipment in order to count the number of objects.

The sensor equipment also has a high cost and it is not a multitasking tool [1]. Compared to the video camera device, many users have now already installed video camera devices in their offices or buildings for security. We can use these video cameras for multitasking - not only for security, but also as the sensor equipment. That means we can count the number of objects or people in the detection area. The users do not have to install much equipment since we can combine the tasks together. So, the users can reduce the cost of investment. Image processing technology can fulfill this requirement.

Image processing is the technique which is based on the software component that does not require special hardware. With a typical video recording device and a normal computer, we can create the detection and counting device. By using the basic pattern matching theory, we can find the number of objects in the video scene from the known size of the object [2-6].

Few image processing key methodologies have been applied to this project. The image differentiation is used in the object detection process, level of difference for the image analysis and pattern matching to find the number of objects [7-9]. The project is still in the prototype mode. We require more research and development in order to reduce the system limitation and enhance the efficiency of the system to be able to perform for a real-world application. The topic of further enhancement will be discussed at the end of the document.

This project is about developing an image processing system that can detect the number of people from the video scene, which is running on specific constraints. The system is intended to replace the sensor equipment, which has some weak points as explained before. The system will be created in the form of software which consists of input and output parts. The input part is accepts 2 formats of data that are image file and video stream from the video camera. The output part is responsible for reporting the result to the user. The system can report in 2 ways. The first way is on the program interface and the second is on the website for the remote user.

2. Related Work

Due to the traffic jam problem in Bangkok, the idea of applying the technology to solving the

978-0-7695-3119-9/08 $25.00 © 2008 IEEE
DOI 10.1109/CISP.2008.108

problem has been launched. Genius Traffic System is used to represent the state of traffic in real time for the traveler. The system will present the state of the main streets in Bangkok in order to resolve the traffic problem [10]. The traveler can make the decision to choose a better way by trying to avoid streets that have traffic jams shown in Figure 1.

Figure 1 The genius traffic system in the real life

The genius traffic board will be installed near main cross-roads that usually have traffic jams. The genius traffic board represents the traffic state in a graphic view which shows street structure. It uses three colors to present the traffic states, which are green, yellow and red, as shown in Figure 2.

Figure 2 The genius traffic board

2.1 How the Genius Traffic System Works

First, the detector cameras or closed circuit televisions (CCTVs), which are installed at each point will detect the traffic movement. Then the data will be sent through fiberglass cable to the computing centre.

At the computing centre, the system will determine the traffic status by using the image processing technique. It uses occupancy ratio to consider the traffic state.

With the occupancy ratio algorithm, the computer system will generate a square area whose size is close to the size of a car on the street. The detector camera will detect the total time for each car to run through a square. If the total time is quite little, it means that a car can run with speed and so there is no traffic jam. If the total time is very much, it means that a car can only run slowly and so there is a traffic jam occurring. The computer system will measure the total time that the car has occupied the detector over the time unit. For example, the time unit is 100 seconds and a car has occupied the square area for 100 seconds. This means that there is a traffic jam. The occupancy ratio is between 0.0 and 1.0 and the related color is shown in Table 1.

Table 1 The occupancy ratio of genius traffic system

Occupancy Ratio	Color
0-0.3	Green
0.3-0.8	Yellow
0.8-1.0	Red

Finally, the result will be sent from the computing centre back to the genius traffic board. The genius traffic board will show the traffic state using the color result. The traffic state is almost in real time since it takes some time to compute and transfer the result.

3. System Design

This part introduces our approach of creating a system of object detection and counting from the video scene. We will start with the overall framework of the system and the description of each component in the framework and the basic understanding of the technique that we are using in each component.

3.1 Overview of Object Detection and Counting Framework

The hardware requirement for the object and detection and counting system is as follows: The system consists of a normal PC which is connected to a video camera, as shown in Figure 3. An input of the system must be a background scene without any moving objects and a background scene with moving objects. The height from the floor to the video camera must be known. The basic idea of the system is to calculate the number of objects that move through the detection area from the difference between the background frame and other captured frames. The simulated scene with moving objects from top view is shown in Figure 4.

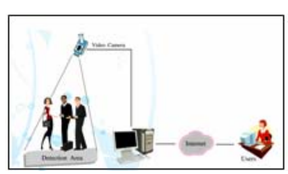

Figure 3 The required system hardware for gathering the input objects

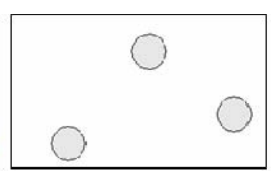

Figure 4 The simulated scene with moving objects from top view

3.2 Object Detection and Counting Model

From the framework which is shown above, we analysis and design the structure chart of the object detection and counting system as shown in Figure 5. The structure chart consists of six major components, which are 1) Image Acquisition, 2) Image Enhancement, 3) Image Segmentation, 4) Image Analysis, 5) Object Counting, and 6) Report. The details of each part will be described as follows:

