Research Article

Traffic Flow Detection at Road Intersections Based on *K*-Means and NURBS Trajectory Clustering

Jun-fang Song ⓘ, Shu-yu Wang, and Hai-li Zhao

School of Information Engineering, Xizang Minzu University, Xianyang, Shaanxi 712082, China

Correspondence should be addressed to Jun-fang Song; 284786635@qq.com

In view of the variety and occlusion of vehicle target motion on the urban intersection, it is difficult to accurately detect the traffic flow parameters in all directions and categories of the intersection, so an improved *k*-means trajectory clustering method based on NURBS curve fitting is designed to obtain the traffic flow parameters. Firstly, the B-spline quadratic interpolation function is used to fit the smooth NURBS curve of vehicle trajectory; secondly, *K*-means clustering is used to measure the minimum distance, and the location of the first and last end points of the vehicle trajectory is used to realize the automatic division of the intersection area; finally, according to the intersection area where the start and end points of vehicle trajectory belong, respectively, the moving mode of a vehicle is determined, and the traffic flow parameters are classified and counted. Experiments show that the method has high accuracy and simple algorithm, which can meet the application requirements of intelligent transportation. It can provide effective data for traffic congestion analysis and lane occupancy estimation, and it is an important parameter for dynamic time setting of intersection information lights.

1. Introduction

The perception and prediction of traffic scene mainly includes the acquisition of vehicle flow parameters, the identification of traffic abnormal behavior, and the prediction of traffic congestion status, which is one of the hot areas of traffic scholars in various countries. It is an important task in the traffic scene perception system to obtain the traffic flow parameters in the video sequence. Real-time and accurate vehicle flow parameters can provide important data support for urban traffic control and management organizations. In addition, vehicle flow parameters on urban roads also reflect the current traffic conditions to a certain extent, for example, road traffic congestion status, lane occupancy rate, and road status; these information are conducive to traffic accident early warning, road congestion prediction and automatic planning of travel path [1]. Statistics of the traffic flow information of urban road intersections in different periods and in all directions can also be used as an important basis for the setting of dynamic timing parameters of intersection signal lights. It can effectively improve the traffic efficiency of the road, improve the traffic condition of the urban road network, and bring significant economic benefits to the society [2]. The traffic scene of urban intersection is complex, the vehicle motion mode is changeable, and the occlusion is serious, which make it difficult to detect and track the moving target accurately and continuously. In order to obtain the traffic flow parameters of urban intersections, it is very important to accurately identify the vehicle target and determine its moving direction.

According to the different calculation methods and principles, there are currently four categories of traffic flow statistics: the target detection method [3–6], feature point motion trajectory clustering method [7–9], regional regression method [10, 11] and density estimation method [12–14]. Seenouvong et al. [3] completed the vehicle count in the set area by combining background subtraction with morphological filtering, with an accuracy of 96%; Memon et al. [4] introduced the Gaussian mixture model (GMM) to quickly detect and count vehicles in the field of view and realized the self-classification of moving targets by contour comparison; Chen [5] proposed to use the B-spline curve

method to obtain the vehicle area, so as to realize the vehicle count; Leonel et al. [6] proposed a video sequence vehicle counting method based on augmented quantum space learning. The accuracy rate of counting vehicles at an average speed of 26 frames per second is 96.6%, and it can well deal with camera shake and sudden illumination changes caused by the environment and automatic camera exposure. Rabbouch et al. [7] designed a pattern recognition system based on unsupervised clustering of tracks, which realized intelligent detection, counting, and recognition of traffic targets; Mehboob et al. [8] built a real-time system through analyzing the temporal and spatial characteristics of vehicle trajectory, which it can deduce and track the target behavior online, and joint Hungarian tracking algorithm to count the number of vehicles; in terms of vehicle flow statistics by different types, Rauf et al. [9] proposed a method based on target tracking and convolutional neural network transfer learning; the regional regression method is well reflected in the literature [10, 11]. Liang et al. [10] constructed a regression model by extracting the edge features and gradient features of vehicles on the highway to obtain the vehicle flow parameters; Chen [11] suggested a hierarchical classification-based regression model for accurate vehicle counting in view of the complex and changeable characteristics of the actual traffic scene; Lempitsky and Zisserman [12] proposed the target counting algorithm framework based on density estimation in 2010, which is the most common and concerned counting framework at present. The method first generates the truth value image set of the target density distribution by using the target center graph manually annotated. Zhang et al. [13] introduced a vehicle density estimation method based on rank constraint regression and Fully Convolutional Networks (FCN), so as to realize accurate statistics of the number of vehicles. Based on the convolutional neural network (CNN), Sindagi and Patel [14] used deep learning to estimate vehicle density in crowded situations.

