ISSN: 2688-9323

Computer Artificial Intelligence Technology in Vehicle Image Recognition

Zhilie Gao, Yanjun Li, Yuyuan Cao, Guangtao Wang

College of civil aviation, Nanjing University of Aeronautics and Astronautics, Nanjing, Jiangsu, 210016, China

Abstract

This article proposes an image pre-processing method applied to the video vehicle recognition system. This method deblurs and restores the collected vehicle images, and then uses a fast gradient algorithm to obtain the background and vehicle edge images. We use two edge images to subtract and segment the vehicle image, then apply the image mean shift algorithm, and perform straight line fitting and filling processing, and finally use the contour tracking method to obtain a complete vehicle contour map, so as to effectively extract feature parameters.

Keywords

Computer; Artificial Intelligence; Vehicle Image Recognition.

1. Introduction

In recent years, with the rapid development of the national economy and the gradual improvement of comprehensive national strength, the needs and applications of intelligent surveillance systems in the early warning, security prevention and modern management of military, banking, transportation and other key industries have become more and more extensive. Especially with the acceleration of urbanization in various countries, the number of motor vehicles in cities is increasing year by year, while the construction of urban transportation infrastructure is relatively lagging, the pressure on urban traffic has greatly increased [1]. The main means of obtaining traffic information is vehicle detection. At present, image recognition technology has been more and more widely used in the detection of traffic flow in intelligent transportation. The application of vehicle image recognition technology combined with image processing to identify the number, type and movement characteristics of many vehicles within the same field of view can provide important real-time basis for intelligent traffic control.

2. Mean Shift Vehicle Image Recognition Algorithm

2.1. Model Establishment

Select the initial vehicle image area by manual positioning or vehicle image detection algorithm. In order to reduce the influence of different sizes, this vehicle image template needs to be normalized, and the height and width of the new vehicle image template are h_x and h_y respectively. Considering that the pixels on the periphery of the vehicle image are susceptible to interference from the background or other moving vehicle images, in order to improve the robustness of the tracking algorithm, a kernel function weighted histogram is used to assign different weights to the pixels according to the distance of different pixels from the centre of the vehicle image [2]. This paper uses the Pantechnicon kernel function, which is defined as follows:

$$K_E(x)=\begin{cases}\frac{1}{2}c_d^{-1}(d+2)(1-\|x-y_0\|^2), & \|x-y_0\|^2\le 1\\ 0, & otherwise\end{cases} \tag{1}$$

Among them, c_d is the volume of the d dimensional unit ball. In the image, $d=2$, x and y_0 are all points and the centre point in the vehicle image window, respectively, and $\|x-y_0\|^2$ is the normalized Euclidean distance from point x in the window to the centre point y_0. We assume that $\{\vec{x}_i^*\}_{i=1\cdots n}$ is the pixels at different positions after normalization of the vehicle image area, and define $b(\vec{x}_i^*)$ as the index value of each point in the vehicle image area mapped to the corresponding histogram box, then the probability of the feature vector u of the vehicle image The density is as follows:

$$\hat{q}_u=C\sum_{i=1}^{n}k_E(\|\vec{x}_i^*\|^2)\delta\left[b\left(\vec{x}_i^*\right)-u\right] \tag{2}$$

Among them, $\delta(\bullet)$ is the Kronecker Δ function, $u=1,\dots,m$, n is the number of points in the vehicle image area, and m is the number of elements in the feature space. $k_E(\|x\|^2)$ is the contour function of the kernel function $K_E(x)$, namely $K_E(x)=k_E(\|x\|^2)$. C is the normalization constant, defined as:

$$C=\frac{1}{\sum_{i=1}^{n}k_E\left(\|\vec{x}_i^*\|^2\right)} \tag{3}$$

2.2. Calculate the Candidate Vehicle Image Model

Suppose the centre coordinate of the candidate vehicle image area is at the position $\bar{y}$ of the detected image, and $\{\bar{x}_i\}_{i=1\cdots n_h}$ is the point in the current vehicle image candidate area [3]. Using the kernel function $k_E(x)$ with scale h, the probability density of candidate vehicle images can be calculated:

$$\hat{p}_u=C_h\sum_{i=1}^{n_h}k_E\left(\left\|\frac{\bar{y}-\bar{x}_i}{h}\right\|^2\right)\delta[b(\bar{x}_i)-u] \tag{4}$$

