

A Method of Recognition for Fake Plate Vehicles

Dongyu Guo^{a)}

School of Automation Wuhan University of Technology Wuhan, China.

^{a)} guodongyu1209@163.com

Abstract. The investigation of fake plate vehicles mainly relies on the traffic police verifications and public tip-offs. It is inefficient and produces very little effect. Based on above analysis, a method of recognition for fake plate vehicles is proposed which to investigate the fake plate vehicles effectively, Increase the intensity of the fight against crime. In this paper, the saliency areas of training set are detected to remove the partial background of vehicle image. SIFT descriptors are translated as "visual word" and the inputs of SVM methods for training and recognizing vehicles. The experiments show that methods in this paper can detect fake plate vehicles efficiently, and the detection precision is 84%.

Key words: vehicle recognition; vehicle retrieval; saliency areas; SVM; ACS; smart phone.

INTRODUCTION

Recently, with the development of city construction and automobile industry, the number of vehicles grows, and the illegal activities caused by fake plate vehicles increases. Fake plate vehicles violate the interests of the legitimate vehicle owners, provide tools for lawbreakers, and bring difficulties to police to catch the hit-and-run vehicles, cause losses to state taxes and highway maintenance fees.

However, the investigation of fake plate vehicles mainly relies on the traffic police verifications and public tip-offs. It is inefficient and produces very little effect. Based on above analysis, a method of recognition and retrieval for fake plate vehicles is proposed in this paper which to investigate the fake plate vehicles effectively, Increase the intensity of the fight against crime.

Most of the remote vehicle retrieval and recognition systems are implemented through the recognition of vehicle license plate recently. Mutholib et al. used a neural network based optical recognition (OCR) method to extract texts and numbers of license plate [1]. Ahmad et al. developed a mobile LPR (License Plate Recognition) system for Android Operating System which used tesseract-ocr for Optical Character Recognition [2]. Aishwarya Visvanathan et al. designed a vehicle retrieval system, by taking photos of the license plates with mobile phones and identifying the license plate number to search corresponding vehicles from the database [3].

However, the vehicle retrieval and recognition systems which are based on license plate recognition have certain limitations. The method requires the user to provide images containing the license plate of the vehicle so that the users can only shoot front or rear view of the vehicle. Since the collection of images has been restricted, it will not be able to take full advantages of the smart phones' flexibility in real-time acquisition of the images. Moreover, some vehicle owners may use fake license plates or obscured license plates to make it impossible to effectively identify the vehicle license plates.

In order to make up for the limitation and deficiency of the license plate recognition method, this paper has proposed a vehicle recognition technology based on the overall appearance characteristics of the vehicles. This method can be considered as an auxiliary to the traditional license plate recognition method to solve the problem of missing or wrong license plate.

This paper is structured as follows: section II introduces the training set preprocess. Section III shows the generation of visual dictionary. The process of vehicle retrieval is introduced in the section IV. Section V presents

the vehicle type recognition. Finally, Section VI shows the performance of testing the system, summary of this article and directions for future research.

IMAGE PREPROCESSING

Image Saliency Detection

Image saliency detection is a technique based on imitating human visual observation and ROI (Region of Interest) extraction. It is a comprehensive process of image analysis, feature extraction and exploration of human visual characteristics [4]. Saliency detection has great significance for various applications based on image analysis and processing.

In this paper, the saliency areas of training set were extracted in order to remove the partial background of vehicle image and to highlight the vehicle itself. During the feature extracting, by filtering out a lot of background characteristics, we can reduce computational complexity, improve the system's speed and get good training results.

This paper used the method proposed by Federico Perazzi et al. [5] to extract the saliency areas of the vehicle's images. First, a given image was decomposed into compact, perceptually homogeneous elements that abstract unnecessary details. Then compute two measures of the contrast that rate the color uniqueness and the spatial distribution of these elements. A saliency measure was derived from the element contrast. Finally, assign the actual saliency values to the input image to get a pixel-accurate saliency map [5].

Background Filtering

By extracting the saliency areas, the background and foreground of the image can be simply distinguished. The brightness of the saliency areas is significantly greater than that of the background. By setting the pixel value of the saliency areas to one and the background to zero, we can form an image matrix which is consisted of zeros and ones. Then, apply dilation operation to the matrix. In the following step, each pixel value of the matrix is multiplied by the corresponding pixel value of the original input image to obtain the preprocessed image.