In a word, in order to obtain accurate traffic flow parameters, in addition to establishing a reasonable mathematical model, it is particularly important to generate features that can truly reflect the running status of vehicle targets. However, the scene of urban intersection is complex, the moving state of the target is changeable, there are many occlusions, and rich target feature information is needed. The target trajectory can provide a wealth of spatiotemporal data such as direction and speed, and it has good stability in all-weather operation. This manuscript takes the trajectory of vehicle as an important feature, firstly, NURBS is used to fit the trajectory curve; then, k-means algorithm is used to complete the clustering of the trajectory curve; finally, based on the clustering results, the short-term traffic flow in each region and direction of the intersection is predicted.

2. K-Means Trajectory Clustering Based on NURBS Curve Fitting

Trajectory clustering is to use the appropriate similarity measurement method to complete the best classification of the obtained trajectory features. To explore suitable similarity measurement criteria to ensure the best clustering

results, we choose K-means clustering algorithm, which has fast calculation speed and high efficiency and is suitable for finding convex shape clusters. Aiming at the defects of the randomness of initial mean and the sensitivity of noise outliers, this manuscript introduces NURBS curve fitting to solve the problem of noise points when the target pairs to form a trajectory. The NURBS curve can be regarded as the sum of the product of a series of control points $\{V_i\}$ ($i = 0, 1, 2, \ldots, n$) and the base function $N_{i,k}(u)$ determined by the known node sequence $\{u_i\}$, ($i = 0, 1, 2, \ldots, n$). Combined with a series of weight factors that can affect the shape of the curve, the mathematical expression of the vector function is as follows:

$$
\begin{aligned}
P(u) &= \frac{\sum_{i=0}^{n} \omega_i d_i N_{i,k}(u)}{\sum_{i=0}^{n} \omega_i N_{i,k}(u)} = \sum_{i=0}^{n} d_i B_{i,k}(u), \\
B_{i,k}(u) &= \frac{\omega_i N_{i,k}(u)}{\sum_{j=0}^{n} \omega_j N_{j,k}(u)}.
\end{aligned}
\tag{1}
$$

In the formula, $B_{i,k}(u)$ is the basis function of the k-th normal B-spline; d_i is the control vertex, also known as the De Boor point; ω_i is the weight factor; and u is the parameter value.

The weight factor introduced by the NURBS function can realize the local correction of curve flexibly. Before the k-means algorithm clusters the trajectory, first, our method fits the NURBS curve through B-spline quadratic interpolation and, then, calculates the first-order guide vector of all the trajectory points, which is equal to the tangent slope passing through the point in 2D plane. If the curve is smooth and continuous, then the tangent slopes of all trace points on the curve will change between the slopes of the two control points; otherwise, it indicates that there are irregular points on the curve. According to the nearest point search algorithm of k-d Tree [15], all irregular points can be found by traversing the whole trajectory, and the best trajectory can be obtained by NURBS curve fitting again.

2.1. NURBS Curve Fitting of Track Points.
Based on detection results of vehicles, their track matching was used to eliminate irregular measurement points by the piecewise straight-line fitting method. However, the trajectory fitted by this method is not smooth, and there are many inflexion points; secondly, due to the influence of perspective projection transformation of image, the position of many points has been distorted, so it is unable to predict the macromotion trajectory of the vehicle intuitively. In this paper, the NURBS function in the theory of curve geometry [16], is applied to the fitting of vehicle trajectory. It has good invariance to perspective projection transformation. It can effectively eliminate irregular points, improve the representation ability of trajectory, and ensure that the traffic flow at intersections can be accurately predicted through the processing of the trajectory subregion and subdirection. The essence of NURBS curve fitting based on trajectory points is to interpolate or fit the measurement points into smooth curves.

