Vehicle Type Classification based on Improved HOG_SVM

Penghua Ge [a], Yanping Hu [b]
School of Mechanical Engineering, DaLian University of Technology, DaLian 116000, China.
[a]gepenghua@mail.dlut.edu.cn, [b]hypok@dlut.edu.cn

Abstract. There are few differences in the characteristics of vehicles and many interference factors in vehicle identification, especially in complex backgrounds. In order to improve the accuracy of image feature extraction and recognition in complex background, a vehicle-types recognition technology based on improved HOG_SVM is proposed in this paper. In order to obtain abundant vehicle identification information, we perform targeted image preprocessing methods such as grayscale stretching and Gaussian filtering on the original image to reduce background interference factors. The HOG feature is then introduced to obtain rich features of the image, and the SVM classifier in machine learning is trained at the output layer by multitasking learning of a large amount of tagged data. Different from the traditional method, the PCA dimension reduction process is used to speed up the recognition of the improved HOG feature, and the method of SVM is used to avoid the classifier from falling into the local minimum. In this paper, the public vehicle dataset is used as the classifier training dataset and test dataset, and the proposed method is verified by experiments.

Keywords: Vehicle type classification; HOG_SVM; feature extraction; PCA.

1. Introduction

With the dramatic increase in the number of motor vehicles in China, new requirements have been put forward for public management and event handling control of motor vehicles. It is often necessary to quickly obtain the relevant attributes of the target vehicle in a large number of surveillance video images: type, color, brand, size, etc., where the vehicle type is the most critical and basic attribute of the motor vehicle. The classification also requires more explicit and precise, but the current type of recognition task for vehicles in images still faces great challenges. Highways and urban roads have formed a complete high-definition camera network system. How to make good use of these video image big data to realize the automatic identification of road vehicles is a very meaningful and challenging research topic.

Vehicle type recognition technology based on computer vision has included digital image processing, computer vision, pattern recognition and other fields. In recent years, computer vision-based vehicle recognition has achieved great progress, and researchers at home and abroad have done a lot of work. Lim T R used Gabor filtering and support vector machine to complete vehicle detection [1]. A method to recognize vehicles using an improved SIFT and multi-view model was proposed by Liqin Hua [2]. In the literature [3], Dong Z et al. proposed a method based on unsupervised learning convolutional neural network for vehicle classification. Abadi proposed a vehicle model recognition based on using image processing and wavelet analysis [4]. Tong Zhang used corner information as feature point in the front image of vehicle to recognize the type which the vehicle belongs to and vehicle type was recognized by the ratio of matching corner [5]. Others, most of these methods are based on vehicle view images. Model based methods [6] compute the vehicle's 3D parameters such as length, width, and height to recover the 3D model of the vehicle. Appearance-based methods [7] extract appearance features (e.g., SIFT [8], Sobel edges) to represent the vehicle for classification.

In short, the main methods of vehicle recognition technology for appearance are neural network, template matching, pattern recognition and support vector machine. These methods have their own inherent defects and cannot meet the recognition speed and accuracy at the same time, which are two of the most important indicators of vehicle classification. Therefore, this paper proposes a vehicle identification technology based on improved HOG and support vector machine and achieves higher classification accuracy of vehicle models. Firstly, by using a large number of vehicle pictures as training dataset, a support vector machine classifier is used for data analysis. Finally, the proposed

method is implemented based on OpenCV, an open source framework, which achieves high accuracy for vehicle recognition in test dataset.

2. Method Flow, Data Source, Dataset Construction and Image Preprocessing

2.1 Method Flow

The steps for applying SVM for classification are as follows: Firstly, the training set and test set of each class are collected, adopt the Gaussian low-pass filter and contrast-stretching transformation for the image preprocessing, then the appropriate image features for classification are selected, the features are extracted from the training set, the SVM classifier is trained to obtain the classification template, and finally the test images are classified by the template.

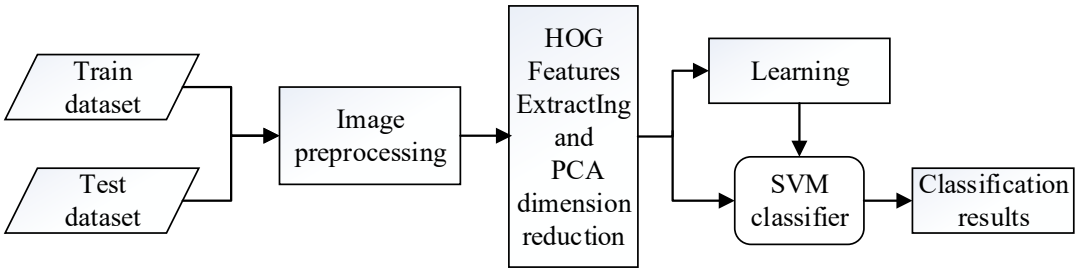

Figure 1. Flow Chart of Vehicle Type Recognition

2.2 Data Source, Dataset Construction

The video images used in the experiments were derived from urban road monitoring. The camera is about 4.5m from the ground and the camera's field of view is about 45 degrees. For camera parameters, the captured video frame rate is 40 frames per second, the video image is 960*540 pixels, and the bit depth is 24. These images are used as test dataset to evaluate the performance of vehicle identification.