After the preprocessing, some of the background of the training images are eliminated to retain a complete body of the vehicle. This can avoid extracting a lot of background-characteristics during the feature extraction and contribute to improve recognition and retrieval precision.

VISUAL WORDS TRANSLATION

The “bag of words” model is a popular method in the field of image processing. Analogy with an article made up by a lot of textual words, if the image is expressed as a combination of “visual words”, the techniques of text retrieval can be used in image retrieval. Moreover, the image's features were quantized into “visual words” also helps to improve the efficiency in large-scale image retrieval system [6]. The steps of generating “visual words” are as follows [7]:

1. Detect local features of the images;
2. Describe the feature as a vector;
3. Cluster all the vectors that are produced from the image database to generate a “visual dictionary”;
4. Map each vector of the image to a “visual word” according to the “visual dictionary”;
5. Count the frequency of occurrence of each “visual word” in the relevant image to generate a “code book” for the image.

Feature Detection and Description

In recent years, people have developed a variety of multi-scale local feature detection algorithms to obtain stable feature points or areas in the images, such as Harris [8], DOG [9] (Difference of Gaussian), and MSER [10] (Maximally Stable Extremal Regions) methods.

For this system, as a result of using mobile phones, the resolution of the images is relatively low. In addition, stronger robustness for viewpoint changing is necessary in this system because the analysis and processing of the images is based on the characteristics of the vehicle's overall appearance.

SIFT [9] method works well than SURF under the condition of multi-scales and rotation, while the speed of SURF is faster [11]. For the image matching based on similarity, SIFT performs well, while for the image matching based on affine distortion or large viewpoint changing, SIFT has worse performance.

Even though the ASIFT [12] method has great robustness in viewpoint changing, its computational complexity, high requirements of CPU and memory, and the slow processing speed make it not applicable for vehicle recognition and retrieval system based on embedded devices.

This paper has adopted the Hessian-Affine detector which is robust in scale, rotation, illumination and viewpoint changing to detect features. The Hessian-Affine detector uses multi-scale iterative algorithm to determine the key point invariant in scale and affine distortion. In each scale, the method based on Hessian matrix is used to select interesting points.

Compared to SIFT, the Hessian-Affine detector is not as sensitive to the affine transformation and is more robust in viewpoint changing [13]. Although the number of features detected by Hessian-Affine is less, the feature matching is more accurate.

After detecting the key points by the Hessian-Affine method, we need to describe the key points' features. A vector which is used to describe the corresponding key point is generated by calculating the gradient histograms of the neighbor pixels' values surrounding the key point [9]. This vector is the information to represent the area around the key point. This paper uses the classical SIFT descriptor to describe the key points. Calculating the gradient information of 8 directions over 4×4 subregions surrounding the key point, then, a $4 \times 4 \times 8 = 128$ -dimension vector is generated to represent the key point.

“Visual Dictionary” and “Code Book”

When the features of all the images in the database have been extracted and changed into 128-dimension vectors, by clustering all these vectors, K centers can be formed. Each center and the vectors belonged to the center can be seen as a “visual word”. Each “visual word” represents a group of similar features and K “visual words” form the “visual dictionary”.

Since each image can be expressed as a set of vectors, calculating the distance between each vector and each “visual word” in the “visual dictionary”, and each vector can be mapped to its nearest word. By counting the frequency of the occurrence of each “visual word” in each image, the image can be compiled into a K -dimensional “code book”.

VEHICLE RETRIEVAL

Vehicle retrieval means that, the user uploads the vehicle image taken by phone to the server, then the server searches for similar images from the database and sends the results which are arranged according to the similarity of descending order to the phone terminal.

The Establishment of “Code Book” Index Matrix

The “code book” index matrix should be established in order to facilitate the similarity calculation between the input image and the images in the database. Because the vocabulary of “visual dictionary” for this system is 5000 and the total number of images in the database is 1000, the size of the index matrix is 5000×1000 . The 5000 rows of the index matrix represent 5000 “visual words”, while the 1000 columns represent 1000 images in the database. The matrix element means the j th image possesses the i th “visual word” and the number of the word is $a_{i,j}$. By building the index matrix, people can conveniently inquire every image respectively includes which words and the number of each word, it brings great convenience to computation for this system.

