IAENG International Journal of Computer Science, 32:4, IJCS_32_4_15

Image Processing Application in Toll Collection

Yu-fai Fung, Homan Lee, and M. Fikret Ercan

Abstract— **The toll rate charged for the usage of facilities such as a tunnel or a bridge is usually proportional to the number of axles possessed by a vehicle. However, it is sometimes difficult to determine the number of axles of a vehicle by the toll-booth operator and therefore, an automatic system that can identify the number of axles is sought. Instead of detecting the axle, wheels of a vehicle are tested and a method based on the Hough transform for detecting circles is proposed. As the system must be able to detect the correct number of wheels in real-time, sub-sampling based on the Haar Wavelet transform is applied. Experimental results show that the system is able to identify the wheel correctly and to process the input images in real-time.**

Index Terms—**Image Processing, toll collection, Haar Wavelet, Hough Transform.**

I. Introduction

The toll rate charged for a vehicle in using a tunnel or a bridge is usually proportional to number of axles of a car. For example, for the Eastern Cross Harbour Tunnel of Hong Kong, each additional axle is charged for $3 USD. Currently, the toll is collected by an operator, who is responsible to determine the axle number when the vehicle arrives at the toll booth. However, the booth operator has a limited angle of view and sometimes the operator may be blocked by the vehicle as well. It is sometimes difficult to determine the correct number of axles. In most cases, operators determine the axle number based on their experiences. It is, therefore, desirable to have an automatic system to either assist the booth operator to count the axle well in advance or to verify that the toll is correctly charged.

Traditional method to detect axles of a vehicle is by laying sensor, or treadle, in the lane leading to the toll booth. When a vehicle passes through the sensor, a signal will be generated and by processing the signal, the number of axle can be determined. However, the major concern of such a setup is maintenance. When a sensor, or treadle, is broken then the lane must be closed down. It is a time consuming and costly process, if the sensor laid underneath the road surface needs to be replaced, or repaired. This is highly undesirable if the traffic-flow of the facility is high. Therefore, an alternative solution is sought and image processing is considered because, the setup is simple and easy to maintain.

In the next section, we will introduce the setup of the system and it is followed by a discussion in the software algorithm proposed. In Section III, experimental results will be presented and conclusion of the paper is given in Section IV.

II. The Image Processing System for Wheel Recognition

Initially, the system will be used as a tool to validate the fare charged. The setup cost is a major design criterion to be considered and therefore, components of the system are low-cost commodity products. The basic setup includes a low-end digital video camera and a computer. The resolution of the image produced by the camera is only 320x240 pixels. Since it is difficult to examine axles of a vehicle and therefore, the system will detect wheels instead of axles. The camera is located so that wheels of a vehicle can be captured when it passes through the toll booth. Figure 1 shows the position of the camera relative to the road and a sample image captured by the camera is given in Figure 2. From observations, it takes about two seconds for the front wheel and the rear wheel of a car to pass through the camera, implying that the processing system must complete the recognition process within the two-second time frame.

Due to the position of the camera, the image of the wheel is slightly distorted; however, it still resembles to a circle. Similar to [1], we applied Hough transform for circle [2] in order to detect the presence of the wheel. The major advantage of Hough transform is its capability of identifying a wheel even when it is only partially appearing in an image.

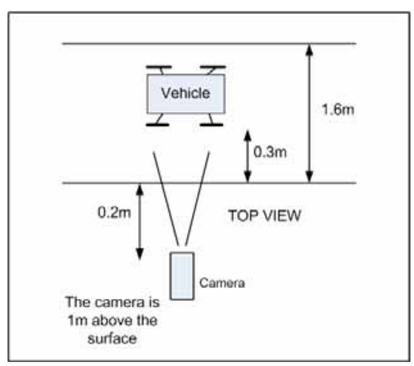

Figure 1 Camera set up for the wheel detection system

A. *Hough Transform for circles*

Hough transform (HT) for detecting circles is based on the same principle introduced by the Hough transform [3] for line

Yu-fai Fung is with the Department of Electrical Engineering, The Hong Kong Polytechnic University, Hong Kong. (phone: 852-2766-6171; fax: 852-2330-1544; e-mail: eeyffung@ polyu.edu.hk).

