

# An Efficient Vehicle Model Recognition Method

Huihua Yang

Guangxi Experiment Center of Information Science, Guilin, China  
Email: yanghuihua@tsinghua.edu.cn

Lei Zhai, Lingqiao Li, Zhenbing Liu, Yichen Luo, Yong Wang

School of Electronic Engineering and Automation/Guilin University of Electronic Technology, Guilin, China  
Email: keylei203@gmail.com, zblu@guet.edu.cn, 54pe@163.com, louisluo@guet.edu.cn, wang@guet.edu.cn

Haiyan Lai, Ming Guan

Guangxi Communications Investment Group CO., LTD, Guilin, China  
13517810019@163.com, gm9099@sina.com

**Abstract**—An efficient vehicle model recognition method based on Adaptive Harris corner detector is presented in this paper. First, the vehicle radiator grid is selected as ROI and Harris corner detection is used to detect corner as vehicle model features, to solve a problem of inconsistencies in the number of corner between different models or the same model in different environment. Second, an adaptive threshold function is constructed to control the number of corner replacing a fixed threshold, ensuring that the image is always able to produce a certain number of strong corners. Third, a parallel scheme is designed to accelerate the vehicle recognition algorithm via GPU/CPU heterogeneous computing model to meet real-time requirement, which includes parallelization of algorithm and parallelization of process. The experiments on 1096 big truck images of 12 vehicle models obtain the recognition accuracy rate of 99.5%, and achieve 58x speedup on average by a platform with Intel Core i5 2400 and NVIDIA C2075. The results show that our proposed method can meet the requirements of practical application.

**Index Terms**—vehicle model recognition; ROI positioning; adaptive Harris Algorithm; GPU/CPU collaborative computing

## I. INTRODUCTION

Vehicle model recognition is an interesting and difficult subject of intelligent transportation study. In [1] and [2], features of corner point are utilized to achieve the coarse classification of cars, buses, heavy goods vehicles (HGVs) etc. In [3], an interesting approach for vehicle model recognition from frontal view vehicle images is presented, whose recognition accuracy achieves 93%. The vehicle manufacture and model were treated as a single class and recognized simultaneously, but no results for recognition speed were reported. In [4] and [5], a comparative knowledge acquisition system were introduced, consisting of several object recognition modules which represent a car image viewed from the rear, such as a window, tail lights, and so on, based on color recognition. This approach has the drawback of

being sensitive to lighting condition. In [6], a vehicle model recognition method was presented which was extracted of textural features of the radiator grille using based gray level co-occurrence matrix (GLCM), but ROI positioning is difficult in an outdoor environment. In [7], scale invariant feature transform (SIFT) features is proved to be suitable for vehicle manufacturer and model recognition, but it does not have real-time performance.

In summary, the above model identification methods still can't be a good solution to the actual vehicle recognition task in an outdoor environment. There are two main reasons:

1. ROI Positioning and feature extraction are difficult in the complex outdoor environment.
2. Recognition algorithm can't meet the real-time requirement for practical applications.

In the same time, GPU/CPU collaborative computing has become more and more important in data processing of computing-intensive tasks, and attracts the attention of many application developers in recent years<sup>[9][11,12]</sup>. This is because GPU can offer extensive resources even for non-visual, general-purpose computations: massive parallelism, high memory bandwidth, and general purpose instruction sets. GPU/CPU collaborative computing uses CPU to process sequence recognition tasks and uses GPU to process a large number of repetitive repeatability calculations to improve the key performance of applications. GPU/CPU collaborative computing is widely used in the field of image processing<sup>[10]</sup>. To the best of our knowledge, there is no report about the application of this accelerated technology in the vehicle model recognition fields.

This paper present a new vehicle model recognition based adaptive Harris corner detector. This is a part of the "JT-G green channel inspection system", which uses vehicle model recognition technology based on machine vision to achieve the cab's automatic detection and safe avoidance. First, radiator grid is positioned as ROI based on logo. Then, using the adaptive Harris corner detector to detect ROI's corner, an adaptive threshold function is

constructed to control the number of corner replace a fixed threshold. This method can ensure the image always produce a certain number of strong corner, avoiding dramatic changes of corner number between the different models, or the same model in different environment, thus it can greatly improve the recognition accuracy and robustness of the recognition.