The trajectory curve fitting steps are as follows:

(1) We calculate the parameter values and node vectors in the U or V direction of the curve from the measurement point data of the vehicle target trajectory

(2) The node vector interval of each parameter value is determined, and all basis functions are obtained

(3) A set of equations with curve control point (type value point) as unknown quantity is established, and the coordinates of the curve control point are obtained by solving the equations

We assume that the measurement point (track point) of the track is

$$\{B_k\}, \quad k = 0, 1, \ldots, n. \tag{2}$$

The NURBS curve is fitted by the interpolation function to the measuring points:

$$C(u) = \sum_{i=0}^{n} N_{i,p}(u)P_i, U$$

$$= \left\{ \underbrace{0, \ldots, 0}_{p+1}, u_{p+1}, \ldots, u_{m-p-1}, \underbrace{1, \ldots, 1}_{p+1} \right\}. \tag{3}$$

Among them, P_i is the type value point, the parameter $\overline{u}_k$ corresponding to measurement point B_k is calculated by the chord length parameterization method, and the node vector U is determined by the mean value method. In this way, after the basis function is found, it is substituted with the type value point into the following formula for calculation:

$$B_k = C(\overline{u}_k) = \sum_{i=0}^{n} N_{i,p}(\overline{u}_k)P_i. \tag{4}$$

After that, a system of equations with the point P_i as an unknown is obtained, and it is composed of $n + 1$ equations but $n + 3$ unknowns, belonging to the overdetermined system of equations with no linear solution, so we need to establish other two equations to solve it. Considering the particularity of the first and end points, the mathematical relationship between the first-order derivative vector D_0, D_n and the adjacent type value points P_0, P_1 and P_{n-1}, P_n is as follows:

$$-P_0 + P_1 = \frac{u_{p+1}}{p}D_0, \tag{5}$$

$$-P_{n-1} + P_n = \frac{u_{m-p-1}}{p}D_n. \tag{6}$$

Combining equations (5) and (6) with equation (3), a system of linear equations whose coefficient matrix is $(n+3)$

$* (n+3)$ is obtained, and then, we can solve this system and get the 2D coordinates of $n + 3$ control points. Then, the NURBS curve is fitted with a nonrational quadratic B-spline interpolation function, and the derivative of the vector is obtained at any point as follows:

$$C'(u) = -2(1-u)P_{j-1} + 2(1-2u)$$

$$\times \frac{-(1-u)^2 P_{j-1} + -u^2 P_{j+1}}{2u(1-uP_j)} + 2uP_{j+1}. \tag{7}$$

Among them, P_{j-1}, P_j, and P_{j+1} are three adjacent measurement points in the NURBS curve, and the parameter $u = (\sqrt{(x_{j_-}x_{j-1})^2 + (y_{j_-}y_{j-1})^2})/(\sqrt{(x_{j_-}x_{j-1})^2 + (y_{j_-}y_{j-1})^2} + \sqrt{(x_{j+1} - x_j)^2 + (y_{j+1} - y_j)^2})$.

2.2. K-Means Trajectory Clustering. There is a big deviation in the k-means algorithm due to the uncertainty of the initial clustering center. In our manuscript, the minimum distance is proposed as a measurement criterion to select the initial clustering center. In view of the fact that the start and end points of each moving vehicle's trajectory at the intersection must be clustered in each intersection area, it is accurate and simple to predict the trajectory trend with them. All the vehicle's movement trajectories obtained by fitting NURBS are represented by sets $\{P_{\text{start}}\}$ and $\{P_{\text{end}}\}$ of start and end points. We calculate the normalized Euclidian distance between any two points in set $\{P_{\text{start}}\}$ and $\{P_{\text{end}}\}$; find the samples with the shortest distance by the quicksort method and write it as P_{d0} and P_{dn}; calculate the Euclidean distance between them and residual sample points in the set $\{P_{\text{start}}\}$ and $\{P_{\text{end}}\}$; find out the sample points with the smallest distance from the center of P_{d0} and P_{dn}; increase them to P_{d0} and P_{dn}, and repeat this process until we have k sets of objects. Finally, k object sets are averaged numerically to form a group of clustering centers, which are used as the initial clustering centers when k-means algorithm is used to complete the trajectories clustering, and k clustering centers J_k are obtained. The centroid is taken as the central location of the intersection J_e, and k is equal to the number of intersection areas.

3. Regional Traffic Flow Prediction

In order to facilitate the prediction of partitioned traffic flow, in the polar coordinate system, the motion direction angle (relative to J_e) of the trajectory point is introduced, which is as follows:

TABLE 1: Experimental scene data.

