A Traffic-Flow Parameters Evaluation Approach Based on Urban Road Video

Yonghong Yue *

Wuhan Digital Engineering Institute, 718 Luoyu Road, Wuhan, Hubei 430074, P.R. China
* Corresponding author's Email: yueyh09@yahoo.com.cn

Abstract: This paper presents an approach to evaluate traffic-flow parameters under urban road environment. Virtual line based time-spatial images are used for vehicles counting. Three error impacts in complex urban road environments are discussed and related methods are proposed to improve detection rate. These impacts include traffic congestion impact, environments impact and night vehicle headlight impact. The experimental results on real road videos show that the time-spatial method is robust in complex lighting and traffic environment.

Keywords: Time-spatial imagery; Traffic-flow counting; Vehicles detection; Error correction

1. Introduction

Video imagery has always been a very important source to get traffic information and been widely used in traffic monitoring and guidance area. Recent years, many megalopolises in China built real-time road video surveillance system. In most applications, video signal is captured by CCD/CMOS camera and transferred to a monitoring center, no processing is used for these signals. The purpose of the work presented here is to provide an urban road traffic-flow parameter statistical method based on video imagery.

Most traffic information in ITS is based on ground sensors like induction loops, bridge sensors and stationary cameras. Physical devices based system need pre-installed, thus restrict its use in China as well as high cost. A single camera can monitor multiple lanes along different roads without the professional installation and calibration requirements [1]. This feature conforms to the actual conditions of China urban road construction.

Some modules for traffic monitoring are proposed. The first module involves background detection and objects extraction. M. Vargas presents their vehicle detection work based on an enhanced background estimation algorithm [2]. They did not discuss the application in night and congestion situation. The second model involves frames subtraction methods [3], which employ a threshold technique over the inter-frame difference, where pixel differences or block difference (in order to increase robustness) has considered. The vehicle detection rate in congestion situation is unsatisfactory. The third model use virtual loop methods. Belle L. Tseng *et al.* [4], A. Liu *et al.* [5], J. Wu *et al.* [6] proposed methods based on virtual line for vehicle detection. But their experiment environments are different for us. Few of the existing vision system use urban road as detection environment.

In our system, we first propose virtual line based time-spatial image methods for urban road traffic-flow count. We considered different methods in complex environment. Road's light condition, vehicles' headlight impact and urban road environment impact have been considered in our algorithms. The algorithms meet the requirements and demonstrate good results. The virtual line based time-spatial image method is also the most convenient for multi area/lane counter in a single camera.

The paper is organized as follows. In section 2 describes the proposed methods and algorithms. Section 3 presents comparison experimental results and Section 4 concludes this paper.

2. Virtual line based vehicle count methods

2.1. Generation of Time-spatial image

Traffic volume is the most important traffic-flow parameters. In order to count vehicles in a period, we use a *virtual detection line* to generate time-spatial images. The selection of virtual line determines the analysis of vehicles as they cross the lines. In two dimension video image, the relationship between moving objects and static background isn't easy to be found. Since a vehicle usually takes more than one frame to cross a *virtual detection line* on the road, sometime memory must be included in the algorithm [7].

Each virtual detection line generates a corresponding sequence in video frames. We acquire a new frame to store certain lines of the frame onto what we have called *time-spatial image*. These stacks of lines contain all the required information to detect vehicle status, and can be considered itself as an image. [8]

Time-spatial image merges the characteristic both of temporal series and spatial sequence. It's good for reflecting differences of moving vehicles and static objects. In a time-spatial image, the image spreads in space and time coordinate as Figure1. (a).

Such images can be perceived as detector staring-map by a line detector. We could say that stacking acts like an information condenser, passing from a sequence from Time-spatial image slice. Different rows are reflected in the corresponding detection line in time. In the paper, a red line is selected as a line detector. As shown in Figure1.(a)

in chronological order from frame 1 to frame n these detection lines generated time-spatial images shown in Figure1.(b).