Homan Lee, is studying in Cornell University, New York, USA. (e-mail: hl334@cornell.edu).

M. Fikret Ercan is with the School of Electrical and Electronic Engineering, Singapore Polytechnic, Singapore. (e-mail: mfercan@sp.edu.sg).

(Advance online publication: 12 November 2006)

segments. The Hough transform is a well known technique for detecting parametric curves (such as circles) in images. The basic operations of Hough transform is to map feature points in the image space to the parameter space, or the Hough space, where the parameters are used to describe the curve to be detected.

A feature point extracted from an image is used to define circles in the parameter space with different radii, and therefore, feature points belong to the same circle in the image space will produce circles intersecting at the same point in the Hough space. A peak detection method is used to locate local peaks in the Hough space. The location of each peak gives the parameters of each detected circle.

As a circle is governed by three parameters as shown in the following equations (1 and 2), the centre coordinates (a, b) and the radius (r), the Hough space is a three dimensional space, with the z-axis representing the radius.

$$x = a + r\cos\theta \qquad (1)$$

$$y = b + r\sin\theta \qquad (2)$$

In order to perform an exhaustive Hough transform, we should consider all possible values for the radius. However, in our application, we are searching for wheels of a vehicle; hence we can limit our search space by defining a suitable range for the radius of the wheel. The range for parameter r is set as 60 to 200 pixels.

Figure 2 Sample image capture by the system

B. Image Proccessing

Hough transform is based on feature points extracted from the original image and usually, edges are used as the feature points. Result of applying Sobel filter to one of the captured image is shown in Figure 3. As shown in Figure 3, the wheel can produce sharp edges, which could produce a peak in the Hough space, as we will discuss in the next section. However, other areas in the image, as shown in Figure 3, also produce a significant number of edge points, which are due to the texture of the road.

If we apply the Sobel filter to an image when there is no presence of a vehicle, the result, depicted in Figure 4, is a huge number of edge points produced. The edge points come from the texture of the road can be regarded as noise, which will induce a huge overhead in the execution time of the Hough transform and most importantly will produce measurement errors so technique(s) to reduce the unwanted edge points is sought.

The Canny edge detector [4] is a very powerful tool for detecting edges in a noisy environment and is considered in our application. The Canny edge detector first smoothes the image to eliminate noise and then finds the image gradient to highlight regions with high spatial derivatives. The algorithm then tracks along these regions and suppresses any pixel that is not at the maximum (nonmaximal suppression). The gradient array is now further reduced by using two thresholds (T1, T2), with T1<T2. The process is to track along the remaining pixels that have not been suppressed. If the magnitude of an edge pixel is below the first threshold (T1), it is set to zero (made a nonedge). If the magnitude is above the high threshold (T2), it is made an edge. If the magnitude is between the 2 thresholds, then it is set to zero unless there is a path from this pixel to a pixel with a gradient above T2. The result of applying Canny edge detector to Figure 2 is given in Figure 5, comparing Figure 3 and Figure 5, the Canny edge detector can remove most of the unwanted edge points. But on the other hand, due to the complexity of the Canny detector, it takes a long time to finish and therefore, based on our studies, we cannot complete the process within the two-second time frame.

In order to shorten the processing time for the edge detection algorithm, we simplify the process by retaining only the first two steps. A Gaussian filter is first applied and it is followed by the Sobel filter. The processed result based on these two steps for Figure 2 is given in Figure 6. Comparing to Figure 3, it is obvious that the unwanted edge points are reduced and the result can now be used in the Hough transform.