Feature	Scenario		
	Scenario 1	Scenario 2	Scenario 3
Area number	4	4	3
Direction number	12	12	6
Time quantum	PM 5:00 to 5:45	AM 7:00 to 8:00	AM 7:00 to 8:00
Intersection property	Intersection	Intersection	T-junction

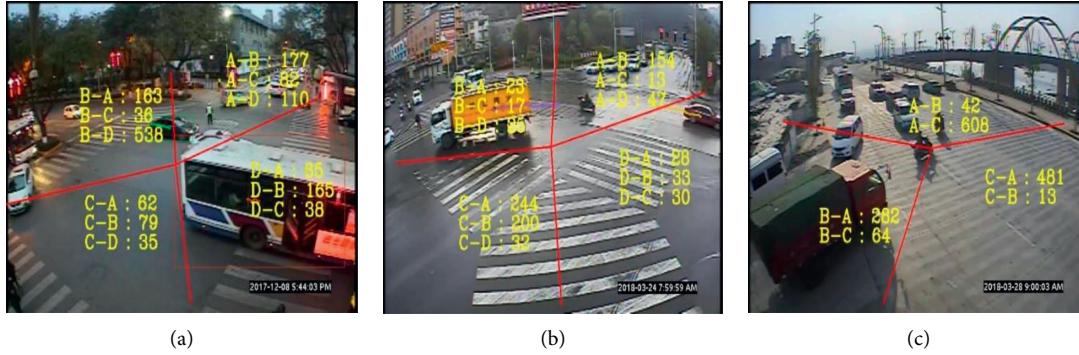(a)   (b)   (c)

FIGURE 1: Vehicle statistical results of each intersection scene by region. (a) Scenario 1. (b) Scenario 2. (c) Scenario 3.

$$
\begin{cases}
\dfrac{180°}{\pi * \arctan\left((y_i - y_e)/(x_i - x_e)\right)}, & \text{if: } x_i > x_e, y_i > y_e, \\[2mm]
90°, & \text{if: } x_i = x_e, y_i > y_e, \\[2mm]
\dfrac{180° + 180°}{\pi * \arctan\left((y_i - y_e)/(x_i - x_e)\right)}, & \text{if: } x_i < x_e, y_i > y_e, \\[2mm]
180°, & \text{if: } x_i < x_e, y_i = y_e, \\[2mm]
\dfrac{270° + 180°}{\pi * \arctan\left((y_i - y_e)/(x_i - x_e)\right)}, & \text{if: } x_i < x_e, y_i < y_e, \\[2mm]
270°, & \text{if: } x_i = x_e, y_i < y_e, \\[2mm]
\dfrac{360° + 180°}{\pi * \arctan\left((y_i - y_e)/(x_i - x_e)\right)}, & \text{if: } x_i > x_e, y_i < y_e, \\[2mm]
360°, & \text{if: } x_i > x_e, y_i = y_e, \\[2mm]
0°, & \text{others.}
\end{cases}
\tag{8}
$$

Taking the intersection traffic scene as an example, four θ values can be obtained, and the ascending order is $0° \leq \theta_1 < \theta_2 < \theta_3 < \theta_4 \leq 360°$; we calculate $\theta' = \theta_1 + \theta_2/2, \theta_2 + \theta_3/2, \theta_3 + \theta_4/2, \theta_4 + \theta_1/2$ and arrange it in ascending order as $0° \leq \theta'_1 < \theta'_2 < \theta'_3 < \theta'_4 \leq 360°$; then, we call (θ'_1, θ'_2) A-block,

(θ'_2, θ'_3) B-block, (θ'_3, θ'_4) C-block, and $(\theta'_4, 360°)$ and $0, \theta'_1$ the D-block. Other intersection scene intersection area division methods can be analogized.

Calculating the direction angle θ_k and θ'_k of motion of the points P_{start} and P_{end}, the vehicle's movement pattern can be known by judging their respective regions. Assuming that θ_k belongs to zone A and θ'_k belongs to zone C, it is believed that there are vehicles in the direction from A to C. After traversing all the trajectories, traffic flow prediction results in each direction of the classification vehicle can be obtained. Vehicle types can also be obtained based on vehicle target detection technology, so each vehicle trajectory can be marked with category, so that the flow parameters of different types in each region can be counted.