Besides, a parallelization scheme for vehicle model recognition is designed, including the parallelization of adaptive Harris algorithm and the parallelization of workflow, to ensure that the algorithm has a fast response time and meet real-time requirement.

II. ROI POSITIONING

Selection and robust positioning of ROI in complexity field environment is difficult. In [6], the vehicle radiator grid as ROI completes a good vehicle recognition effect. This paper also uses this method, but adds logo's information in order to increase ROI's distinguishing degree, as shown in Fig. 1.



Figure 1. ROI selection of vehicle model recognition

The ROI is positioned based on logo's position. The position of the number plate region is steady-going and fixed in size. If the number plate can be fast and coarse detected, it can also be obtained the vehicle logo position in the vertical direction. The relationship between the number plate and the vehicle logo is as shown in Fig. 2. General logo is always at right above the license plate's position, we only need to detect logo in the vertical direction or position and the large white rectangle image is as the input image.



Figure 2. Through the license plate to determine the approximate location of logo

To position ROI, logo vertical position is needed to determine. Algorithm 1 describes the process of vertical position location for vehicle logo.

**Algorithm1:** Location for logo's vertical position.

**Input:** Grayscale image of extracted based on plate.

**Output:** The binary image after coarse location.

**Step:** a. Suppress horizontal texture using operator [1,-1].

$$D_x(i, j) = |g(i, j) - g(i + 1, j)| \tag{1}$$

b. Suppress vertical texture using operator[1,-1]<sup>T</sup>.

$$D(i, j) = |D_x(i, j) - D_x(i, j+1)| \tag{2}$$

c. Suppress noise.

$$f(i, j) = \begin{cases} 1 & D(i, j) > 0 \\ 0 & D(i, j) = 0 \end{cases}$$

$$sum(i, j) = \sum_{i-r}^{i+r} \sum_{j-r}^{j+r} f(i, j) \tag{3}$$

$$pixel(i, j) = \begin{cases} 255 & sum(i, j) > T \\ 0 & sum(i, j) \leq T \end{cases}$$

$$T = r \times r / 2 - 1$$

Where *r* is kernel's size, *T* is adaptive threshold.

d. The edge extraction: detect edge using Canny operator, the output is *pixel*\*.

e. Obtaining the row that its pixel values is maximum as the horizontal axis.

ROI positioning for logo vertical position is shown in Fig. 3. Where *h* is plate's height, *w* is plate's width, *y* is vertical position of logo, *n* is an empirical parameter.



(a) Coarse location based on plate



(b) Location of logo vertical position



(c) Location of ROI based on logo

Figure 3. Location of ROI

### III. ADAPTIVE HARRIS ALGORITHM

#### A. Harris

Corner is known as interesting points in the image. They are discrete, reliable and meaningful. Feature-corner is shown to perform with good consistency on natural imagery, this paper selects corner as characteristic of vehicle's model.

There are many corner detectors, such as Harris, Fast, SIFT, SURF and so on. Fig. 4 is a corner-image detected by Harris, SURF and SIFT respectively.

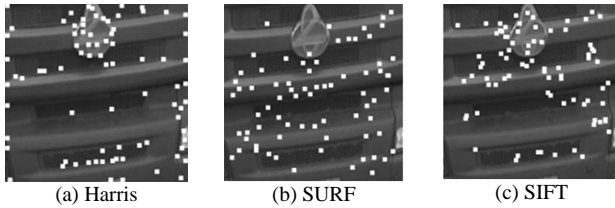


Figure 4. Corner-image detected by three detectors

As shown in Fig.4, Harris corner detector has the best results in the view of the positioning accuracy of corner. Since Harris corner detector is based on first order Hessian matrix, so it has higher detection accuracy. SIFT or SURF is based on High-dimensional scale space, and has higher dimensional feature vectors, robustness better than Harris. Because only information of corner's position is used in the process of matching, we selected Harris corner detector as this paper's detector.