The training dataset in this paper uses a public dataset, and the BIT-Vehicle dataset contains 9,850 vehicle images. These images contain various lighting conditions, changing in vehicle surface colors and viewing angles [9]. All vehicles in the dataset are divided into six categories: bus, microbus, minivan, sedan, SUV and truck.

2.3 Image Preprocessing

Due to the influence of image shooting environment and vehicle itself (such as acquisition error, noise, weather and uneven lighting, as well as vehicle manufacturer's logo, car body advertisement, graffiti on the car body, etc.) all cause interference to image, resulting in the image is not necessarily suitable for subsequent processing, so we need to perform a series of algorithmic processing on the captured initial image to highlight the part of the image what we are interested in in order to locate the vehicle accurately. According to the above situations of traffic surveillance videos, the reasonable pre-processing scheme adopted in this paper mainly includes the following items:

1)Image denoising. The noise in the video captured by the camera is mainly salt and pepper noise and random noise, Gaussian filtering has a great removal effect on these two kinds of noises, and does not make the outline of the vehicle blurred, so this article uses Gaussian Filtering for Image denoising. As the Figure 2 shows: (a) to (b).

2)Histogram equalization method can transform the pixels concentrated in a certain gray level into uniform distribution in the whole gray level range, so as to improve the contrast of the image.

3)Gray-scale stretching can make the critical edges of the interesting region more obvious in the captured images, so that the vehicle's attribute information (positions, colors, contours) can be more clearly expressed. As the Figure 2 shows: (b) to (c).

(a) (b) (c)

Figure 2. Image Denoising (a), Histogram Equalization (b), Grayscale Stretching (c)

3. Image Feature Extraction and PCA Dimensionality Reduction

3.1 Vehicle Feature Fxtraction and Optimization

Vehicle feature extraction is the key link of vehicle type recognition. The result of vehicle type recognition is directly related to the selection of vehicle features. Therefore, before the vehicle model is recognized, it is necessary to extract vehicle features as the basis of recognition. In order to ensure that the extracted vehicle features can describe the vehicle category information more completely with less data, this paper proposed the PCA dimension reduction based on the improved HOG feature of the vehicle as the basis for vehicle-type identification.

3.1.1 HOG Feature Extraction Procedures and Algorithmic Principles

The Histogram of Oriented Gradient (HOG) feature is a feature descriptor used for object detection in computer vision and image processing. It constructs features by calculating and statistic the gradient direction histogram of the local area of the image. In an image, the directional density distribution of the gradient or edge can well describe the characteristics of the local target region [10]. HOG uses this idea to make statistics on the gradient information and generate the final feature description. In the process, an image is divided as follows: image -> detection window (win) -> image block (block) -> cell unit (cell).

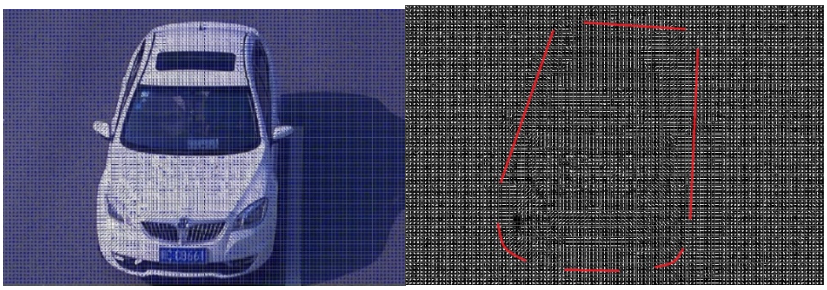

Figure 3. HOG feature map

The extraction process of HOG features is as follows:

Step 1: Image graying: convert the input RGB three-channel image into a single-channel image, and the transformation formula is as follows:

$$Gray = 0.3 * R + 0.59 * G + 0.11 * B \tag{1}$$

Step 2: Gamma correction: Gamma correction is used to normalize the color space of the input image. The correction formula is as follows:

$$I(x, y) = I(x, y)^{\gamma} \tag{2}$$

γ =1/2, Gamma compression processing, can reduce the effects of illumination changes and local image shadows;

