Real-Time Vehicle Counting Method Based on Image Sequences with Laser Line

Yun Ye, Xingang Wang

***Abstract*—Being robust to changeable illumination and shadow are *e*ssential requirements of a practical vehicle counting system. In this paper, we come up with a method that detects vehicle by analyzing the state of laser line projected on lane. In addition, we use the entropy of histogram of oriented gradients (HOG) descriptor extracted from the region of interest (ROI) to quantify area percent of vehicle in a ROI. Normalized HOG descriptor is with better invariance to illumination and shadow, which makes a great contribution to the robustness of our method. Once the laser line disappears or appears in an image, we use entropy of HOG to judge that whether a vehicle is driving into or driving out of the ROI. We tested this method under different conditions, experimental results also prove that our method is robust to the change of illumination, and shadow of vehicles has no impact on accuracy and recall rate of detection result.**

I. INTRODUCTION

OVER the last three decades, intelligent transportation system (ITS) is playing an increasingly important role in our daily life, and reliable traffic flow information is the key of several elementary links in ITS, such as traffic flow modeling [1], [2], and traffic behavior analysis [3], [4]. It's quite crucial to develop a reliable vehicle counting system due to the important role it plays.

There are mainly two branches in current vehicle counting methods: either by modeling the background and then extracting the foreground in current image [5]-[7], or by detecting the foreground through an object detector directly [9]-[12]. As for the former, the simplest way is to treat the background as static or changing gradually [5], but this method can't handle the situation that luminance changes suddenly. Besides, this counting system needs a picture of current scene as background each time it begins to execute. To make the background modeling method more robust, Gaussian Mix Model(GMM) was introduced to modeling the background [6], [7], values on each pixel are modeled by a mixture of K Gaussian distributions with specific weight parameters, the calculation is of pixel level, so this algorithm is exactly time-consuming. Authors of literature [8] attempted to reduce computation complexity by carrying out Kalman filter to update the current background, but it still requires these sequence of images change gradually. In general, background model based method can't adapt to those situations where luminance differs greatly among continuous frames, what's more, shadow of vehicles may reduce accuracy of counting result as well.

As for the latter, the typical one is to train a classifier to detect vehicles in current image frame. Both supervised learning algorithms and unsupervised learning algorithms have been applied to training classifier. Training a boost classifier on Haar-like feature to detect vehicles has been proved to be a feasible supervised learning based method [9], [10]. With the rapid development of deep learning in computer vision, CNN is also used to detect vehicles [11], due to the extreme high computation complexity, we still can't use CNN to detect vehicles in real time even using GPU. One common shortage of those supervised learning algorithm is that they all need plenty amount of data to train the classifier off-line, a same system applied to different scenes may need to collect training data repeatedly. To overcome this shortcoming, authors of [12] try to transform the process of detecting vehicles to a task of Bayesian density estimation, which related the number of vehicles to pixels that may correspond to vehicles in current frame, another solution is to combine the background modeling method with foreground modeling method [13], this method first uses enhanced adaptive background mixture model to identify positive and negative examples, and then uses these samples to train a SVM detector. For most of machine learning based method, the vehicle detector needs to be trained again when position of camera changes, this is quite inconvenient, otherwise, the processes of feature extracting and feature matching are time-consuming.

This paper proposes an algorithm that bases on a laser line projected on lane, this laser line will broke into two parts once a vehicle drives into the ROI, and then, this line will recover to a whole line again after the vehicle drives out of this ROI. However, when partial edge of vehicles overlaps with this line, the disconnected laser line will be detected as a whole line by mistake, to avoid this false detecting, we suppose extracting HOG feature of the current frame, and using the entropy of HOG to judge if there are vehicles in ROI. Our algorithm determine state of a ROI via HOG descriptor, which has better invariance to illumination and shadow, so this method can still work well even when luminance varies greatly between continuous frames. Besides, entropy of HOG makes this algorithm can handle such situation that vehicle drives slowly within ROI, or even stops in ROI and shadow of vehicles will not reduce the detecting accuracy. What's more, this method does not train a classifier to detect vehicles, so it does not require users to collect training samples before applying it to a specific scene, and it only needs to compute entropy of HOG when there are vehicles in a ROI, so it can

Yun Ye is with the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China (e-mail:y423876867@163.com).