#### B. Harris corner detector

Harris corner detection is a classic corner detection algorithm proposed by Harris C and Davis L S.A in 1988. It is a signal-based feature-corner extraction operator, assuming window  $W$  as processed image, the minute displacement  $(u, v)$  moving in any direction, then the gradation change amount can be defined as (4):

$$\begin{aligned}
 E(u,v) &= \sum_{x,y} w(x,y)[I(x+u,y+v) - I(x,y)]^2 \\
 &= Au^2 + 2Cuv + Bv^2 \\
 &= [u \ v]M[u \ v]^T
 \end{aligned}
 \tag{4}$$

Where  $M = \begin{bmatrix} A & C \\ C & B \end{bmatrix}$  is a symmetrical autocorrelation matrix,  $A = X^2 \otimes w$ ,  $B = Y^2 \otimes w$ ,  $C = (XY) \otimes w$ ,  $w(x,y)$  is window function.

Harris's feature is defined as the maximum value of the local, and its response function is:

$$\begin{aligned}
 CRF &= \Delta M - K(\text{trace}M)^2 \\
 \Delta M &= \lambda_1 \lambda_2 \\
 \text{tr}(M) &= \lambda_1 + \lambda_2
 \end{aligned}
 \tag{5}$$

As is a constant set by experience,  $K$  is generally between 0.04-0.06. If a point's  $CRF$  is more threshold  $T$ , is a corner. Harris corner detection operator need to set the parameters  $K$  and  $T$ . The parameter selection arbitrariness will affect the final test results significantly. Algorithm process is as shown in Fig. 5.

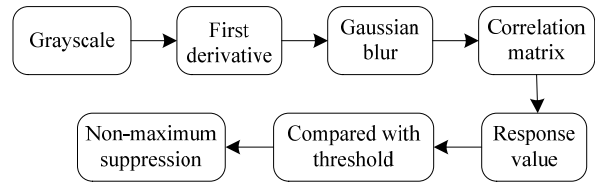


Figure 5. Processing steps for Harris corner detector

#### C. Defects of classic Harris for vehicle model recognition

The corner response value of  $R$ ,  $R$  is more than a fixed threshold value  $T$ . It is considered that point is the corner point, otherwise it is not. Fixed threshold will have some problems in the model recognition.

1. The differences of corner points' counts between different vehicle model lead to mismatch. As shown in Fig. 6, vehicle A has little corner points, vehicle B has many corner points, A and C with the same models, A and B with the different models. The counts of corner points matched between A and B successfully more than those between A and C in this case, so we concluded that A and B have the same model, vehicle model recognition for A is failure.

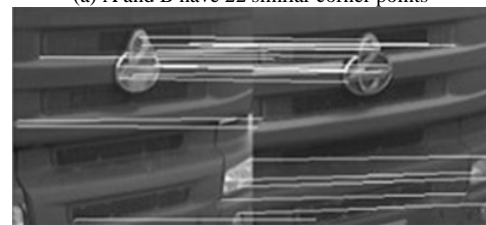
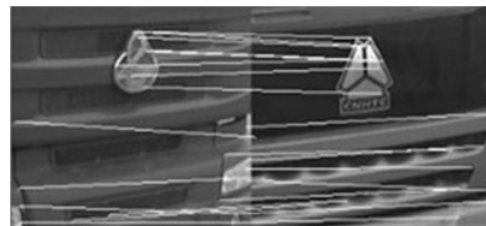


Figure 6. Differences of corner points between different vehicle model lead to mismatch

2. The differences of corner points' counts in different illumination condition lead to mismatch. As shown in Fig. 7, vehicle D and F have the same models in different illumination condition, vehicle D and E have different models in well-lighting condition. Because Harris corner detector's adaptive character toward lighting is not well, the corner points' count in well-lighting condition is large more than that in poor lighting condition. The counts of corner points matched between D and E is successfully more than those between D and F in this case. So we concluded that D and E have the same model, vehicle model recognition for D is failure.