Step 3: Calculate the image gradient: including the gradient value and gradient direction of each pixel. The horizontal gradient $G_x(x,y)$ and vertical gradient $G_y(x,y)$ of $I(x,y)$, and the gradient $G(x,y)$ and gradient $\theta(x,y)$ of each pixel are respectively:

$$\begin{aligned} G_x(x,y) &= I(x,y) - I(x-1,y) \\ G_y(x,y) &= I(x,y) - I(x,y-1) \\ G(x,y) &= \sqrt{G_x(x,y)^2 + G_y(x,y)^2} \\ \theta(x,y) &= \arctan\left(\frac{G_y(x,y)}{G_x(x,y)}\right) \end{aligned} \tag{3}$$

Step 4: Construct a 9-dimensional hog eigenvector:

The whole window is divided into cells (8*8 pixels) with the same size and no overlap, calculating the gradient value and direction of each. And then the gradient direction of each cell is divided into 9 bins in the range of 0-180 degrees (non-directional: 0-180, directional: 0-360). The pixels in each cell are weighted by their magnitude and weight the voting for the gradient histogram in which it is located.

Step 5: Combine cell units into large blocks, normalized gradient histograms within blocks:

Individual cell units are combined into large, spatially connected blocks. In this way, the feature vectors of all cells in a block are concatenated to obtain the HOG feature. The normalized block descriptor (vector) is a HOG descriptor. In the experiments, the optimal parameters of vehicle detection are 3*3 cell/block, 8*8 pix/cell and 9 histogram channels. Then the feature of a block is: 3*3*9=36 dimensions;

Step 6: Collect HOG features:

The final step is to collect the HOG features from all overlapping blocks in the detection window and combine them into the final feature vector for classification. After normalizing the block histogram, all the block feature vectors are combined to form 32*32*36=36864-dimensional feature vectors, which is the HOG feature and can be used to represent the entire image. The HOG feature map we got as the Figure 3 shows.

3.1.2 Improved HOG Feature (FHOG)

In this paper, we remove the block based on hog, consider the four neighborhoods of each cell, and normalize and truncate them. The feature is divided into direction-sensitive (360/2=18 dimensions) and non-sensitive (180/2=9 dimensions) types. If four neighborhoods are considered, there are 108(4*9+4*18) dimensions feature. Obviously, the computational complexity is too large and the feature information is too redundant. Like hog feature extraction, the (4+27= 31) dimensional feature can be obtained by accumulating rows and columns of 108-dimensions feature. Compared with the original 36-dimensional feature extraction, not only the dimension is reduced, but also each feature takes into account multiple directions.

3.2 PCA Dimensionality Reduction

In the feature extraction and processing of the previous section, the problem involving high-dimensional feature vectors is often easy to fall into dimensional disasters. In addition, with the dimension increases, the sparseness of the data will become higher and higher, and it is more difficult to explore the sparse dataset in the high-dimensional vector space. Principal Component Analysis (PCA), also known as Karl-Hunin-Lough Transformation, is a technique for exploring high-dimensional data structures [11]. PCA can synthesize potentially correlated high-dimensional variables into linear-independent low-dimensional variables, called principal components.

In this paper, the normalized photo size is 256*256 pixels. Although these pictures are not large, the feature vectors arranged according to the intensity of the pixels in each picture also have 36864 dimensions. We can convert the pixel intensity matrix of a photograph into a vector, and then build a matrix with all the training photograph vectors. Each photograph is a linear combination of the

principal components of the dataset. Assuming there are m points in space, we want to perform lossy compression on these points so that the dimension R^n of the data becomes R^l, which is strictly $l < n$.

The original data is composed of n rows and m columns matrix X. The concrete steps of using PCA algorithm are as follows:

1) Zero-averaging each row of X (representing an attribute field): $x_i = x_i - \frac{1}{m}\sum_{i=1}^{m} x_i$;

2) Calculating the sample covariance matrix: $C = \frac{1}{m} XX^T$;

3) Performing SVD decomposition on the covariance matrix to obtain the eigenvalues of the covariance matrix and the corresponding eigenvectors;

4) The eigenvectors are arranged in rows from top to bottom according to the sort of the corresponding eigenvalues' size, and the first k eigenvectors are taken to form the matrix P;

5) $Y = PX$ is the matrix data after reduction to k dimensions.