Xingang Wang is with the Research Center of Precision Sensing and Control, Institute of Automation, Chinese Academy of Sciences, Beijing, 100190, China (e-mail:xingan.wang@ia.ac.cn).

count vehicles in real time, those experiments we performed said that it could execute at an average rate of 30 frames per second.

II. ALGORITHM DESCRIPTION

A. Algorithm Overview

Procedures of our algorithm include following steps: image pre-processing, laser line detection, ROI state analyzing. Each frame of a video inputted into this algorithm needs to be pre-processed. To detect laser line in an image, we applied morphology transform called Dilate to the grayscale image, and then we used Hough Transform algorithm cooperating with some priori knowledge of laser line to detect the laser line in a ROI. Finally, we combined the state of laser line with entropy of HOG to determine if there exists vehicle in a specific ROI. The block diagram of this system is demonstrated as following.

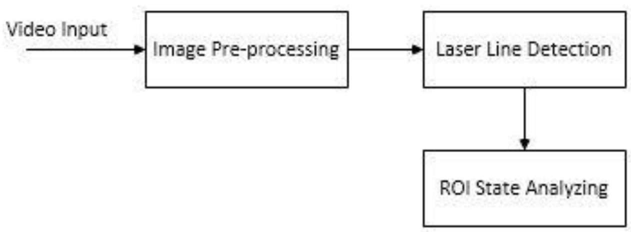

Fig. 1. Block diagram of this algorithm

The innovation of this method is using information provided by a laser line, and using entropy of HOG descriptor to quantify area percent of vehicle in a ROI when the laser line is broken.

B. Image Pre-processing

In a practical computer vision based system, image pre-processing is often applied as the first step to transform images to such kinds that are more suitable for this system running efficiently. Our system also obeys this principle. Firstly, we applied Gaussian filter to the original image to reduce noise in an image. Secondly, we turned the RGB image into grayscale image. Thirdly, we used Canny algorithm to extract edge of the grayscale image that had been smoothed, so far, we have obtained edge that contains information of all possible lines.

C. Laser Line Detection

To make this algorithm work efficiently and accurately, and make it adapt to various environment as well, we projected a laser line on lane, once the laser line disappears or is separated into two parts, this algorithm then calculates entropy of HOG of the ROI to confirm that whether there exists vehicle within the ROI. The foundation of this algorithm is detecting laser line in current frame of image. Hough Transform is a mature method to detect all lines in an image, and we also applied it to our algorithm. The laser line projected on laser lane is shown in the figure 2, it can be seen that width of laser line is not uniform, the smallest width of a laser line may be even only several pixels in an image, when we apply edge detection to such images, those laser lines will be recognized as disconnected in the most narrow point by mistake. To make sure that this algorithm detects laser line correctly, we applied morphology transform called Dilate to edge of image, those disconnected points will be recover to continuous again. Then, Hough Transform is applied to those processed images to obtain lines candidate. Finally, we used some prior information of laser line to distinguish it from lines candidate.

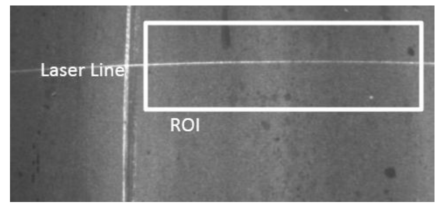

Fig. 2. Laser line and ROI in an image

Procedures of determining the laser line can be described as following:

Step1. Applies dilate processing to edge that is extracted before, and then uses Hough Transform to obtain all lines in a specific ROI;

Step2. Applies filter to exclude those lines with large gradient, short length, and exclude those lines lie on inappropriate position as well;

Step3. Acquires the longest line in line candidate, and marks it as the laser line.

Once the longest line is shorter than width of ROI, the laser line may be truncated by a vehicle, and then, the entropy of HOG feature is used to determine whether there exist vehicles in this ROI, and when vehicles drive out of this ROI.