Using above method, time-spatial images reflect the image of a fixed region (line) changes with time. If no object moves in the detect region, there is no gray changed. Otherwise, gray will be changed by moving objects.

In certain sample frequency, the faster the movements object, the shorter it stays in the region and the contrary is longer. In the image, the length of gray changes t is inversely proportional to object velocity v. After normalized, it's defined as:

$$v=1/t \tag{1}$$

When a object is static, the length t (x axis) is infinite; When its speed is not zero, the length t is limited, with equation (1)

In a time-spatial image, horizontal axis is the time line shows vehicle time information, while the vertical one is space–axis which shows vehicle space information. The image height equal to the detection line (red line) length; and the image width (T) can be calculated as:

$$T=N*w/a \tag{2}$$

Where N is the number of frames, and a is the frame rate per second, w is the width of virtual detection line. It is fixed to 1 pixel. In the paper $N=150$, $a=15$. The time-spatial image stacks one line per frame, thus the image width is 150 pixels, and T is 10 seconds.
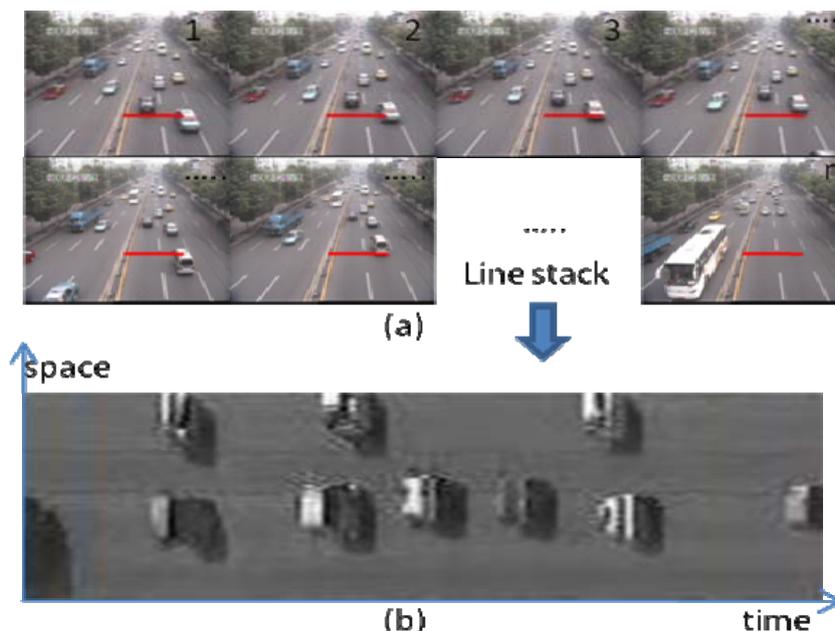

Figure1. The generation procedure of time-spatial image. (a) Frame sequence. (b) Time-spatial image generated by virtual line iteration.

2.2. Vehicle detection and counting

The time-spatial image facilitates the detection of vehicles as they cross the virtual detection line. After pre-processing of the image, vehicles are identified as dominant non-background objects. When a non-background object passes the virtual detection line and occupies over a threshold percentage of the lane width, a possible vehicle is detected.
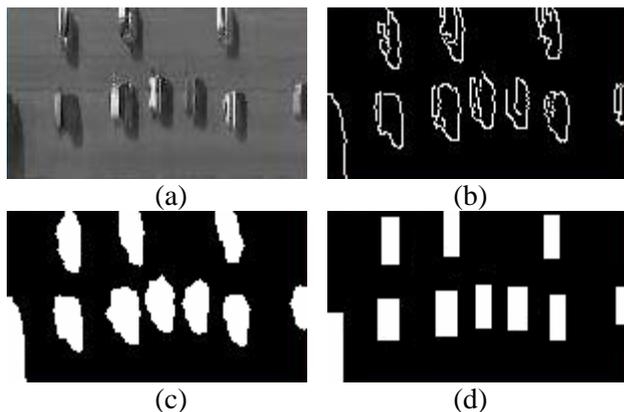

Figure2. Vehicle detection from time-spatial image. (a) Time-spatial image. (b) Image after edge detection. (c) Image after morphological close operation and hole-filling. (d) Image after moment computing.