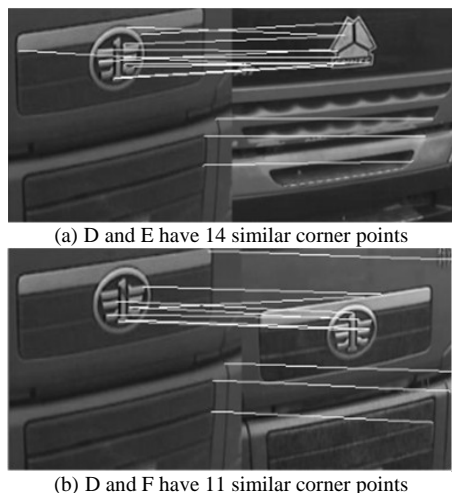


Figure 7. The differences of corner points in different illumination condition model lead to mismatch

As shown in Fig.6 and 7, the classic Harris corner detector algorithm for vehicle model recognition effect is not good. The key reason is that the number of corner points is difficult to control. There is a large gap of the number of corner points between detected in the different models or in different environments, so it often leads a mismatch.

*D. Adaptive Harris Algorithm*

Corner points' count caused by the fixed threshold number of inconsistencies affect the recognition accuracy of the models. Corner threshold should be related to the overall distribution of the response value and the change range. So we take the mean of responses and the variation range weighted sum as a final adaptive threshold, Specific (6):

$$\begin{aligned}
 \text{threshold} &= \frac{1}{w \cdot h} (\sum_{x=1}^w \sum_{y=1}^h f(x, y)) + \delta * [\max(f(x, y)) - \min(f(x, y))] \\
 f(x, y) &= \begin{cases} r(x, y) & r(x, y) < t \\ 0 & r(x, y) \geq t \end{cases} \\
 r(x, y) &= \frac{\Delta(M)}{tr(M)^2} \tag{6}
 \end{aligned}$$

Where  $r(x, y)$  represents coordinate  $(x, y)$ 's response values.  $t$  presets a small empirical threshold.

The adaptive threshold is decided by mean value and change range of candidate corner, which reflects variation in the intensity of the image as a whole. If a certain vehicle model has single texture detail or in poor lighting conditions, the gradient of the image is small, the relatively response value is low, the adaptive threshold is also reduced, and still be able to ensure that the angle of the output of a certain number of points. In contrast, the adaptive threshold is also high, and it can limited excessive corner.

As shown in Fig.8, figure corner points' count is 60 points. Fig. 8(b) is corner-image using fixed threshold and Fig.8(c) is corner-image using (3). We can see that adaptive Harris threshold has a higher accuracy. The adaptive threshold can avoid the setting of the threshold value, ensure the detection accuracy at the same time.

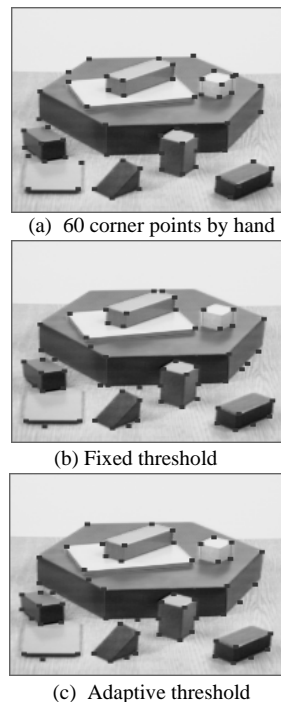


Figure 8. Comparison-image using fixed threshold and adaptive threshold

After the calculation of the (6), corner points' count in different models tends to be consistent in different environment, but not entirely. Sorting corner's response value is executed in order to improve accuracy:

1. Sorting corner by their response value. Strong-corner front, weak-corner behind.
2. Setting a fixed parameter  $n$ , it indicates the desired output number of corner points. It is usually less than the expected minimum number of corner.

Adaptive Harris algorithm process is shown in Fig. 9.

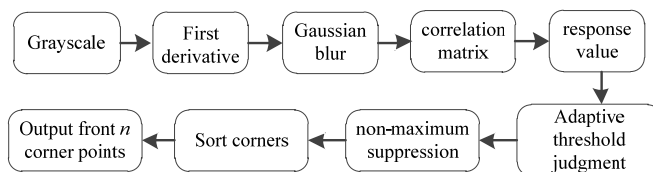


Figure 9. Processing steps of Adaptive Harris

Adaptive Harris algorithm avoids models and differences between model or environment affect recognition accuracy. As shown in Fig. 10 ( the same vehicles as Fig. 6 ), vehicle A, B and C all have 15 corner, vehicle A and B have 5 corner points and A and C have 13 corner points, we conclude A with C are more similar than A with B.