D. ROI State Analyzing

There are two possible when laser line is disconnected in a ROI, one is that a vehicle is just driving into the ROI and truncates laser line projected on lane, unstable illumination, such as illumination of vehicles' headlights and illumination of sunlight, will also make this algorithm detect disconnected laser line. There are also two possible when computer detects a connected laser line, one is that the laser line projected on lane is actually connected, the other is that computer recognizes edge of vehicles as the laser line, to avoid those false detections under these two situations, we come up with using the entropy of HOG in the ROI to determine whether to change state of counter or not. Results of our experiments said that information brought by entropy of HOG can really help to reduce those false detections caused by unstable illumination and edge of vehicles, in addition, the adverse effect brought by shadow of vehicles is also restrained via introducing entropy of HOG to our algorithm.

HOG feature has been used widely in human detection [14], [15], the essential thought of HOG descriptor is that appearance of object can be described by the distribution of intensity gradients. In our algorithm, we define a cell as a region containing 8×8 pixels, and define a block as a region containing 16×16 pixels with a block stride of 8×8(that mean a block contains 4 cells, and adjacent blocks have an overlap region with two cells). For each pixel in a cell, we calculated its histogram of gradient directions, and then, concatenated them to the initial HOG descriptor to a cell. Then, we normalized HOG feature of each cell that distributes in the same block, this normalization step can obviously improve robustness of our algorithm to changes in illumination and shadowing [14]. For each cell in a ROI, we can get entropy of HOG, and use this information to judge whether there exist vehicles within a ROI.

This algorithm first get the HOG feature for each cell in a specific ROI, value of ith box in the histogram can be marked as $H_i(i = 1, 2, 3, \dots n)$. HOG descriptor of a whole image can be visualized as figure 3. In our experiment, number of boxes in a histogram of orientation is nine (n = 9), so each box ranges over forty degrees. If H_m is maximal among $H_i(i = 1, 2, 3, \dots n, i \neq m)$, it's most possible that the main direction of this cell is within the range $[\frac{360°}{n}(m-1), \frac{360°}{n}m)$, the larger the difference between H_m and $H_i(i = 1, 2, 3, \dots n, i \neq m)$, the larger confidence that the main direction of this cell is inferred to be within the range of $[\frac{360°}{n}(m-1), \frac{360°}{n}m)$. We define entropy of HOG of a cell as following:

$$E = -\sum_{i=1}^{n}\left(\frac{H_i}{\sum_{j=1}^{n}H_j}\log_2\frac{H_i}{\sum_{j=1}^{n}H_j}\right) \quad (1)$$

As the entropy can be used to quantify the state of disorder, confusion, and disorganization, we introduce entropy of HOG for quantifying the disorder of gradients of a cell. In other words, the smaller the entropy of HOG is, the more certain that the main direction of this cell is within an exact interval. In general, road has no special texture, if the ROI contains only road, gradients of most of cells within this ROI distribute uniformly in all directions. That is, each box in the histogram of oriented gradient has almost the same value, thus the entropy of HOG feature of most of cells within this ROI is large. Otherwise, if there exists vehicle within a ROI, the percentage of cells with lower entropy of HOG will rise.

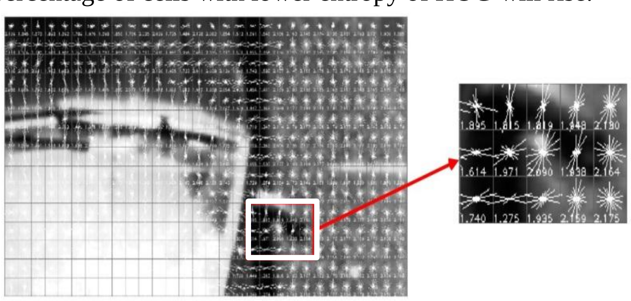

Fig. 3. Visualization of cell and HOG feature. Each grid in the left figure is defined as a cell, we distribute all gradients of a cell into 9 intervals according to their directions, each interval ranges forty degrees. In each cell in the right figure, value of HOG feature in a specific direction can be visualized as the length of a segment, and the corresponding entropy of HOG is listed on the bottom of each cell. We can see that if an image contains a vehicle, the texture will be complicated, and the percentage of cells with lower entropy of HOG will be higher.