We first obtain a time-spatial image shown as Figure2.(a), then Canny Edge Detector is used to detect edges as Figure2.(b) [9]-[11]. In order to extract vehicle objects from canny edge, image morphological closing operation and hole filling algorithm are used to get blobs of vehicles as shown in Figure2(c) and (d) shows the result after geometric moment computing. Thus, we can count vehicles in this period. For instance, in Figure2, the image width is 150 pixels and the count of moments is 10. It means there are 10 vehicles past the virtual detection line in 10 seconds.

To count vehicles in a long period, the position and height of vehicles on right and left edge of each time-spatial image should be recorded. If the character of left edge is matched with right edge of last image, the matched number should be subtracted from total.

2.3. Vehicle counting error analysis on urban Road

Urban road traffic is more complex than highway. On high way the distance of vehicles always is longer than 20 meters, as a result, it is enough to separate each vehicles by time-spatial image.

However, in heavy traffic status in urban road we always need count vehicles in night or in complex environment. In order to improve the accuracy, we need to consider following issues.

(1) Traffic Congestion counting error

In congestion situation, vehicles passing time on virtual line is longer than usual. The worst situation is vehicles stop on the virtual lines. The rectangle width will be very long. Since camera projection has an angle to road, vehicles image on time-spatial image may be overlapped. The number of vehicles may less than actual number.
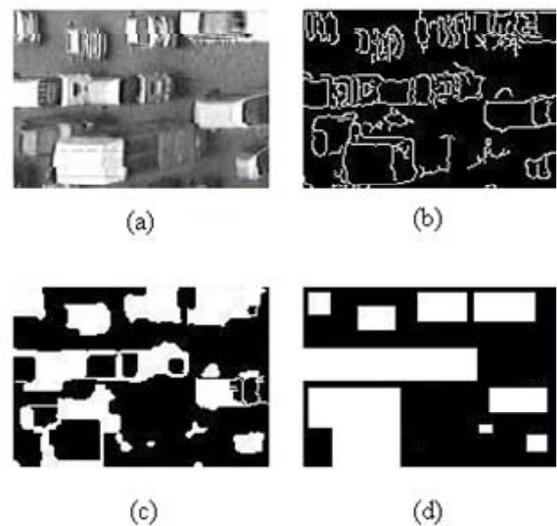

Figure3. A video frame of congestion situation (left). The length of virtual detection line is 106 pixels. (a) Time-spatial image in congestion situation. (b) Image after edge detection. (c) Image after morphological close operation and hole-filling. (d) Image after moment computing

We use vehicles width model to reduce overlapped error. Figure3 demonstrates the congestion situation, left is original image, (a)-(d) are image processing and counting procedure.

(2) Night counting error

Most counting error at night situation comes from incoming vehicles' headlight of incoming. As shown in Figure5 shows, the left lane of the original image is the incoming direction. When headlight shines on the sensor of camera, it comes to a white block. In this situation, headlight block and vehicles blocks are difficult to be separated. The number of vehicles may more than actual value, if we use vehicles length model to find light block. If possible, we choose virtual detection line near bottom of original image to avoid headlight shine straightly on the camera sensor. In the outgoing direction, the headlight error could be ignored. Figure 4 demonstrates the night situation for vehicle counting.