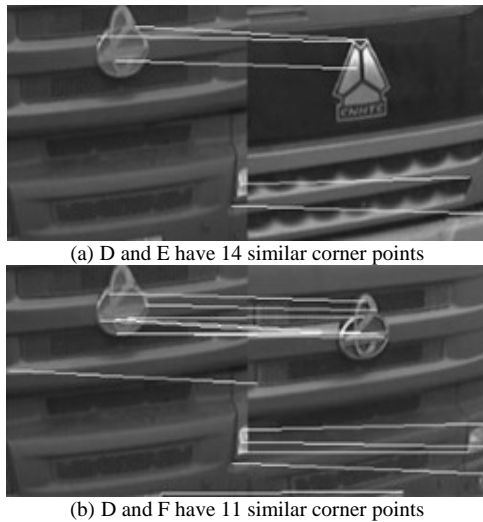


Figure 10. Model recognition for vehicle D

Adaptive threshold can be guaranteed a certain amount of corner points to some extent by setting  $t$  and  $\delta$ . The number of corner points impacts on the recognition accuracy of vehicle model. Few feature-corner generates a false match easily, more feature-corner increases amount of calculation. The perfect quantity is sufficient to reflect the characteristics of the models, but also to avoid the “false corner”. Fig. 11 is statistical analysis about vehicle model accuracy under different number of corner points detected by Harris, SURF and SIFT algorithm respectively. The results show that when the number of corner points in the range of 100-150 ( $t = 15, \delta = 0.05$ ) the recognition accuracy is relatively stable. When the corner points to 130, the Harris algorithm models achieve the highest recognition accuracy rate up to 99%.

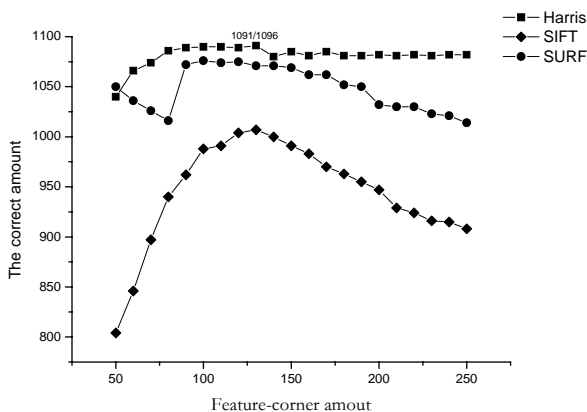


Figure 11. Effect of the number of corner points on vehicle recognition accuracy

IV. GPU/CPU COLLABORATIVE COMPUTING

The proposed method for vehicle model recognition includes ROI positioning, corner detection and corner matching. The corner detection uses adaptive Harris algorithm, and corner matching uses the maximum correlation algorithm (Max-correlation).

As shown in Fig. 12, using database matching to calculate the similarity of models' image and the advantage is that the method is simple, maintain and upgrade easily. Its disadvantage is that the calculation is extraordinarily time consumption in the case of big database and can't meet the real-time requirements.

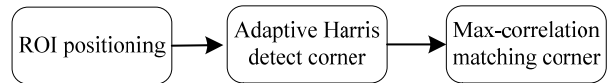


Figure 12. Processes of vehicle model recognition algorithm

A. Parallelization of algorithm

Parallelization of algorithm includes parallel adaptive Harris algorithm and Max-correlation algorithm. These two algorithms are both compute-intensive, very suitable for parallel acceleration. Table I analysis of the processing flow of these two algorithms carefully, irrelevant loop and matrix operations are transmitted to GPU processing, transmission of signals and logic processing by CPU.

Sample images are compressed to the size of 200 × 200 in this paper. Every calculation of corner detection and corner matching volume is not too large. In addition, the communication between the GPU and CPU overhead is large, and the speedup of parallel algorithm is not very high.