Having introduced the definition of entropy of HOG, we can know that, in this special application, a cell tends to be background if gradients distribute chaotically within this cell as well as the entropy of HOG is large. For a specific ROI, we define the jst cell as part of background if it satisfies the following formula: $E_j > T$ (T is a threshold), and mark it as C_b, we infer that there exit no vehicles in this ROI if percentage of C_b within this ROI is higher than a specific threshold P. So, if the algorithm detects laser line disappears in the ROI, it locks the counter, then calculates the entropy of HOG descriptor to judge whether there exists vehicle in this ROI. If there are no vehicles within this ROI, this algorithm will unlock the vehicle counter, otherwise, it pluses the vehicle counter by one, and the vehicle counter will not unlock until this algorithm infers vehicles have driven out of ROI.

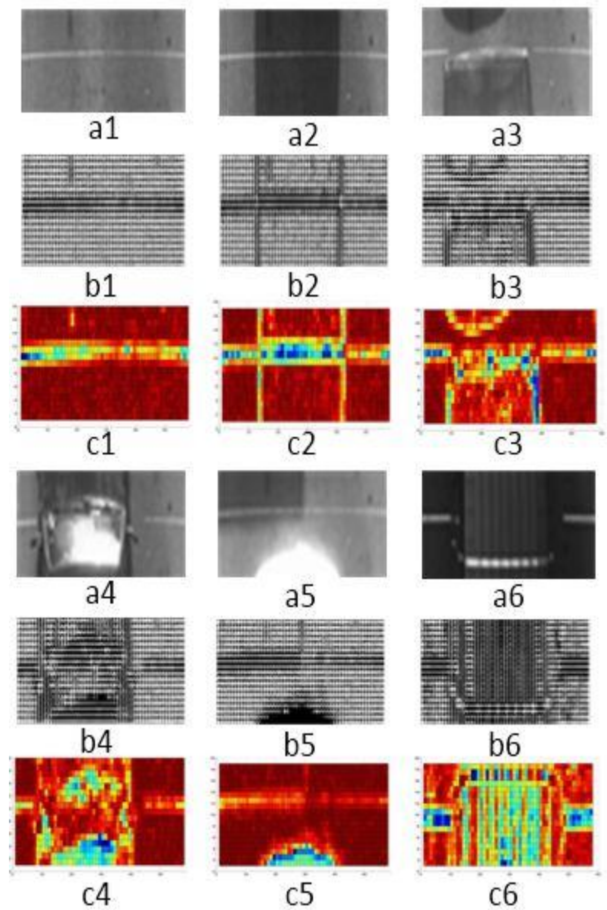

Fig. 4. Visualization of experiments, ai are original images of ROI, bi are visualized HOG feature of those ROI, ci are visualized entropy of HOG corresponding to those ROI(i = 1, 2, 3, 4, 5, 6), cells painted with warm color are with higher entropy of HOG(cells painted with deep red have highest entropy of HOG, cells painted with deep blue have smallest entropy of HOG).

In figure 4, we extract six images from videos. It can be seen that the percentage of C_b declines when there exists vehicle in a ROI. For example, by comparing the first frame and the sixth frame in figure 4, we can see that percentage of cells painted with deep red in c1 is higher than that in c6, so cells with high entropy of HOG in c1 is obvious more than that in c6. We can quantify the percentage of area occupied by vehicle in an image through the entropy of HOG, so entropy of HOG can be applied to judge state of ROI. Key procedures of this algorithm in detection and counting within a specific ROI are described as following :

TABLE 1
INFORMATION OF VIDEOS

Video	Shadow of Vehicles	Illumination Intensity	Illumination variation	Density of Traffic	Exposure Time
1	invisible	quite weak	stable	2s	100ms
2	invisible	weak	stable	2s	200ms
3	visible	changeable	change gradually	2s	3ms
4	invisible	changeable	change gradually	2s	16ms
5	visible	changeable	change sharply	2s	15ms
6	visible	strong	stable	2s	8ms
7	invisible	normal	stable	2s	80ms