4. Experiment and Result Analysis

The experiment only counts the vehicle flow. Based on the statistical data, more traffic flow parameters such as queue length and road proportion can be calculated. Three different intersection scenes including morning and evening rush hours are selected for experiments about 7 days, and the characteristics of experimental data are shown in Table 1.

For each scene, intersection partitions were obtained by combining k-means trajectory clustering through NURBS curve fitting, and vehicle trajectories in all directions were statistically analyzed to obtain specific results of vehicle flow prediction of different types.

Figure 1 is the vehicle statistical results of each intersection scene on a certain day by region using our system. Figure 2 is the traffic flow of the scene separately counted every day. Tables 2–4 are the comparison over 7 days between the average results obtained by our algorithm and the manual statistical results based on the absolute error. Among
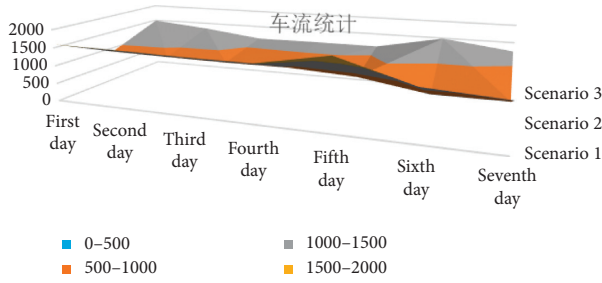
Figure 2: Vehicle statistical results of each day.

Table 2: Error analysis of average flow counting results in scenario 1.

| From | Traffic flow | To | | | | |
		A	B	C	D	Total
A	Manual statistics	0	201	93	123	417
	Detection	0	292	90	114	379
	Error	0%	4.9%	3.3%	7.3%	9.1%
B	Manual statistics	187	0	45	601	833
	Detection	179	0	44	546	762
	Error	4.3%	0	2.2%	9.2%	8.5%
C	Manual statistics	75	88	0	43	206
	Detection	71	83	0	42	188
	Error	5.3%	5.7%	0	2.3%	8.7%
D	Manual statistics	78	189	46	0	313
	Detection	74	179	45	0	300
	Error	5.1%	5.3%	2.2%	0	4.2%

Table 3: Error analysis of average vehicle flow counting results in scenario 2.

| From | Traffic flow | To | | | | |
		A	B	C	D	Total
A	Manual statistics	0	169	16	53	238
	Detection	0	154	16	52	224
	Error	0	8.9%	0%	1.9%	5.9%
B	Manual statistics	27	0	21	29	77
	Detection	27	0	21	28	77
	Error	0%	0	0%	3.4%	0%
C	Manual statistics	273	229	0	39	541
	Detection	267	208	0	39	496
	Error	2.3%	4.8%	0	0%	8.3%
D	Manual statistics	34	38	37	0	109
	Detection	34	37	37	0	102
	Error	0%	2.6%	0%	0	6.4%

them, N_E is the experimental value and N_T is the manual monitoring value.

$$\text{Error} = \left| \frac{N_E - N_T}{N_T} \right|. \tag{9}$$

The experimental results show that the statistical error of traffic flow is small, no matter for the T-junction or the intersection, and the maximum error is less than 10% and it can meet the needs of intelligent transportation

Table 4: Error analysis of average vehicle flow counting results in scenario 3.

| From | Traffic flow | To | | | |
		A	B	C	Total
A	Manual statistics	0	49	682	731
	Detection	0	49	628	664
	Error	0	0%	7.9%	9.1%
B	Manual statistics	313	0	72	385
	Detection	303	0	70	360
	Error	3.2%	0	3%	6.5%
C	Manual statistics	535	15	0	550
	Detection	512	15	0	521
	Error	4.3%	0%	0	5.3%

applications. In view of the traffic condition unblocked scene, the target is shielded less, and the relative accuracy is slightly higher. For the scene of intersection, the vehicle target has more occlusion, the trajectory caused by it has fault discontinuity, which is effectively bridged through adoption of NURBS curve fitting to irregular points. Therefore, the vehicle flow error of k-means trajectory clustering statistics is much lower than that of the single-lane or multilane statistical method, and the new idea effectively solves the problems of incomplete tracking trajectory and complex calculations.