TABLE I. PARALLELIZATION OF ADAPTIVE HARRIS

Algorithm	Step	Parallelizable (Y/N)
Adaptive Harris	1.Grayscale	Y
	2.Gradient	Y
	3.Gaussian blur	Y
	4 Correlation matrix	Y
	5.Response	Y
	6.Non-maximum suppression	Y
	7.Adaptive threshold	N
Max-correlation	8. Calculating a correlation matrix	Y
	9.Search match corner	Y
	10. Calculating number of match corner	Y

B. Parallelization of workflow

For the entire vehicle recognition method, due to corner detection in the whole process is called only once, the contribution of the acceleration to the entire model recognition is limited. Max-correlation algorithm is called every matching process, but the amount of calculation of the algorithm itself is small. So it is not the bottleneck of the whole method efficiency. Most of the recognition time consuming in the processing of data transmission, acceleration efficiency is not very high.

Since every corner matching process is independent, the matching of the different vehicle models can be parallelized. We select NVIDIA CUDA platform, used optimization techniques are as follows:

1. Loading all features of datasets first, some constant as Gauss coefficients were loaded to constant registers of GPU.
2. Setting the number of feature database as thread blocks, setting the number of corner as the number of threads in a thread block.

Parallelization of workflow is realized in Fig. 13.

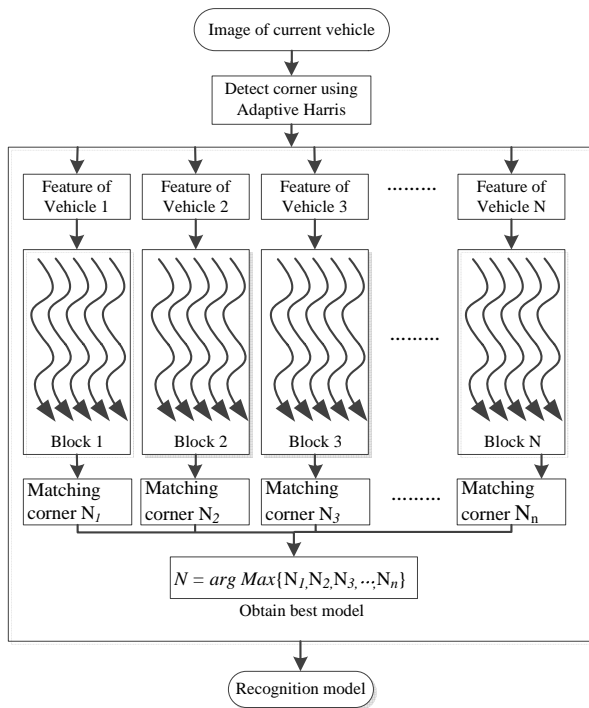


Figure 13. Frame of parallel model for vehicle model recognition

V. EXPERIMENT RESULTS AND ANALYSIS

A. Data set description and experimental environment

The datasets come from "JT-G green channel inspection system" on Guangxi "Quan-Yellow" high-speed toll station. Datasets include 6 categories logo, 12 vehicle models and 1096 front and side images with the speed of 3-20 km/h speed passing the equipment. Affected by the sensitivity of the sensor and the vehicle's uniform speed, vehicle distance and angle may have a slight deviation. The samples also contain different lighting conditions such as sunny, cloudy, and evening. Experimental environment as following:

1. PC: Intel Core i5-2400@3.10GHz x4, 4GB memory, win7 sp1 x64 operating system.
2. GPU: NVIDIA Tesla C2075, CUDA v4.2.

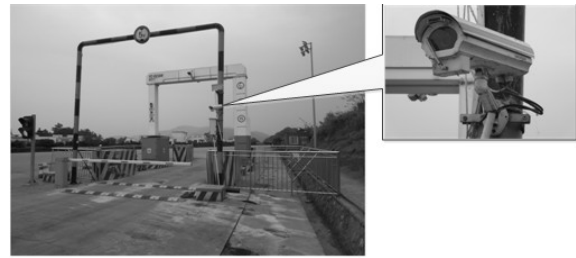


Figure 14. Green channel inspection system on the freeway

B. Recognition Accuracy

The classic image "Lena" for corner detection is selected in order to verify the accuracy of the adaptive Harris algorithm based on GPU/CPU collaborative computing. The comparative results are shown in Fig. 15.