Among them, C_h is the normalization constant, which can be expressed as:

$$C_h=\frac{1}{\sum_{i=1}^{n_h}k_E\left(\left\|\frac{\bar{y}-\bar{x}_i}{h}\right\|^2\right)} \tag{5}$$

2.3. Judging the Similarity between the Vehicle Image Model and the Candidate Model

This paper adopts the similarity measure based on the Bhattacharyya coefficient. The Bhattacharyya coefficient is a measure of divergence, and its geometric meaning is the cosine value of the angle of two n-dimensional vectors [4]. Define the distance between the two discrete distributions of the vehicle image model and the candidate model as:

$$d(\bar{y})=\sqrt{1-\rho[\hat{\bar{p}}(\bar{y}),\hat{\bar{q}}]} \tag{6}$$

Then the Bhattacharyya coefficient can be expressed by vehicle image template distribution $\bar{q}$ and candidate vehicle image distribution $\bar{p}(\bar{y})$ as:

$$\hat{\rho}(\bar{y})\equiv\rho[\hat{\bar{p}}(\bar{y}),\hat{\bar{q}}]=\sum_{u=1}^{m}\sqrt{\hat{p}_u(\bar{y})\hat{q}_u} \tag{7}$$

Formula (7) is a simplified formula of Bhattacharyya coefficient, used to measure the correlation coefficient between p and q. The value of $\hat{\rho}(\bar{y})$ is between [0, 1]. The larger the value, the higher the similarity between the two models, and the closer the candidate model is to the real position of the vehicle image.

2.4. The Positioning of the Vehicle Image in the Current Frame

In order to find the candidate model that best matches the vehicle image model, the Bhattacharyya coefficient must be maximized. In the current frame, use the position $\bar{y}_0$ of the vehicle image in the previous frame as the initial position of the search vehicle image, and then find the optimal vehicle image position $\bar{y}_1$ in the neighbourhood of $\bar{y}_0$ so that $\hat{\rho}(\bar{y}_1)$ is the largest. Use formula (1) and formula (4) to calculate the probability density $\bar{q}$ of the vehicle image template and the probability density $\{\hat{p}_u(\hat{\bar{y}}_0)\}_{u=1\cdots m}$ of the candidate area of the vehicle image cantered at $\bar{y}_0$ in the current frame, and then use Taylor formula to expand $\hat{\rho}(\bar{y})$ at $\hat{p}_u(\hat{\bar{y}}_0)$ to get :

$$\rho[\hat{\bar{p}}(\hat{\bar{y}}),\hat{\bar{q}}]\approx\frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{\bar{y}}_0)\hat{q}_u}+\frac{1}{2}\sum_{u=1}^{m}\hat{p}_u(\bar{y})\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\bar{y}}_0)}} \tag{8}$$

Substituting $\hat{p}_u$ into the deformation formula is:

$$\rho[\hat{\bar{p}}(\hat{\bar{y}}),\hat{\bar{q}}]\approx\frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u(\hat{\bar{y}}_0)\hat{q}_u}+\frac{C_h}{2}\sum_{i=1}^{n_h}\omega_i k_E\left(\left\|\frac{\bar{y}-\bar{x}_i}{h}\right\|^2\right) \tag{9}$$

In,

$$\omega_i=\sum_{u=1}^{m}\sqrt{\frac{\hat{q}_u}{\hat{p}_u(\hat{\bar{y}}_0)}}\delta[b(\bar{x}_i)-u] \tag{10}$$

The first term on the right side of the equation (10) is independent of $\bar{y}$, that is, it has nothing to do with the minimization of $d(\bar{y})$, and the second term can be regarded as the density of the profile function with $k_E(x)$ as the kernel function and the weight of ω_i at the sample point $\bar{y}$ estimate. To maximize this item, use the mean shift process to detect and gradually recursively calculate the new vehicle image position $\bar{y}_1$ from $\bar{y}_0$:

$$\vec{y}_1 = \frac{\sum_{i=1}^{n_h} \vec{x}_i \omega_i g\left(\left\|\frac{\hat{\vec{y}}_0 - \vec{x}_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h} \omega_i g\left(\left\|\frac{\hat{\vec{y}}_0 - \vec{x}_i}{h}\right\|^2\right)} \tag{11}$$

Figure 1 shows the flow chart of the Manshift algorithm applied to moving vehicle image tracking. According to the properties of the Pantechnicon function in the Bhattacharyya coefficient, at this time $g(x)$ is a constant, then the formula (11) can be simplified as:

$$\hat{\vec{y}}_1 = \frac{\sum_{i=1}^{n_h} \hat{\vec{x}}_i \omega_i}{\sum_{i=1}^{n_h} \omega_i} \tag{12}$$