In this paper, by improving the HOG feature extraction method and using the principal component analysis (PCA) method to reduce the dimension, reduce the dimension of the eigenvectors, reduce the computational complexity and improve the recognition speed. In the figure 5, we can see that the feature-reflection image after the dimensional reduction of the PCA method still reflects the features of original image very well.

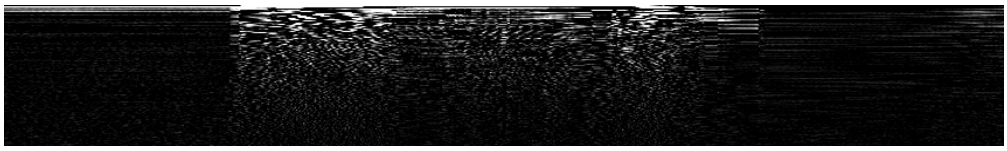

Figure 4. Feature map after PCA dimension reduction

(a) (b)

Figure 5. Original image (a), feature inverse map after PCA dimensionality reduction (b)

4. Support Vector Machine(SVM)

In order to study and realize the automatic identification of vehicle models, in view of the fact that the road traffic video surveillance system is based on the limited sample of real-time processing of a large number of data, this paper uses SVM theory to build the classifier.

Support Vector Machine (SVM) is initially classified for two types, and is applied in two cases: linear separable and linear indivisible. Nonlinear problems can be transformed into linear problems in high-dimensional space according to non-linear transformation. Of course, SVM can be extended to multi-classification problems, and can skillfully solve many practical problems such as small sample size, high dimension, non-linearity, local minimization and so on.

4.1 Linear (Approximately) Separable

Given the training sample set $D = (x1, y1), (x2, y2), \cdots, (xm, ym)$, where $yi \in \{-1, +1\}$, the most basic idea of classification learning is to find a partition hyperplane in the sample space based on the training set D, and separate samples of different categories. In the sample space, the division of the hyperplane can be described by a linear equation: $W^T x + b = 0$, where W is the normal vector,

which determines the directions of the hyperplane, and b is the displacement, which determines the distance between the hyperplane and the coordinate origin.

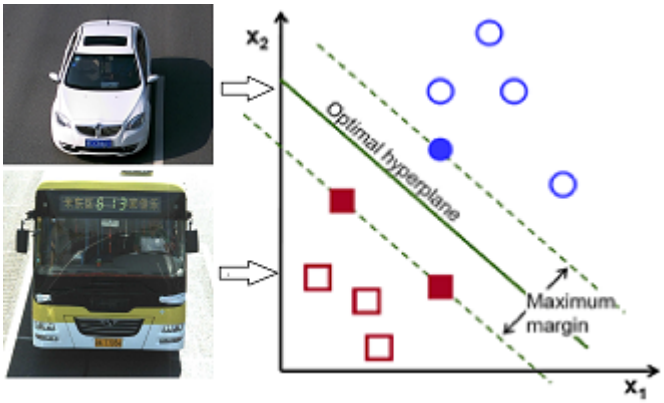

Figure 6. There are multiple divided hyperplanes separating the two types of training samples.

Intuitively, there are many hyperplanes that can separate training samples, but we should find the green one in Figure 6, which is located in the "positive middle" of the two types of training samples, because the partitioned hyperplane has the best "tolerance" to the local perturbation of the training samples. For example, due to the limitations of the training set or the noise factor, the samples outside the training set may be closer to the separation boundaries of the two classes than the training samples in Figure 6, which will cause many partitioning hyperplanes to have errors. The green hyperplane is least affected, that is, the result of this division of the hyperplane is robust.

4.2 Nonlinear Separable

Support Vector Machine (SVM) realizes non-linear classification by mapping input vectors to a high-dimensional feature space through a pre-selected non-linear mapping (kernel function), in which the optimal classification hyperplane is constructed. The classification plane of transformation space is: $W^T\varphi(x)+b=0$.

If a function $\mathrm{K}(xi, xj)$ can be constructed in the original space to be equal to the inner product operation $[\varphi(xi), \varphi(xj)]$ of the transformed space, then the sample data can be mapped to the high-dimensional or even infinite-dimensional space by non-linear transformation, and the optimal classification hyperplane can be constructed in the high-dimensional space. However, when solving the optimization problem and calculating the classification plane, it is not necessary to explicitly calculate the nonlinear function, and even without knowing its specific form, it is only necessary to calculate the function $\mathrm{K}(xi, xj)$, that is, the kernel function. The commonly used kernels mainly are polynomial kernels, radial RBF kernels and Sigmoid kernels.