We can see that the two corner-image are almost the same, which proves the accuracy of the algorithm. It is noteworthy that, these two corner-images are not same completely. That is because the inconsistencies of the floating calculation accuracy between CPU and GPU. It leads to the result of the calculation having a small deviation.



(a) corner-image of CPU-Version algorithm



(b) corner-image of GPU/CPU-Version algorithm

Figure 15. Effect comparison of CPU-Version and GPU/CPU-version of the algorithm

Vehicle model recognition has many feature-extraction methods, this paper compared GLCM, PHash, DCT, SURF, SIFT and Harris, the results are shown in Table II.

TABLE II.  
8 FEATURE-EXTRACTION METHODS FOR VEHICLE MODEL RECOGNITION

Feature-extraction methods	Accuracy/%
GLCM	62.3
PHash	80.0
DCT	89.9
SURF	78.0
SIFT	52.0
Harris	95.1
SURF(adaptive)	92.1
SIFT(adaptive)	98.1
Harris(adaptive)	99.5

Adaptive Harris achieves 99% accuracy. SIFT, SURF and Harris corner detector with adaptive threshold all achieve a high accuracy rate, the accuracy of the adaptive corner of Harris algorithm is the highest.

C. Recognition Time

Speedup relates to the algorithm’s computational complexity, the amount of data, IO Frequency, GPU hardware performance and software version. Table III is the acceleration effect of the parallel Harris algorithm under different image sizes. It can be seen that the greater the amount of data processed, the higher the acceleration efficiency.

TABLE III.  
SPEEDUP OF PARALLEL ALGORITHM

Size(x <sup>2</sup> )	CPU(ms)	GPU(ms)	Speedup
100 <sup>2</sup>	9.3	1.5	6.2
200 <sup>2</sup>	31.4	2.9	10.8
300 <sup>2</sup>	54.4	5.4	10.1
400 <sup>2</sup>	80.2	7.9	10.2
500 <sup>2</sup>	118.9	9.8	12.1
600 <sup>2</sup>	184.9	15.0	12.3
700 <sup>2</sup>	251.9	20.0	12.6
800 <sup>2</sup>	332.8	27.1	12.2
900 <sup>2</sup>	398.4	30.0	13.3
1000 <sup>2</sup>	498.2	38.0	13.1

Table IV is a parallel accelerated situation for the entire vehicle model recognition method. Because the amount of data processed is small, speedup of method only based on parallel acceleration of adaptive Harris algorithm and Max-correlation is not high. Their acceleration contribution to the entire model identification method is limited. The task parallel acceleration effect is the most obvious, and the speedup of the entire vehicle recognition is up to 58x.

TABLE IV.  
THE SPEEDUP OF VEHICLE MODEL RECOGNITION BASED ON GPU/CPU COLLABORATIVE COMPUTING

Parallel content	CPU/ms	CPU+GPU/ms	Speedup
Adaptive Harris	31.1	2.9	10.7
Max-correlation	4.0	2.6	1.5
The entire method	4071.5	70.1	58.1

VI. CONCLUSION

A high-accuracy method for vehicle model recognition and its GPU/CPU collaborative computing is presented in the paper, and this recognition method has a high recognition accuracy and good robustness. GPU/CPU collaborative computing is used to accelerate method, which satisfies the real-time requirements.

ACKNOWLEDGMENTS

This work has been supported by the National Natural Science Foundation of China under contract Nos. 61105004 and 61172053, the Natural Science Foundation of Guangxi province of China under contract No. 2012GXNSFAA053230, Program for Excellent Talents in Guangxi Higher Education Institutions([2011]40), Innovation Project of Guangxi Graduate Education under contract Nos. YCSZ2012073 and 2011105950811M24.