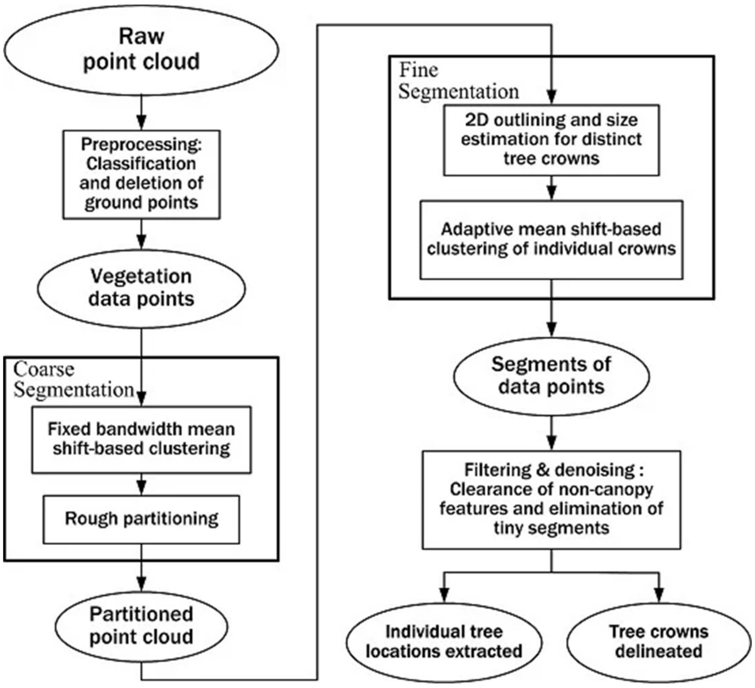

Figure 1. Tracking flowchart of Manshift algorithm

The specific implementation steps of the Manshift tracking algorithm are as follows:

Step1. Select the vehicle image area in the initial frame, extract the effective features of the moving vehicle image, use the formula (3.8) to calculate the probability density of the vehicle image at $\vec{y}_0$, obtain the vehicle image model $\{\hat{q}_u(\hat{\vec{y}}_0)\}_{u=1\cdots m}$, and set a small normal number ε as the centroid convergence Threshold

Step2. In the current frame, initialize a candidate area of the vehicle image cantered on $\vec{y}_0$, and use formula (5) to calculate the probability density of the candidate area at the current frame $\vec{y}_0$ to obtain the candidate model $\{\hat{p}_u(\hat{\vec{y}}_0)\}_{u=1\cdots m}$ and the Bhattacharyya coefficient $\rho[\hat{\vec{p}}(\vec{y}_0), \hat{\vec{q}}] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(\vec{y}_0)\hat{q}_u}$;

Step3. Calculate the weight $\{\omega_i\}_{i=1\cdots n_h}$ according to formula (12), and then calculate the new candidate vehicle image position $\vec{y}_1$ by formula (8);

Step4. Calculate the probability density $\{\hat{p}_u(\hat{\bar{y}})\}_{u=1\cdots m}$ and the Bhattacharyya coefficient $\rho[\hat{\bar{p}}(\bar{y}_1),\hat{\bar{q}}]=\sum_{u=1}^{m}\sqrt{\hat{p}_u(\bar{y}_1)\hat{q}_u}$ of the candidate vehicle image at the new position $\bar{y}_1$ respectively according to formulas (11) and (2);

Step5. Judge whether $\rho[\hat{\bar{p}}(\bar{y}_1),\hat{\bar{q}}]<\rho[\hat{\bar{p}}(\bar{y}_0),\hat{\bar{q}}]$ is established. If it is established, then $\bar{y}_1 \leftarrow \frac{1}{2}(\bar{y}_0+\bar{y}_1)$, and then judge whether $\|\bar{y}_1-\bar{y}_0\|<\varepsilon$ is established [5]. If the condition is satisfied, the loop judgment is terminated; otherwise, $\bar{y}_0 \leftarrow \bar{y}_1$, jump to step 2 to continue to find the candidate vehicle image position that meets the condition.

Step6. Determine whether the current frame is the last frame. If it is, the tracking process ends. Otherwise, read the next frame as the current frame, return to Step2, and continue to iteratively find the position of the vehicle image in the current frame.