The preliminary “code book” index matrix is probably a statistic of term frequency for each “visual word” owned by each image. However, the word has high frequency may not be a meaningful word, and the image features presented by this word may not be the significant feature. As a result, this paper uses the “IDF” weighting method to assign a weight to each “visual word” in order to reflect the “importance” of the word.

The index matrix that is weighted by the “IDF” method can highlight the major features of the images and ignore the nonsignificant features, such that the images can be described more appropriately. The “IDF” calculation formula is as:

$$idf_i = \log \frac{|D|}{|\{j : t_i \in d_j\}|} \quad (1)$$

where the numerator is the number of images in the database (in this paper, it is 1000) and the denominator is the number of images which include the “visual word”. The result is the “IDF” value of the given word.

After calculating the “IDF” value for each “visual word” in the “visual dictionary”, people need to build a diagonal matrix with the “IDF” values and multiply the diagonal matrix with the preliminary index matrix to get the weighted index matrix. In order to facilitate the calculation in the future, the weighted index matrix should be normalized to form the final index matrix. Each column vector of the final index matrix describes the weight of “visual words” of the corresponding image.

Vehicle Retrieval

The user takes a picture of a vehicle and uploads the image to the server. The server translates the image into a “code book” (a 5000×1 matrix). The “code book” is multiplied with the diagonal matrix mentioned above to get the “IDF” weighted “code book” which is finally normalized for convenient computation.

The “code book” of the input image is in fact a 5000-dimension column vector. Compute the distance of the “code book” that is generated from the input image with each column vector of the index matrix. The distance of two vectors represents the similarity of the corresponding two images. The smaller the distance, the more similar the two images.

In general, to calculate the cosine distance to compute the similarity between two vectors, the cosine similarity calculation formula is:

$$\cos \theta = \frac{A \bullet B}{|A| \times |B|} \quad (2)$$

A and B are n-dimensional vectors. The numerator is the inner product of A and B. $|A|$ and $|B|$ are the modules of A and B respectively. The denominator is a product of $|A|$ and $|B|$. The closer the result of formula is to 1, the closer the angle between the two directions of the vectors is to 0 degree, that is, the two vectors are more similar.

However, the cosine similarity only reflects the relationship of the vectors’ directions, it does not reflect the connection between the vectors’ values. Cosine similarity is not sensitive to the numerical, which results in the error of the result. Therefore, this paper uses ACS (Adjusted Cosine Similarity) to compute the similarity between the vectors. The calculation formula of the ACS is:

$$d_{s,t} = \frac{(r_s - \bar{r}_s)(r_t - \bar{r}_t)}{\sqrt{(r_s - \bar{r}_s)^2} \sqrt{(r_t - \bar{r}_t)^2}} \quad (3)$$

where r_s and r_t represent the two vectors that are used to compute the similarity, \bar{r}_s and \bar{r}_t are the mean values of r_s and r_t respectively. The result $d_{s,t}$ is the ACS of the two vectors. The smaller the value of $d_{s,t}$ is, the higher the similarity of the two images. The retrieval result is the list of images arranged according to the ACS by ascending order.

VEHICLE RECOGNITION

Training the SVM Classifier

In this paper, a total of 1000 vehicle images (50 common types, 20 images for each type) were collected from the website <http://club.xcar.com.cn/> as the database. Both the vehicle training and retrieval were based on this database.

In order to adapt to arbitrary angle of picture captured by the smart phone, there are at least 8 different shooting directions corresponding to different view angles for each type of vehicle.

First, extract saliency areas for all images in the database. Second, detect and describe the features for the saliency filtered images. Third, cluster the feature vectors to generate the “visual dictionary”. Fourth, quantify each feature as a “visual word” according to the “visual dictionary”. Finally, collect statistics of occurrence frequency for each “visual word” to translate the image into a “code book”.

To make manual annotation for each image, 1000 images can be classified into 10 categories by labeling the corresponding type for each vehicle. Then, collect the “code books” of the same kind of vehicle type together as a set and train the set by SVM method to learn a classifier, each classifier for each vehicle type can be obtained by this method.

Vehicle Recognition

The process of vehicle recognition is shown in Fig. 1 and Fig. 2.