Figure4. A video frame of night situation with headlight impact (left). The length of virtual detection line is 110 pixels. (a) Time-spatial image in headlight impact situation. (b) Image after edge detection. (c) Image after morphological close operation and hole filling. (d) Image after moment computing
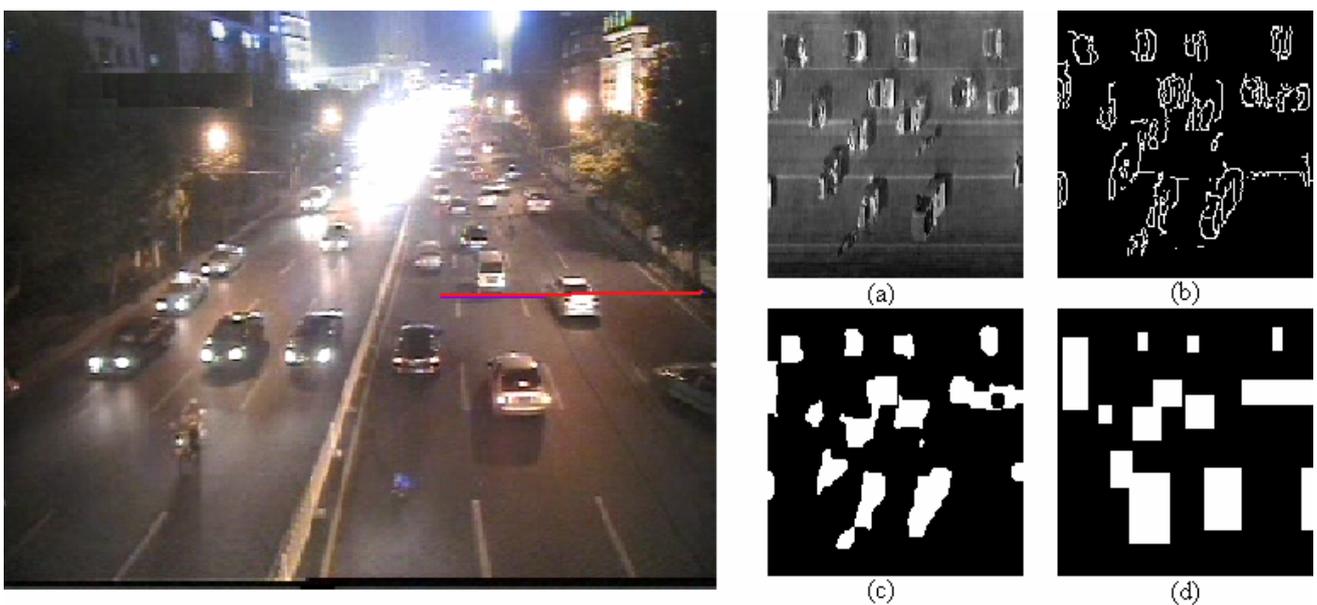

Figure5. A video frame of night situation with shadow (left). The length of virtual detection line is 158 pixels. (a) Time-spatial image in environment impact situation. (b) Image after edge detection. (c) Image after morphological close operation and hole filling. (d) Image after moment computing

(3) Urban road environment impact counting error

Urban road environment impact includes building and street tree shadow, pedestrian, bicycle, etc. Vehicles geometry model is used to filter small objects. The fix impact of shadows could be avoided by virtual line setting.

The unwanted shadow exists on daylight and night condition. To ensure the accuracy of vehicle detection, we adopt a method based on minimum boundary rectangle [12], which generates bounding boxes for individual vehicles.

Figure5 demonstrates unwanted shadow impact on night situation. In the figure, vehicle shadow comes from street lamps. The vehicles shadow's proportion is almost constant on virtual detection line position.

Based on the above analysis, a set of rules are proposed in virtual detection line setting. These rules are listed as follows:

Rule1: Set the line perpendicular to the direction of cars travelling;

Rule2: Span of the line includes vehicles' projection area on the road;

Rule3: Set the line near bottom of the original image to increase projected angle;

Rule4: Do not to cross lane line and unwanted shadow;

2.4. Vehicles counting procedure

The vehicle counting procedure by time-spatial image includes image preprocessing, Canny edge detection, image morphological operation, vehicle detection and counting, error correction etc.

The detailed steps are listed as follows:

Step1: Image preprocessing, covert color frame into gray;

Step2: Setup detection line using rule1-4. In our work, virtual line can be set on multi-lanes and bi-direction, so as to meet all kinds counting demand;

Step3: Generate Time-spatial image according to virtual lines;

Step4: Detect edges using Canny Edge Detector;

Step5: Detect vehicle objects using Morphological operation and moment computing. Then count the objects number and record edge information.