Through the above algorithm steps, it can be concluded that the iterative process of the Manshift algorithm can be seen as a process of continuous movement from $\bar{y}_0$ to $\bar{y}_1$. Each Manshift vector determines a movement and increases the value of the similarity function, eventually reaching the extreme value of the similarity function, the position at this time is also the best matching position of the vehicle image.

3. Experimental Inspection

We apply the Manshift algorithm to human tracking in video sequences. The experimental simulation is implemented on a computer with Core23GHz CPU and 2G memory using MATLAB programming, and the size of the processed video image is pixels. Figure 2 shows the results of tracking vehicles in a set of real shot videos of traffic intersections, showing frames 8, 42, 83, and 135 respectively [6]. It can be seen from the experimental results that the Manshift algorithm can accurately track the vehicle image under a simple background. The processed image contour has a change compared with the real vehicle contour, but it can be tolerated with a certain accuracy. At the same time, it makes a vehicle completely form an object and improves the accuracy of vehicle counting.

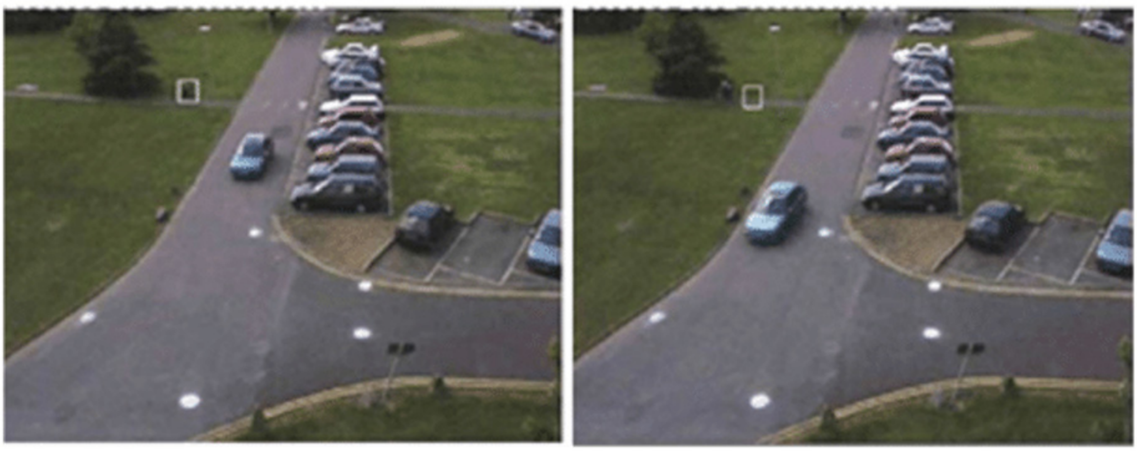

Figure 2. Manshift algorithm applied to vehicle tracking

4. Conclusion

The video vehicle recognition system has broad application prospects in modern traffic management, and the image pre-processing technology plays a vital role in the recognition

ISSN: 2688-9323

efficiency and accuracy of the entire system. The image processing method adopted in this paper is suitable for realization and fast calculation speed, which is a useful exploration of online vehicle recognition.

References

[1] Chang, W. J., & Chen, L. B. Design and implementation of an intelligent motorcycle helmet for large vehicle approach intimation. IEEE Sensors Journal, 19(10) (2019) 3882-3892.

[2] Gwak, J., Jung, J., Oh, R., Park, M., Rakhimov, M. A. K., & Ahn, J. A review of intelligent self-driving vehicle software research. KSII Transactions on Internet and Information Systems (TIIS), 13(11) (2019) 5299-5320.

[3] Soon, F. C., Khaw, H. Y., Chuah, J. H., & Kanesan, J. Vehicle logo recognition using whitening transformation and deep learning. Signal, Image and Video Processing, 13(1) (2019) 111-119.

[4] Wu, D., & Hu, H. Automatic recognition of vehicle image based on monocular vision and environmental perception. International Journal of Vehicle Information and Communication Systems, 5(2) (2020) 173-186.

[5] Asgarian Dehkordi, R., & Khosravi, H. Vehicle Type Recognition based on Dimension Estimation and Bag of Word Classification. Journal of AI and Data Mining, 8(3) (2020) 427-438.

[6] Qi, G., Wang, H., Haner, M., Weng, C., Chen, S., & Zhu, Z. Convolutional neural network based detection and judgement of environmental obstacle in vehicle operation. CAAI Transactions on Intelligence Technology, 4(2) (2019) 80-91.