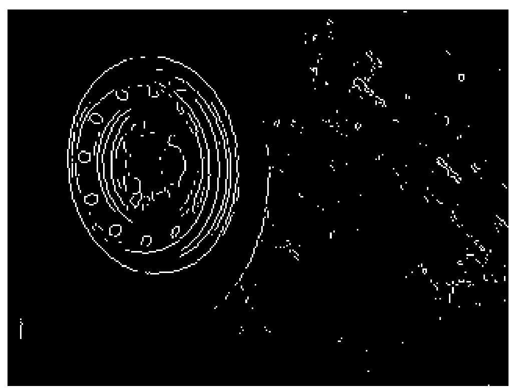

Figure 3 Result of Sobel Filtering

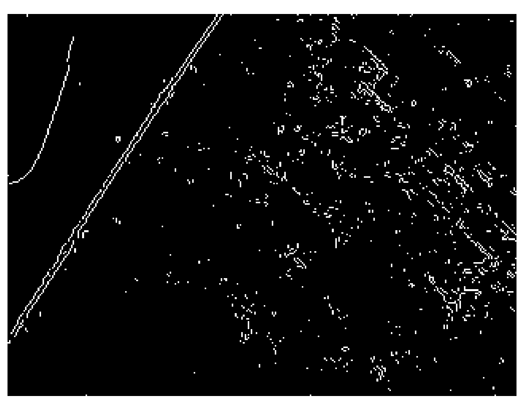

Figure 4 Result of Sobel Filtering to background image

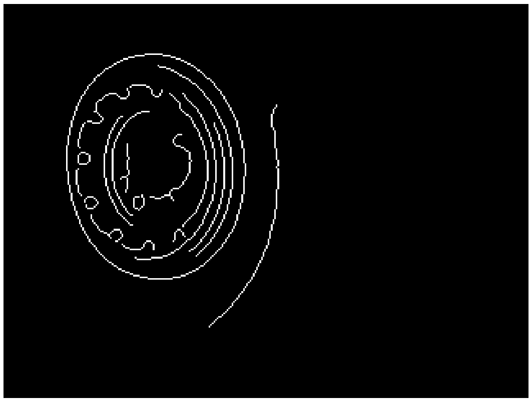

Figure 5 Result of Canny edge detection

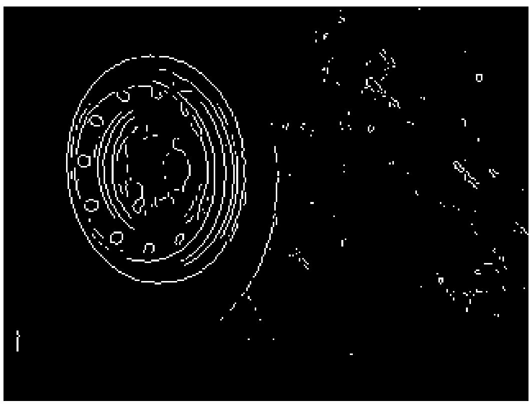

Figure 6 Result of Gaussian and Sobel Filtering

C. Image Subsampling

With Gaussian smoothing function and Sobel filtering, the image is processed by the Hough transform. Figures 7 illustrates the results obtained. A circle is overlaid on the original image if a circle is identified. As we can see that the results are satisfactory and the wheel can be identified in all the images. However, it takes 38 seconds to process 24 images which are captured when the car moves in front of the camera, and it cannot meet the timing requirement imposed on the application, therefore, technique to reduce the total processing time is necessary.

The most straight forward approach to reduce the processing requirements is by using a smaller image. One way to reduce the image size is by using a sub-window approach, i.e. only part of the image covered by the sub-window will be processed. The sub-window approach is easy to implement but it may not be easy to locate the best sub-window, both in terms of its size and position. As the car is moving therefore, the position of the wheel varies in different frames of image. In addition, cars will not follow the exact path when crossing the toll booth therefore, the wheel position changes. A large sub-window will certainly cover all possible locations of the wheel but it then will not reduce the processing requirements significantly.

Sub-sampling is adopted and the technique is based on the Haar Wavelet [5]. By sub-sampling, the size of the image, becomes only 160x120, is reduced to only a quarter of the original but most importantly, the sub-sampled image includes all the information required for performing the recognition algorithm as tested with the original image.

The number of operations performed in the Hough transform is governed by the product of two parameters (R, N), where R is the range of radius of the wheel being tested and N is the number of edge pixels. After sub-sampling, the wheel image will be reduced by half and therefore, the new possible range for the radius becomes R/2. Similarly, as the image is sub-sampled to only a quarter of the original then the number of edge pixels will be halved as well. Therefore, the number of operations for executing the Hough transform is in the order of (R/2 * N/2) for the sub-sampled image.

The Haar Wavelet is chosen because it is easy to implement and therefore, will limit the computing overhead induced by the additional process. The sub-image is the result of down sampling in columns followed by down sampling in rows, as illustrated below.