TABLE 2
RESULT OF EXPERIMENT

Video	Ground Truth	Background Modeling Method (GMM)				Our Algorithm			
		FP	FN	Recall	Accuracy	FP	FN	Recall	Accuracy
1	15	2	5	66.7%	83.3%	0	0	100%	100%
2	28	6	3	89.2%	80.6%	0	6	78.6%	100%
3	147	0	19	87.1%	100%	3	0	100%	98%
4	87	1	32	63.2%	98.2%	0	1	98.9%	100%
5	54	6	0	100%	90.0%	2	0	100%	96.4%
6	53	2	2	96.2%	96.2%	2	0	100%	96.4%
7	88	3	11	87.5%	96.3%	0	6	93.2%	100%

Step1. Detects laser line, if laser line disappears in this ROI, this algorithm goes to step2, otherwise, it repeats step1;

Step2. Calculates the HOG descriptor for this ROI, obtains the entropy of HOG for every cell in this ROI, calculates percentage of C_b in this ROI, then, this algorithm goes to step 3;

Step3. If the percentage of C_b is lower than threshold P, pluses number of vehicles by one, then calculates the new percentage of C_b in new frame, this algorithm will not return to step1 until the new percentage of C_b is higher than P, as well as the laser line appears in the ROI again.

III. EXPERIMENTAL RESULTS

We tested this algorithm using seven videos, information of these videos is listed in table 1, each video contains five aspects of information. We grade intensity of illumination as five levels: quite weak, weak, normal, strong, quite strong. To make sure that our algorithm can detect laser line in an image, we did not take situation with quite strong illumination into account (such situation that the laser line can't be seen by humans). Besides, we quantified density of traffic by time interval of adjacent vehicles driving through ROI on average. For better analyzing experimental results, we listed camera exposure time in the last column. We compared this algorithm with GMM background modeling, experimental results are listed in table 2.

To determine the threshold of T, we collect 30 frames from these videos, including images with different illumination and different types of vehicles, we cropped image region containing vehicles and calculated average entropy of HOG for each region, finally chose the value that is 1.3 times of the biggest average entropy as T. To determine the threshold of P, we collected 25 images (ROI) without vehicles and 25 images(ROI) with vehicles, and then, we calculated percentage of C_b in these ROI, and chose the P that can classified region with vehicle or without vehicle correctly.

As for experimental results, we define the number of times that one method pluses vehicle counter by mistake as false positive (FP), and define the number of vehicles that are missed as false negative (FN), so the recall rate of counting result can be described as:

$$\frac{Ground\_Truth-FN}{Ground\_Truth}\times 100\% \tag{2}$$

The accuracy of counting result can be describes as:

$$\frac{Ground\_Truth-FN}{Ground\_Truth-FN+FP}\times 100\% \tag{3}$$

The Ground_Truth is the actual number of vehicles in each video.

By analyzing results corresponding to third, fourth and fifth videos listed in table 2, we can see that our algorithm can reduce adverse effects caused by unstable illumination. After referring to these three videos, we found that 66.7% of false detections are caused by re-counting trucks for twice, and these false are permitted in traffic flow statistics.

By looking into results corresponding to third, fifth, sixth videos listed in table 2, reason of false detecting is also re-counting trucks for twice, so it is obvious that shadow of vehicles have no effect on accuracy of our algorithm. On such situation that the illumination is quite weak (video 1), our algorithm can get a pretty good result, since the laser line can be detected easily, but such condition is a challenge to the background modeling method on the contrary, because it is hard for the background modeling method to extract foreground from a quite dark image.

However, if camera exposure time is set to a high value (video 2), images of ROI would become not clear enough for our algorithm to detect vehicles sensitively, this resulted in the low recall rate of our algorithm in experiment corresponding to video 2. Two examples of FN are shown in figure 5.

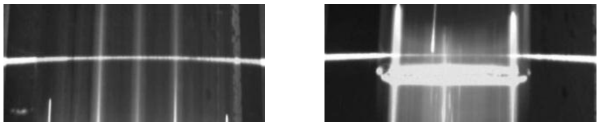

Fig. 5. Two examples of FN in video 2, due to long camera exposure time, laser line can still be detected even when vehicle is on it.