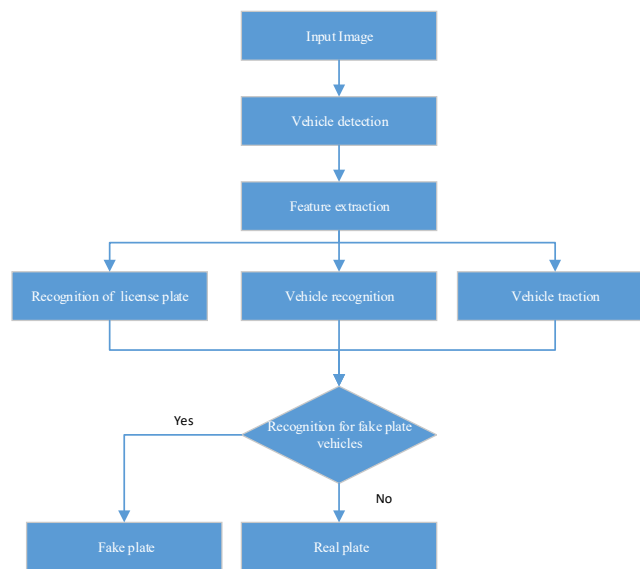


FIGURE 1. The process of vehicle recognition

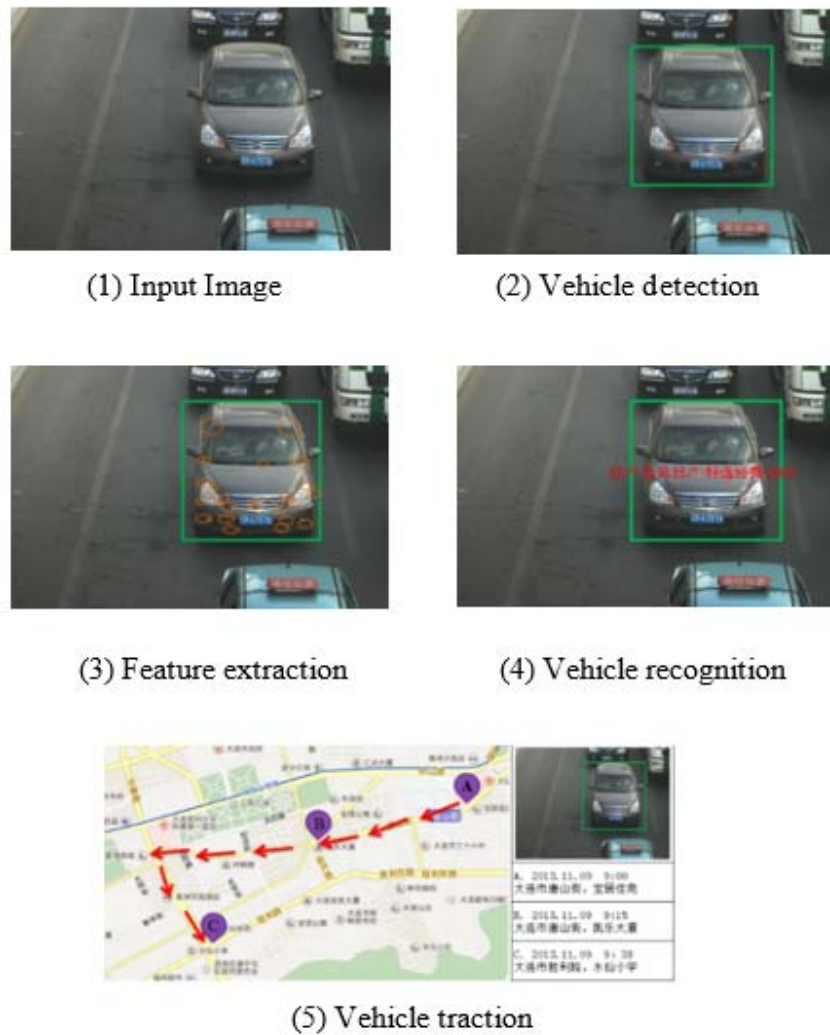


FIGURE 2. vehicle recognition and traction

EXPERIMENTS

To illustrate the effectiveness of the method of recognition for fake plate vehicles, two sets of experiments of same types and different types are conducted.

200 normal vehicle images and 100 fake plate vehicle images which include the license plate occlusion, license plate defilement, no license plate, license plate forgery is recognized. The experimental results are shown in Table 1. The recognition precision for fake plate vehicle is 84%.