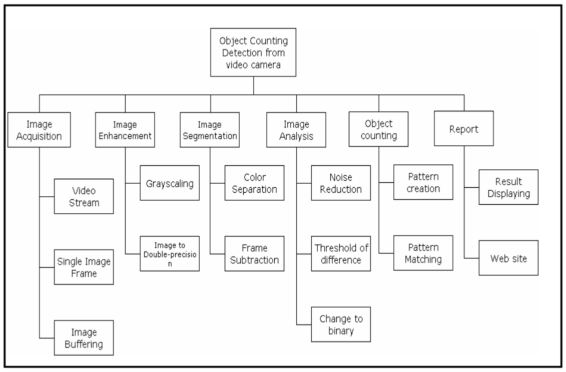

Figure 5 The object detection and counting system structure chart

3.2.1 Image Acquisition

We get the input from a video camera which is video stream. It is very difficult to perform the detection and counting process from the streaming level because all required abstractions are in the single frame level. So, we gather the image in frame level.

3.2.2 Image Enhancement

We have to set the level of difference between two images, the background and the image with moving objects. The level must be settled appropriately because it is used to decide whether the value in each pixel between two images is different enough or not. If the values of difference are greater than the level of difference, we assume that there are some objects in the scene. The level of difference is very important. Sometimes the values in the same pixel between two images are slightly different because of light energy, not the moving objects.

3.2.3 Image Segmentation

We have to calculate the approximate size of the objects that we have to detect using the height of the video camera and the size of objects that we have gathered before.

3.2.4 Image Analysis

This is the operation in charge of finding the number of objects that are moving in the detection area. By using a pattern matching technique, we will get the number of objects in the frame.

3.2.5 Object Counting

This process is to count the number of objects in a scene. The number of objects that are detected will be collected in the excel file to be analyzed later.

3.2.6 Reporting

The number of objects will be reported to the user in a proper way and is easy to understand on the website. The website is serviced for the people who benefit from the system, such as a shop's manager, to see whether there are a lot of customers in the shop or not. The manager can remotely detect them even if he/she cannot go to the shop.

4. System Implementation

This section illustrates the deeper detail of the implementation of the Object Detection and Counting system. We will give more explanation of each component and show the resultant image [11, 12].

First component, image acquisition - the processes in this component are "video stream" which is an input from the video camera, "single image frame" which translates the video stream into single images, and "image buffering" which will store the images in the buffer. The result of this process is shown in Figure 6.

Figure 6 Captured Image

Figure 7 Grayscale Image

Figure 8 The Image after subtraction

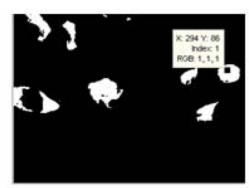

Figure 9 Binary Image

Second component, image enhancement - the processes in this component are "gray scaling" which will translate the RGB image into a grayscale image, as shown in Figure 7.

Third component, image segmentation – the processes in this component are "color separation" which will separate three dimensional arrays of RGB into single arrays, and "frame subtraction" which will subtract the background image with the captured image to find the difference or movement. The resultant image of this process is shown in Figure 8.

Fourth component, image analysis - the processes in this component are “noise reduction” which is used to eliminate the noises and improve the quality of the image, “threshold of difference” which the user has to insert, and “change to binary” which will use the threshold of difference to compare with the values and change the values to binary. The result of this process is shown in Figure 9.

Figure 10 Image after pattern matching

Figure 11 Reporting on the first interface

Fifth component, object counting - the processes in this component are “pattern creation” where the user has to insert the size, and the system will generate the pattern according to the size and “pattern matching” which will match the pattern with the group of values in each array. The image after pattern matching is shown in Figure 10.

Last component, report - the processes in this component are “result displaying” which will display the number of people that the system can detect on the system interface, and “website” which will display the output on the website for remote users. The report screen is shown in Figure 11.

5. Experimental Results

This part presents an experimentation on the object detection system, which is developed and based on the concepts and design mentioned in the previous part. In this system, the experiment’s results are focused on the usability and the effectiveness of the system.

5.1 Usability Proof

We connected the video recording device to the machine and allowed the objects to move through the detection area from any direction. The first image is the background image which is used to subtract with other images to find the people.

According to the captured screens showing in Figure 12, it displays the blank area of images with empty size of object for the user to insert. After inserting the size of object and clicking the start button, the system will start to capture the image and find the number of objects in the captured image as shown in Figure 13.

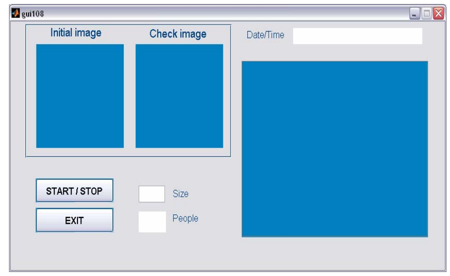

Figure 12 A program display screen

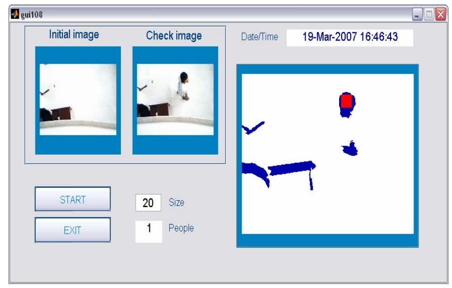

Figure 13 A screen showing the background image, captured image, how the system detects the number of people, time and the result of the current image

5.2 Effectiveness Proof

It is necessary to have some data as our input for the system in order to perform the experiment. We decide to get the images from different heights and different numbers of people moving around the detection area. The testing scenario consists of the following components.