Downsampling in columns

$$f_{dc}(x,y) = \frac{f(x,2y) + f(x,2y+1)}{2} \qquad (3)$$

$$\textit{where } y = 0 \textit{ to } \frac{N-1}{2}; \; N \textit{ is the width of the imge}$$

$f_{dc}(x,y)$ *is the downsampled image in columns*

Downsampling in rows

$$f_{dr}(x,y) = \frac{f_{dc}(2x,y) + f_{dc}(2x+1,y)}{2} \qquad (4)$$

$$\textit{where } x = 0 \textit{ to } \frac{M-1}{2}; \; M \textit{ is the height of the image}$$

$f_{dr}(x,y)$ *is the downsampled image in rows*

Due to the properties of the Haar wavelet, the sub-sampled image is created via two lowpass filters and therefore the texture of the road in the sub-sampled image is blurred. The edge image of the sub-sampled image, as depicted in Figure 8, is almost identical to the result obtained after applying the Canny edge detector to the original image, as shown in Figure 5. Therefore, after sub-sampling, we can by-pass the Gaussian filter and apply the Sobel edge detection directly so the computing overhead induced by the Haar wavelet transform can be minimized.

III. RESULTS

The processing time for 24 frames of sub-sampled image takes about 1.5s and therefore, it meets the timing requirements, which is about two seconds. By analyzing the peak value obtained from the Hough space in each image obtained from a movie clip capturing a vehicle passing in front of the camera, Figure 9, a local maximum is observed when a wheel is approaching. The front wheel of the car enters the view of the camera from frame 3 to frame 7 and a local maximum emerges at frame 5. Similarly, the rear wheel appears in images from frames 15 to 17, since the car is accelerating when the car leaves the toll booth area, therefore the total number of image frames covering the rear wheel is less than the front wheel. However, a local maximum is still available in frame 16. Based on such a phenomenon, we can easily determine the number of wheels captured by the camera by detecting the number of peak from the Hough transform plot similar to Figure 9. Such an approach is more robust because the wheel is detected according to a sequence of images rather than based on a single image.

In order to substantiate such an approach, peak values in the Hough space of three different kinds of images were examined. Images are divided into three categories including those consist of a complete wheel, with a partial wheel and without a wheel. The images cover different kinds of vehicles including mini-van, lorry and a coach. The peak Hough space values for 25 frames from each category were plotted and shown in Figure 10. As we can see, the peak Hough space values for images with a complete wheel and a partial wheel are always higher than those images without a wheel. The average peak Hough space values for the three cases are given in Table 1. As we can observe, there are significant differences between the average peak values for the three different kinds of images, therefore, we can expect that a local peak in the Hough space will present when a wheel is passing through the camera, similar to results depicted in Figure 9. By employing a proper thresholding method and by analyzing the distribution of the peak values extracted from the Hough space, it is very easy to determine the number of wheels passing through the camera.

IV. SUMMARY

In this paper, we described an automatic system to count the number of axle of a vehicle in real-time for toll collection purposes. The Hough transform for circle is used for detecting the presence of a wheel. Our experiments show that the Hough transform is suitable for such an application. Currently, we can process up to 24 images within 1.5s and it satisfies the timing constraint imposed upon the system. Our system setup is simple and by using commodity components, its setup cost is also low.

Currently, we can process 24 images in 1.5 seconds, however, different vehicles passing through the toll booth may be in different speeds and therefore, it may be necessary to increase the number of images being processed. As the computing complexity of the software system is governed by the image size as well as the complexity of the Hough transform. Reducing the image size by sub-sampling is an effective approach to speedup the process. In Section II (C), we have shown that by reducing the image to one-quarter of the original, the number of operations performed will be reduced by a quarter as well. Therefore, if we apply Haar transform twice then the number of operations can be reduced to only 1/16.