TAB. 1. Recognition of fake plate vehicle

Class	Image Number	correct number	Accuracy
Fake plate vehicles	100	84	84%
Normal vehicles	200	196	98%

Recognition of Fake Plate Vehicles Between Different Vehicle Types

The so-called fake plate vehicles of different types refer to vehicles that have different types but use the same license plate number. As shown in Figure 3, the “Toyoto” car has used the license plate of the “Benz” car with the license plate number “Su L00V88”.



FIGURE 3. Example of fake plate vehicles between different vehicle types

Recognition of Fake Plate Vehicles Between Same Vehicle Types



FIGURE 4. Example of fake plate vehicles between same vehicle types

The fake plate vehicles of same type refer to different vehicles that have same type, the appearance is consistent and using the same license plate. For the same type of fake plate vehicle, the results cannot be detected by vehicle identification alone. This article uses the vehicle search to detect fake plate vehicle of the same type. First of all, extract the foreground area from the vehicle image- vehicle face and the part of window above it. Extract the foreground area can filter out a lot of background information in the image. At the same time, the area of window

contains many details of the car, such as decorations, various notes attached, etc. The detailed information is used as reference information in the image retrieval process, which help to improve the retrieval accuracy. And the use of window details plays an important role in the retrieval of the same vehicle with different license plates.

License plate is recognized on vehicle images in Fig. 4, and the result of recognition is compared to find out whether there is duplicate license plate number. If it is true, the car is fake license plate vehicle; conversely, the car is the real license vehicle. Fig. 4 shows that the two vehicle license plates marked with red circles are "Jin BHG 368", but decorations are different. This indicates that the input vehicle may be a fake plate vehicle.

CONCLUSION

The experimental results show that using the method in this paper can detect the vehicle face and foreground images of unlicensed vehicles, license plate covered vehicles, license plate defaced and forged plate vehicles. The recognition precision is 84%. The research of this paper solved problems of foreground extraction and fake plate vehicle recognition, lays a strong foundation for the future development of fake plate vehicle recognition and retrieval system.

REFERENCES

1. A. Mutholib, T S. Gunawan and M. Kartiwi, "Design and implementation of automatic number plate recognition on android platform," In Computer and Communication Engineering (ICCCE), IEEE, pp. 540–543, 2012.
2. A G.Liaqat, "Mobile real-time license plate recognition," Ph.D. dissertation, Goteborg: Linnaeus University, 2011.
3. A. Visvanathan and T. Chattopadhyay "Car number plate recognition for smart building management," In Digital Signal Processing, IEEE, pp.73--77, 2014.
4. S. He, J. Han, X. Hu, M. Xu, L. Guo, and T. Liu, "A biologically inspired computational model for image saliency detection," In Proceedings of the 19th ACM international conference on Multimedia, ACM, pp. 1465-1468, 2011.
5. F. Perazzi, P. Kr01henbühl, Y. Pritch, and A. Hornung, "Saliency filters: Contrast based filtering for salient region detection," In Computer Vision and Pattern Recognition (CVPR), IEEE, pp. 733–740, 2012.
6. W. Pei, Z J. An, Y Y. Zhu, X. Jia, X W. Zuo, F S. Wangl, "A rapid vehicle recognition and retrieval system," In Systems and Informatics (ICSAI), IEEE, pp. 748-753, 2014.
7. G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray, "Visual categorization with bags of keypoints," In Workshop on statistical learning in computer vision, ECCV, pp. 1-2, 2004.
8. C. G. Harris, and M. Stephens. "A combined corner and edge detector," Proc of Fourth Alvey Vision Conference, pp. 147-151, 1988.
9. J. Matas, et al. "Robust wide-baseline stereo from maximally stable extremal regions. Image Vis Comput," Image & Vision Computing, pp. 761-767, 2004.
10. D G. Lowe, "Distinctive image features from scale-invariant keypoints," International journal of computer vision, vol. 60, no. 2, pp. 91-110, 2004.
11. L. Juan, O. Gwun, "A comparison of sift, pca-sift and surf," International Journal of Image Processing, vol. 3, no. 4, pp. 143-152, 2009.
12. J M. Morel, G. Yu, "ASIFT: A new framework for fully affine invariant image comparison," SIAM Journal on Imaging Sciences, vol. 2, no. 2, pp. 438-469, 2009.
13. K. Mikolajczyk, C. Schmid, "An affine invariant interest point detector," Springer Berlin Heidelberg, pp. 128-142, 2002.