REFERENCES

- [1] H. Y. Du, “Implementation of BP Algorithm Using Matlab Based on Vehicle Type Recognition,” *Computer and Modernization*, vol.5, pp. 20–22, October 2012.
- [2] W. X. Kang, X. T. Cao, Z. Sheng, et al, “Harris Corner and SIFT Feature of Vehicle and Type Recognition,” *Journal of harbin university of science and technology*, vol. 17, pp. 69–53, September 2012.
- [3] V. Petrovic, T. Cootes, “Analysis of features for rigid structure vehicle type recognition,” *British Machine Vision Conference*, vol. 2, pp. 587–596, 2004.
- [4] H. Maemoto, S. Okuma, Y. Yano, “Parametric vehicle recognition using knowledge acquisition system,” *IEEE International Conference on Systems, Man & Cybernetic*, vol. 4, pp. 3982–3987, 2004.
- [5] J. Ma, J. J. He, L. SHI, “Feature Extraction of Car Local Aera and Vehicle Recognition,” *Industrial control computer*, vol. 8, pp. 72–74, October 2012.
- [6] H. Lee, “Neural network approach to identify model of vehicles,” *Advances in Neural Networks-ISNN*, pp. 66–72, 2006.
- [7] A. Psyllos, Anagnostopoulos C. N, E. Kayafas, “Vehicle model recognition from frontal view image measurements,” *Computer Standards & Interfaces*, vol. 33, pp. 142–151, February 2011.
- [8] C. Harris and M. J. Satephens. “A combined corner and edge detector”, *Alvey Vision Conference, Manchester*, pp. 147–152, 1986.
- [9] F. S. Lu, J. Q. Song, F. K. Yin, et al, “Survey of CPU/GPU Synergetic Parallel Computing”, *Computer Science*, vol. 38, pp. 5–9, 2011.
- [10] C. Shuai, M Boyer, Y. M. Jia, et al, “A performance study of general-purpose applications on graphics processors using CUDA” , *Journal of parallel and distributed computing*, vol. 68, pp. 1370–1380, October

2008.

- [11] Y. C. Gui, G. J. Zhang, “An improved implementation of Preconditioned Conjugate Gradient method on GPU” , *Journal of software*, vol.7, pp. 2695–2702, December 2012.
- [12] H. F. Wang, Q. K. Chen, “Power estimating model and analysis of general programming on GPU” , *Journal of software*, vol.7, pp. 1164–1170, May 2012.



**Huihua Yang**, Ph.D., professor of Guilin University of Electronic Technology (GUET), adjunct professor of Beijing University of Posts and Telecommunications, deputy secretary general of the Chinese Council of Near Infrared Spectrum, ACM and CCF member. He received his Bachelor degree (1994) and Master degree (1997) at

Huazhong University of Science and Technology in 1994 and 1997, respectively. He received Ph.D. degree at East China University of Science and Technology in 2005. He participated to work in department of computer science of GUET in 1997. He was a postdoctoral at Analysis Center of Tsinghua University from July 2005 to July 2007.

Currently, he leads some major projects, such as the JT-G Green Channel inspection system, the algorithm software for crude oil fast evaluation funded by RIPP of Sinopec, a Guangxi Natural Science Fund on NIR Internet of Things, etc. He has completed several projects including one funded by National Science Foundation of China on modeling of NIR spectrum.

He has published more than 30 papers in journals or conferences at home and abroad. His main research directions include intelligent information processing, machine learning, data mining, cloud computing, near-infrared spectroscopy and its application.

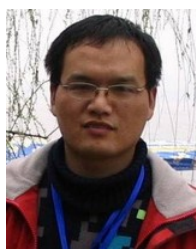


**Lei Zhai**, master (2013) degrees in Guilin University of Electronic Technology.

He joined “JT-G Green Channel inspection system” as a main researcher, studied vehicle model recognition based on image processing. He hosts two Innovation Projects of Guangxi Graduate Education (YCSZ2012073 and

2011105950811M24).

His main research direction is image processing, computer vision, intelligent transportation, high-performance computing etc.



**Zhenbing Liu** received the Ph.D. (2010) degrees in control science and engineering from Huazhong University of Science and Technology, China.

He joined the School of Engineering and Automation at Guilin University of Electronic Technology as an assistant professor in 2010, and is currently vice professor. His research interests include pattern recognition, machine learning, and image processing.

He has published over 10 refereed papers in these areas, including those published in prestigious journals and top international conferences. He is also a frequent reviewer for various international journals and conferences.