5. Conclusions

The real-time and accurate parameters of traffic flow at urban road intersections are the important basis for setting dynamic timing parameters of traffic signals at intersections. This paper mainly studies the vehicle flow counting method at urban intersections based on vehicle trajectory analysis. On the basis of the traditional method to obtain the tracking trajectory, the smooth track curve is fitted based on the NURBS algorithm, and a minimum distance measurement criterion is designed to select the initial clustering center for improving the k-means trajectory clustering method, then through subordinating the first and end points of the track to the intersection area, the vehicle movement mode is determined, and the traffic flow statistics of each area at the intersection are realized. Experimental results show that the method effectively solves the changeable mode and severe occlusion of moving vehicle problems, under the complex scene at the intersection, traffic statistical error is less than the classical method, and the algorithm is simple and can provide reliable data automatically for the traffic accident warning, road congestion prediction, and route planning.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

References

[1] J. Zhang, F.-Y. Wang, K. Wang, X. Xu, and C. Chen, "Data-driven intelligent transportation systems: a survey," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624–1639, 2011.

[2] A. M. Lin, *[IEEE 2018 Smart City Symposium Prague (SCSP)-Czech Republic (2018.5.24-2018.5.25)] 2018 Smart City Symposium Prague (SCSP)–Traffic*, Czech Technical University, Prague, Czech Republic, 2018.

[3] N. Seenouvong, U. Watchareeruetai, C. Nuthong et al., "A computer vision based vehicle detection and counting system," in *Proceedings of the 2016 8th International Conference on Knowledge and Smart Technology (KST)*, pp. 224–227, Chiangmai, Thailand, November 2016.

[4] S. Memon, S. Bhatti, L. A. Thebo et al., "A video based vehicle detection, counting and classification system," *International Journal of Image, Graphics & Signal Processing*, vol. 10, no. 9, 2018.

[5] L. H. Chen, *Intelligent Detection System Of Intersection Traffic Flow Based On Video Image Processing*, Anhui University of Science and Technology, Huainan, China, 2019.

[6] R.-A. Leonel, P.-P. Jose, A. Hernandez-Suarez et al., "Vehicle counting in video sequences: an incremental subspace learning approach," *Sensors*, vol. 19, no. 13, 2019.

[7] H. Rabbouch, F. Saadaoui, and R. Mraihi, "Unsupervised video summarization using cluster analysis for automatic vehicles counting and recognizing," *Neurocomputing*, vol. 260, 2017.

[8] F. Mehboob, M. Abbas, R. Jiang, S. A. Khan, and S. Rehman, "Trajectory based vehicle counting and anomalous event visualization in smart cities," *Cluster Computing*, vol. 21, no. 1, pp. 443–452, 2018.

[9] X. Rauf, X. Xiong, and H. Cheng, "Multi-vehicle flow detection method based on target tracking and transfer learning," *Journal of Guilin University of Electronic Science and Technology*, vol. 39, no. 2, pp. 119–123, 209.

[10] M. Liang, X. Huang, C.-H. Chen, and A. Tokuta, "Counting and classification of highway vehicles by regression analysis," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 5, pp. 2878–2888, 2015.

[11] L. Chen, *Research on Target Counting Method in Video Surveillance*, University of Science And Technology of China, Hefei, China, 2018.

[12] V. Lempitsky and A. Zisserman, "Learning to count objects in images," *Proceedings of Advances in Neural Information Processing Systems*, vol. 1, pp. 1324–1332, 2010.

[13] S. Zhang, G. Wu, P. Costeira, and J. M. F. Moura, "Understanding traffic density from large-scale web camera data," in *Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, Honolulu, HI, USA, July 2017.

[14] V. A. Sindagi and V. M. Patel, "A survey of recent advances in CNN-based single image crowd counting and density estimation," *Pattern Recognition Letters*, vol. 107, no. 5, pp. 3–16, 2018.

[15] Yu Liu and X. You-Lun, "Algorithm for searching nearestneighbor based on the bounded k-d tree," *Journal Of Huazhong University of Science and Technology (Nature Science Edition)*, vol. 36, no. 7, pp. 73–76, 2008.

[16] B. Sun, J.-H. Wang, D.-F. He, D. U. Hu-Bing, and B. Li, "Identification of aero-engine blade surface geometric defects with laser measurement," *Acta Automatica Sinica*, vol. 46, no. 3, pp. 594–599, 2020.