IV. CONCLUSIONS

In this paper, we propose a novel method to count vehicles from images, the framework of this algorithm is based on laser line detection and entropy of HOG feature. We introduce the entropy of HOG descriptor to quantifying area percent of vehicle in a ROI. Results of experiments said that our approach is robust to the change of illumination, it performs well under those conditions that illumination ranges widely, and shadow of vehicles has no influence on the counting accuracy as well as recall rate, and what is practical is that it can execute in real time (more than 30 frames per second).

By referring to results of experiments, it is clear that a long camera exposure time has impact on recall rate of counting result, because the laser line will be detected even when vehicle is driving over the laser line. For future work, improving counting recall rate on such situation would be a necessary step.

REFERENCES

[1] J. J. Sanchez-Medina, M. J. Galán-Moreno, and E. Rubio-Royo, “Traffic signal optimization in ‘La Almozara’ district in Saragossa under congestion conditions, using genetic algorithms, traffic microsimulation, and cluster computing,” IEEE Trans. Intell. Transp. Syst., vol. 11, no. 1, pp. 132–141, Mar. 2010.

[2] J. Wei, A. Wang, and N. Du, “Study of self-organizing control of traffic signals in an urban network based on cellular automata,” IEEE Trans. Veh. Technol., vol. 54, no. 2, pp. 744–748, Mar. 2005.

[3] X. Ma and I. Andreasson, “Behavior measurement, analysis, and regime classification in car following,” IEEE Trans. Intell. Transp. Syst., vol. 8, no. 1, pp. 144–156, Mar. 2007.

[4] D. Yao, Y. Zhang, L. Li, Y. Su, S. Cheng, and W. Xu, “Behavior modeling and simulation for conflicts in vehicles-bicycles mixed flow,” IEEE Intell. Transp. Syst. Mag., vol. 1, no. 2, pp. 25–30, 2009.

[5] Chen, Thou Ho, Y. F. Lin. "Intelligent Vehicle Counting Method Based on Blob Analysis in Traffic Surveillance." Innovative Computing, Information and Control, 2007. ICICIC '07. Second International Conference on IEEE, 2007, pp.238-238.

[6] Hyeok Jang, In-Su Won, and Dong-Seok Jeong. “Automatic Vehicle Detection and Counting Algorithm”. International Journal of Computer Science and Network Security, vol. 14, pp. 99-102, September. 2014.

[7] E. Bas, M. Tekalp, and F. S. Salman. “Automatic vehicle counting from video for traffic flow analysis”. In Proceedings of IEEE Intelligent Vehicles Symposium, pp. 392–397, 2007.

[8] J. Zhou, D. Gao, and D. Zhang, “Moving vehicle detection for automatic traffic monitoring,” IEEE Trans. Veh. Technol., vol. 56, no. 1, pp.51–59, Jan. 2007.

[9] S. Sivaraman and M. Trivedi, “Integrated lane and vehicle detection, localization, and tracking: A synergistic approach,”IEEE Trans. Intell. Transp. Syst. , vol. 14, no.2, pp. 906–917, June .2013.

[10] Haselhoff, Anselm, and A. Kummert. "An evolutionary optimized vehicle tracker in collaboration with a detection system." Intelligent Transportation Systems, 2009. ITSC '09. 12th International IEEE Conference on IEEE, 2009, pp.1-6.

[11] Lichao Huang, Yi Yang, and Yafeng Deng. “DenseBox: Unifying Landmark Localization with End to End Object Detection,” unpublished.

[12] Takayuki Katasuki, Tetsuro Morimura, and Tsuyoshi Ide, “Bayesian unsupervised vehicle counting,” In Technical Report, IBM Research RT0951, 2013.

[13] Tamersoy, Birgi, and J. K. Aggarwal. "Robust Vehicle Detection for Tracking in Highway Surveillance Videos Using Unsupervised Learning." Proceedings of the 2009 Sixth IEEE International Conference on Advanced Video and Signal Based Surveillance IEEE Computer Society, 2009, pp.529-534.

[14] N. Dalal and B. Triggs. “Histogram of oriented gradients for human detection”. In Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recogniza, 2005.

[15] Zhen-Rui Wang, Yu-Lan Jia, Hua Huang, and Shu-Ming Tang. “Pedestrian Detection Using Boosted HOG Features”. In Proc. IEEE. Conf. Intellgent Transportation Systems, 2008.