In this paper, the classification function formed by SVM in dataset has the following property: it is a linear combination of a set of non-linear functions with support vectors as parameters. Therefore, the expression of the classification function is only related to the number of support vectors, and is independent of the dimensions of the space. This method is especially effective when dealing with classification of high-dimensional input space.

5. Experimental Result

In order to verify the performance of this algorithm for vehicle-type recognition, in the training stage, vehicle images are selected as positive samples and different traffic background images are selected as negative samples. The positive and negative sample models are obtained by training with SVM classifier. The test dataset used in the experiment collected more than 3,000 images of various vehicles. The dataset includes different types of vehicle pictures, such as shooting time, angle and distance. For vehicle identification testing experiments, it includes common SUV models, cars, vans, trucks, etc.

The experimental results show that in the same type of vehicle shape recognition, the method has a higher recognition rate, the average recognition rate reaches 92.6%, which is higher than the 90.3% of the neural network algorithm and the 87.4% of the traditional SURF algorithm. The recognition speed is also 20% - 38% higher than that of traditional methods.

Table 1. Accuracy of Vehicle Type Recognition Based on Experimental Algorithms

No.	Algorithm	Accuracy (%)	Average Recognition Time (s)
1	SURF	87.4	0.41
2	CNN	90.3	0.33
3	FHOG-SVM(This paper)	92.6	0.26

6. Summary

In this paper, the traditional HOG features are improved, and FHOG algorithm is proposed to extract the features of vehicle images. At the same time, PCA is used to reduce the dimension of feature vectors and reduce the complexity of calculating feature vectors. Support Vector Machine learning method is used as a sample classifier to classify and recognize vehicle types.

The experimental results show that the improved HOG-SVM vehicle recognition algorithm proposed in this paper has higher recognition rate and recognition speed compared with the traditional HOG algorithm, and has strong anti-interference to different environmental background, different angles and different distances, which can meet the needs of vehicle recognition and supervision in intelligent transportation system. However, how to optimize the algorithm and reduce the recognition time remains to be further studied.

References

[1]. Lim T R, Guntoro A T. Car recognition using Gabor filter feature extraction [C]. Asia-Pacific Conference on Circuits and Systems. IEEE,2002(2), APCCAS, v 2, p 451-455.

[2]. Liqin Hua, Wei Xu, Tuo Wang, et al. Vehicle Recognition Using Improved SIFT and Multi-View Model[J]. Journal of Xi' A Jiao Tong University. Vol. 47 (2013) No. 4, p. 92-99.

[3]. Dong Z, Pei M, He Y, et al. Vehicle type classification using unsupervised convolutional neural network[C]. 22nd International Conference on Pattern Recognition. ICPR 2014. IEEE, p.172-177.

[4]. Abadi, Elyas Abbasi Jennat, Amiri, Soheyl Akhlaghi, et al. Vehicle model recognition based on using image processing and wavelet analysis[J]. International Journal on Smart Sensing & Intelligent Systems, Vol.8(2015), No. 4, p 2212-2230.

[5]. Tong Zhang, Ping Zhang. Method of Vehicle Type Recognition Based on Improved Harris Corner Detection[J]. Computer Science. Vol. 44 (2017) No. 11A, p. 256-258.

[6]. Z. Zhang, T. Tan, K. Huang, et al. Three-dimensional deformable-model-based localization and recognition of road vehicles[J]. Transactions on Image Processing. Vol. 21 (2012) No. 1, p. 1-13.

[7]. Shan Ying, Sawhney Harpreet S, Kumar, Rakesh. Unsupervised learning of discriminative edge measures for vehicle matching between nonoverlapping cameras[J]. IEEE, Transactions on Pattern Analysis and Machine Intelligence. Vol. 30 (2008) No. 4, p. 700-711.

[8]. Lowe, David G. Distinctive image features from scale-invariant key points[J]. International Journal of Computer Vision. Vol. 60 (2004) No. 2, p. 91-110.

[9]. Dong Zhen, Wu Yuwei, Pei Mingtao, et al. Vehicle Type Classification Using a Semi supervised Convolutional Neural Network[J]. IEEE Transactions on Intelligent Transportation Systems (TITS), Vol. 16 (2015) No. 4, p. 2247-2256.

[10]. Yi Zhuang, Wei Chen, Zhong Yi. Self-optimizing Object Recognition Based on Improved HOG+SVM and MFCC Features[J]. Software Guide. Vol. 17 (2018) No. 12, p. 3-8.

[11]. Yang M J, Zheng H R, Wang H Y, et al. Combining feature ranking with PCA: An application to gait analysis[C]. International Conference on Machine Learning and Cybernetics. IEEE, 2010. p. 494-499.