Step6: Error correction and accumulate total number;

Figure6 shows the flowchart of vehicles counting algorithm using time-spatial image.
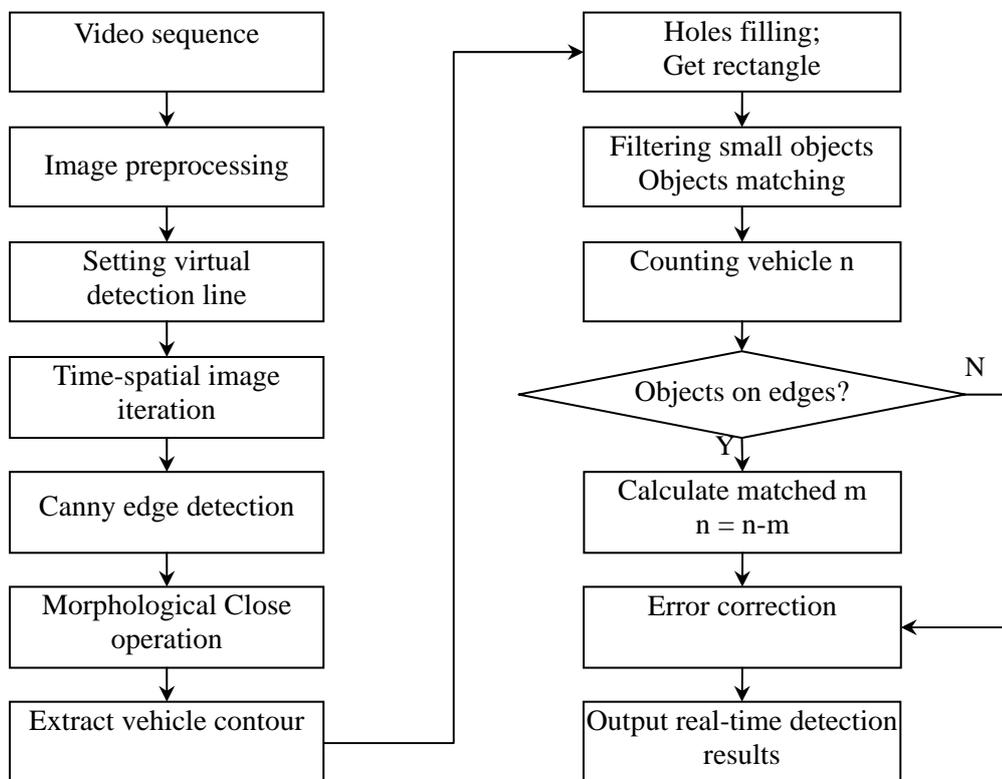

Figure6. Flow chat for urban road vehicle counting method based on time-spatial image.

3. Experimental results

The proposed method is evaluated on real video clips which covered more than 11 roads in Wuhan City of China. These video are collected in various traffic and light conditions: night and daytime. In some of the video, the camera is placed on the viaduct and jitters seriously. The virtual detection lines are set in different width for one lane, two lines, three lanes and bi-direction multi-lanes.

Original video frame is color image with size of 352*288 (CIF/PAL), and frame rate is 15FPS. Each frame is converted into eight bits gray scale in image preprocessing procedure.

Table 1 lists 11 sites detection results under two kinds light situation and different virtual line setting method. It can be seen the detection rate of proposed methods are from 82.54% to 98.39% and the total detection rate is 90.55%. The precision rates vary with the complexity of road traffic and environment situation. Generally, the average detection rates in daylight are greater than that of night.

Besides the reasons discussed in section 2.3, the false estimation results are also due to low contrast, small vehicle block and irregular lane direction, etc. In normal condition our experimental results showed high accuracy compared with manual results.

4. Conclusions

This paper presents a vehicle counting approach which is designed to operate under complex conditions like lighting transitions, traffic congestion, environment impact, vehicles headlight impact, etc.