In order to reduce the operation of the Hough transform, the Random Hough transform [6] (RHT) can be considered. The standard Hough transform maps each pixel in the image to a curve in the parameter space, RHT maps an n-tuple of pixels in the image to a single point in the parameter space. In case of circle detection, triplets of pixels will be randomly chosen from the image and mapped to a single point in the 3-D parameter space. As the points selected from the image is not exhaustive, therefore, the number of operations can be reduced. There are reports [7], [8] on the improvement in efficiency based on the RHT and therefore, we should consider in the future. In addition, the application of parallel processing techniques such as SSE (Streaming SIMD Extensions) [9], which is embedded in the latest Intel microprocessors, can also reduce the processing time and this is another direction that we will investigate.

ACKNOWLEDGMENT

This work is supported by the Department of Electrical Engineering of The Hong Kong Polytechnic University. We would like to thank engineers of KML Engineering Ltd for many useful discussions and for the supply of data used in the experiments.

REFERENCES

[1] Frenze J.F.: A Video-based Method for the Detection of Truck Axles, NIATT Report Number N02-05 (2002)

[2] Davies E.R.: Machine Vision: Theory, Algorithms, Practicalities. Academic Press, London, (1990)

[3] Hough P.V.C.: Method and Means for Recognizing Complex Patterns. U.S. Patent 3,069,654, (1962)

[4] Gonzalez R.C., Woods R.E., and Eddins S.L.: Digital Image Processing Using MatLab. Prentice-Hall (2004) 378-425

[5] Nievergelt Y.: Wavelets Made Easy. Birkhäuser, Boston (1999)

[6] Xu L., Oja E. and Kultanen P.: A New Curve Detection Method, Randomized Hough Transform (RHT). Pattern Recognition Letters. 11 (1990) 331-338

[7] Walsh D. and Raftery A. E.: Accurate and Efficient Curve etection in Images: The Importance Sampling Hough Transform. Pattern Recognition. 35(2002)1421-1431

[8] McLaughlin R.A.: Randomized Hough Transform: Better Ellipse Detection. Proceedings of IEEE TENCON. (1996)409-414

[9] Fung Y.F., Ercan M.F., Cheung W.L., Singh M.G, "Avenues for High Performance Computation on a PC", Lecture Notes in Computer Science, Vol. 3044 (2004) pp.246-251.

Table 1 Average peak Hough space value for different image types

Image Type	Average Peak Hough Space Value
Whole wheel	25.04
Partial	18.32
No wheel	11.56