- The location is in a building.
- The color of the background’s floor is nearly white.
- The recording video camera is installed above the detection area.
- The color of people’s shirts is white and the color of bottoms is black.
- The people moving past the detection area come from every direction.

Before we start the object detection and counting system, we have to find the estimated size of object, which is the estimated size of people in this experiment. The size is calculated from the fraction that we found from the experiment, as shown in Figure 14.

After we got the result from the object counting system with varying levels of difference and the actual number of people in each image, we compared them and found the effectiveness of the system. We calculated the percentage of correctness in two ways. The first was from the number of images that had the same result as the actual image. The second was from the total number of people that the system found, compared to the actual total number of people in the data set.

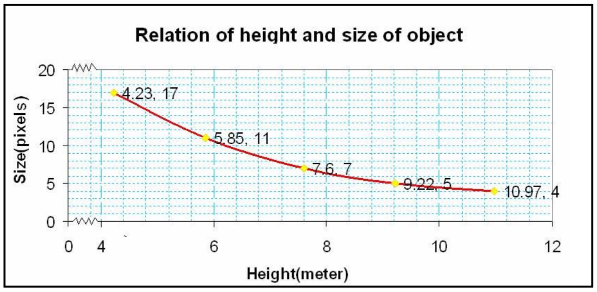

Figure 14 Graph showing the relationship between height and size of an object

For the testing data set, the total number of images is 26, where the first image is used as the background. The actual total number of people in this data set is 156. The experimental results are shown in Table 1 and 2.

Table 1 The percentage of correctness from the number of correct images in the testing data set

Level	**Total number of people**	**Percentage of correctness**
0.2	247	42
0.25	233	51
0.3	197	74
0.35	177	87
0.4	159	98
0.45	140	90

Table 2 The percentage of correctness from the total number of people in a testing data set

Level	**Number of correct images (out of 25 images)**	**Percentage of correctness**
0.2	6	24
0.25	9	36
0.3	10	40
0.35	16	64
0.4	17	68
0.45	11	44

From the experiment, the best level of difference that we can get is between 0.4 through 0.45. We should use this level of difference with the system. The height also has an effect on the system. We can see that the height should not be above 5 meters since higher positioning will cause a worse result.

6. References

[1] Dariu M.G. *"Sensor-Based Pedestrian Protection"*, IEEE transaction on Intelligent Transportation Systems, November/December 2001

[2] Kaweepap Kongkittisan, *"Object Speed Detection from a Video Scene"*, Mahidol University, Bangkok Thailand, May 2003.

[3] Liang Z. Charlea E.T. *"Stereo- and Neural Network-Based Pedestrian Detection"*, IEEE Transactions on Intelligent transaction system, Vol. 1, No 3, September 2000.

[4] Osama M. and Nikolaos P.P. *"A novel Method for Tracking and Counting Pedestrians in Real-Time Using a Single Camera"*, IEEE Transactions on Vehicular Technology, Vol. 50, No. 5, September 2001.

[5] Fengliang X., Xia L. and Kikuo F., *"Pedestrian Detection and Tracking with Night Vision"* , IEEE Transactions on Intelligent Transaction System, Vol. 6, No.1, March 2005.

[6] Massimo B., Alberto B., Alessandra F., Thorsten G, and Marc-Michael M., *"Pedestrian Detection for Driver Assistance Using Multiresolution Infrared Vision"*, IEEE Transaction on Vehicular Technology, Vol. 53, No 6, November 2004.

[7] Dariu M.G., *"A Bayesian, Exemplar-Based Approach to Hierarchical Shape Matching"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol.29, No.8, August 2007.

[8] Hideo M. N.M.Charkari and Takeshi M., *"On-Line Vehicle and Pedestrian Detections Based on Sing Pattern"*, IEEE Transactions on Industrial Electronics, Vol. 41, No. 4, August 1994.

[9] Chia-Jung P., Hsiao-Rong T., Yu-ming L., Hong-Yuan M.L. and Sei-Wang C., *"Pedestrian detection and tracking at crossroads"*, Pattern Recognition, Vol.37, Issue 5, May 2004.

[10] Genius Joint Venture Co.,Ltd., *"Genius Traffic System"*, http://www.forth-its.com,creation date: unknown, last modified date: June 19, 2006, accessed date:November 28, 2006.

[11] The MathWorks, Inc., *"An open exchange for the MATLAB and Simulink user community"*, http://www.mathworks.com/matlabcentral, creation date: unknown, last modified date: unknown, accessed date: November 30, 2006.

[12] Jeff Anderson, *"Creating a Dynamic Reports using ASP and Excel"*, http://www.codetoad.com/asp_excel.asp, creation date: September 1, 2003, accessed date: February 20, 2006.