The key element is the use of time-spatial virtual image to detect vehicles under complex urban road environments.

The hybrid method based on canny edge detector and morphological operation algorithm is proposed to implement preprocessing under different background model.

Although urban road counting precision is lower than that of highway situation which is normal 95%. [4], our work introduced the concept of time-spatial image in urban road vehicle counting. Experiment results on virtual line based vehicle detection show that the proposed method is feasibility and practical. As a result, an average performance of 90.55% vehicle count accuracy in different complex situation is achievable.

Future work would mainly setup traffic-flow parameters model with time-spatial image and optimize preprocessing algorithms.

Table 1 Precision comparison results under different environment

Site	Light condition	Virtual line setting	Actual	count	Precision
1	daylight	2lanes/incoming	35	32	91.43%
2	daylight	2lanes/outgoing	63	54	85.71%
3	daylight	1lanes/incoming	50	43	86.00%
4	daylight	2lanes/outgoing	88	76	86.36%
5	daylight	2lanes/outgoing	99	92	92.93%
6	daylight	2lanes/outgoing	69	59	85.51%
7	daylight	2lanes/outgoing	105	98	93.33%
8	daylight	3lanes/incoming	62	61	98.39%
9	night	2lanes/incoming	58	65	87.93%
10	night	2lanes/outgoing	70	58	82.86%
11	night	2lanes/outgoing	63	52	82.54%
Total			762	690	90.55%

References

[1] C. Setchell and E.L. Dagless, "Vision-based road-traffic monitoring sensor." In: *Vision, Image & Signal Proc.*, 2001.

[2] M. Vargas, S. L. Toral, F. Barrero, "An Enhanced Background Estimation Algorithm for Vehicle Detection in Urban Traffic Video", In: *Proc. of 11th International IEEE Conference on Intelligent Transportation Systems*, pp.784-790,2008.

[3] Guolin Wang, Deyun Xiao, "Review on Vehicle Detection Based on Video for Traffic Surveillance", In: *Proc. of the IEEE International Conference on Automation and Logistics 2008,*pp.2961-2966, 2008

[4] Belle L. Tseng, Ching-Yung Lin and John R. Smith, "Real-time Video Surveillance for Traffic Monitoring using Virtual Line Analysis", In: *Proc. Of 2002. IEEE International Conference on Multimedia and Expo*, pp.541 – 544, vol.2 2002.

[5] A. Liu, Z. Yang, J. Li, "Video Vehicle Detection Algorithm based on Virtual-Line Group", In: *Proc. of IEEE APCCAS*, pp.1148-1151, 2006

[6] J. Wu, Z. Yang, J. Wu, A. Liu, "Virtual line group based video vehicle detection algorithm utilizing both luminance and chrominance", In: *Proc. of IEEE IEAC*, pp. 2854-2858, 2007

[7] Antonio Albiol, Inmaculada Mora, and Valery Naranjo, "Real-time High Density People Counter Using Morphological Tools", In: *IEEE Trans. Intelligent Transportation System*, vol. 2, No.4, pp.204-218, Dec. 2001.

[8] L. Li, L. Cheng, "A Real-Time Congestion Estimation Approach from Video Imagery ", In: *IJIES Volume 1Issue 2, June 2008*, pp.1-9, 2008

[9] X. Ma W. Eric L. Grimson, "Edge-based rich representation for vehicle classification", In: *Proc. of the Tenth IEEE International Conference on Computer Vision*, pp1-8, 2005.

[10] Hiroshi Inoue, Mingzhe Liu, and Shunsuke Kamijo, "Vehicle Segmentation by Edge Classification Method and the S-T MRF Model", In: *Proc. of the IEEE ITSC*, pp.1543-1549, 2006.

[11] J. Canny. "A Computational Approach to Edge Detection", In: *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 8(6), pp.679-698, 1986.

[12] B. Li, Q. Chen, "Freeway Auto-surveillance From Traffic Video", In: *Proc. of 6th International Conference on ITS Telecommunications,* pp.167-170, 2006.