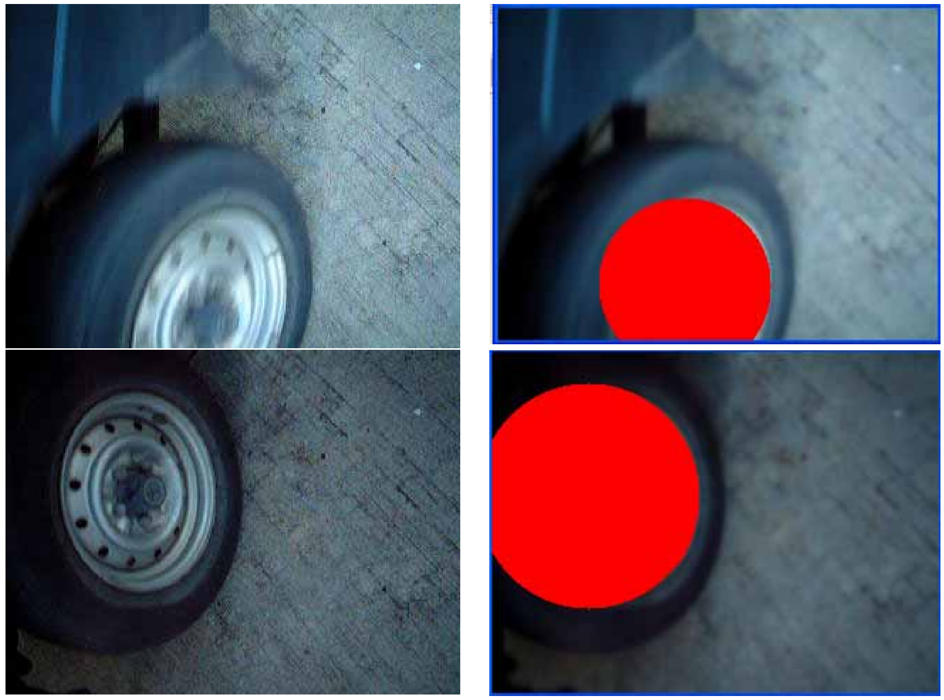

Figure 7 Results of wheel recognition algorithm

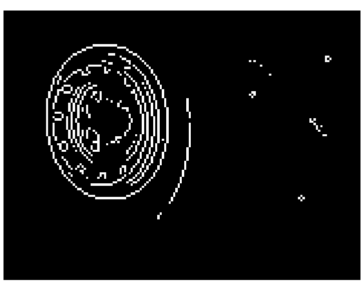

Figure 8 Edges obtained by Sobel filter to sub-sampled image

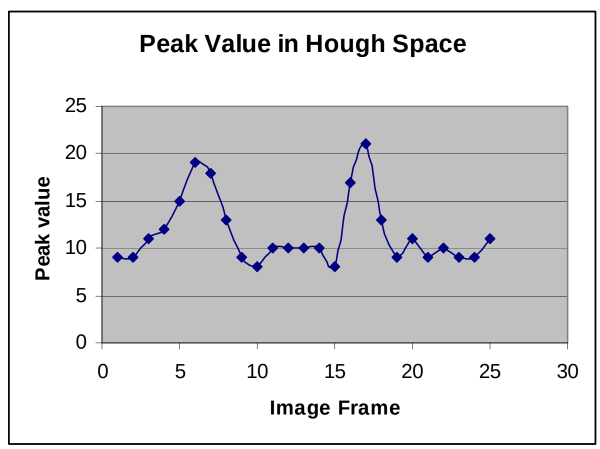

Figure 9 Peak Hough Space Value for Movie Clip

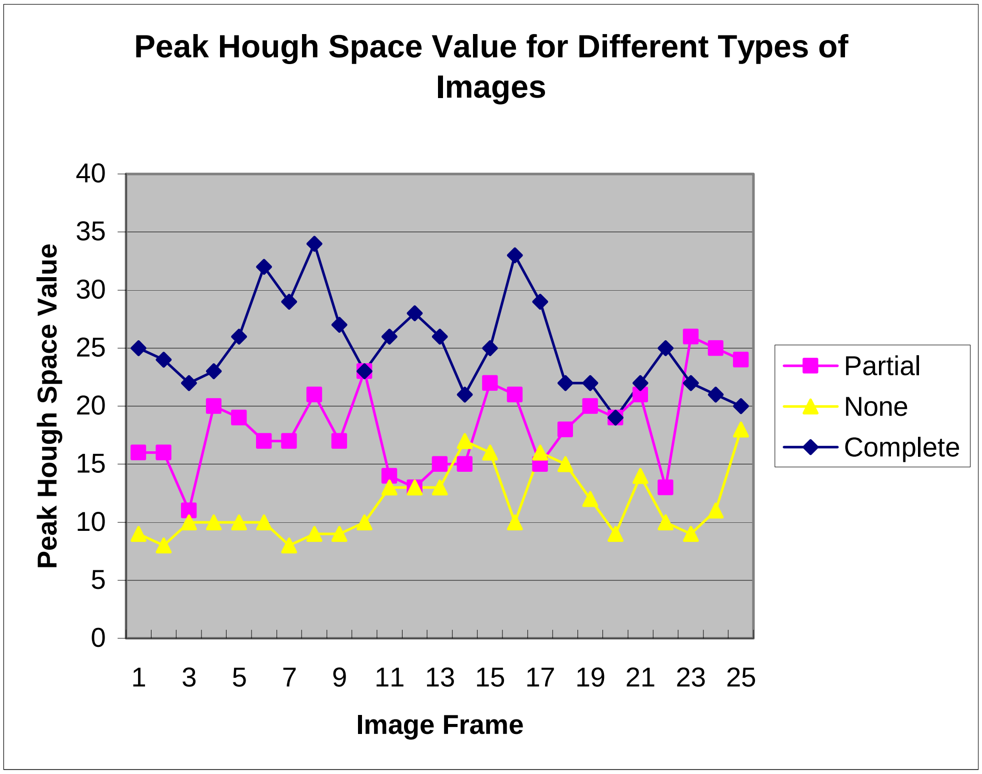

Figure 10 Peak Hough Space Value for Three kinds of Image Frames