RESEARCH ARTICLE

Visual Feature Extraction and Tracking Method Based on Corner Flow Detection

Jiaxi Li[1], **Binbin Wang**[1], **Huijun Ma**[1,*], **Longfei Gao**[1] **and Heran Fu**[1]

[1] School of Computer Science and Artificial Intelligence, Beijing Technology and Business University, Beijing 100048, China

Abstract

Frontend feature tracking based on vision is the process in which a robot captures images of its surrounding environment using a camera while in motion. Each frame of the image is then analyzed to extract feature points, which are subsequently matched between pairwise frames to estimate the robot's pose changes by solving for the variations in these points. While feature matching methods that rely on descriptor-based approaches perform well in cases of significant lighting and texture variations, the addition of descriptors increases computational costs and introduces instability. Therefore, in this paper, a novel approach is proposed that combines sparse optical flow tracking with Shi-Tomasi corner detection, replacing the use of descriptors. This new method offers improved stability in situations of challenging lighting and texture variations while maintaining lower computational costs. Experimental results, validated using the OpenCV library on the Ubuntu operating system, provide evidence of the algorithm's effectiveness and efficiency.

Keywords: Computer vision, feature tracking, optical flow method, visual features, visual tracking.

Academic Editor:
Weimin Zhang

Submitted: 09 January 2024
Accepted: 10 May 2024
Published: 15 May 2024

Vol. 1, **No.** 1, 2024.
10.62762/TIS.2024.136895

***Corresponding author:**
Huijun Ma
mahuijun@th.btbu.edu.cn

Citation
Li, J., Wang, B., Ma, H., Gao, L., & Fu, H. (2024). Visual Feature Extraction and Tracking Method Based on Corner Flow Detection. IECE Transactions on Intelligent Systematics, 1(1), 3–9.

1 Introduction

Visual tracking technology stands as a prominent topic within the field of computer vision. It encompasses a suite of related technologies including the detection of moving objects from continuous image sequences, feature extraction, classification recognition, tracking filtering, and behavior recognition. These methodologies are employed to accurately determine the motion information parameters of a target—such as position, velocity, acceleration, and trajectory—and to perform the requisite processing and analysis for understanding the target's behavior. As information technology and intelligent science have advanced, computer vision has become a crucial component in robotics [1] and unmanned vehicles [2], marking it as one of the key technologies [13, 14] in this domain. Currently, visual tracking technology can be classified into two main types. The first type is based on the feature points within an image, involving the matching of these feature points across different images to ascertain the movement patterns of these points. The second method involves direct recognition of the target, followed by tracking its movement across different frames. This approach must surmount the challenges associated with video object recognition, necessitating the use of deep learning techniques. The framework for this method is complex and requires significant time to implement [3][4]. Consequently, the approach that starts by identifying feature points in the image offers substantial advantages for practical systems.

2 Related Works

In the field of computer vision, feature detection and matching are critical components. Initially, it is essential to extract features from the image and establish a corresponding relationship between images using these features. Here, a "feature" refers to a specific location in the image, typically comprising "point features," "line features," and other geometric features [5]. Point features are the most commonly utilized, also known as "key feature points", "points of interest" or "corner points".

Feature points can be divided into the following types:

1. The pixel corresponding to the local maximum of the first derivative (that is, the gradient of gray);

2. The intersection of two or more edges;

3. A point in an image where the rate of change of both the gradient value and the gradient direction is high;

4. At the corner, the first derivative is the largest and the second derivative is zero, indicating the direction of discontinuous changes in the edge of the object. Common methods for extracting corners include the Harris Corner [6] and the Shi-Tomasi Corner. The Shi-Tomasi corner extraction algorithm builds upon the foundation of the Harris Corner technique. The Harris Corner algorithm utilizes an autocorrelation matrix, specifically, a matrix M associated with the autocorrelation function. The eigenvalues of the M-matrix represent the first-order curvature of the correlation function. If both curvature values are high, the point is identified as a corner feature. Shi and Tomasi developed an improved method, which considers a point a strong corner if the smaller of the two eigenvalues exceeds a certain threshold. Compared to the Harris method, the Shi-Tomasi approach is generally more robust.

The effectiveness of corner detection and tracking significantly influences the performance of video tracking systems. Optical flow is a pivotal concept in motion detection, analyzing the movement of objects, surfaces, or edges relative to the observer. Optical flow is extensively used for motion detection [7] and tracking [8], making it one of the most prevalent methods currently in use. Many researchers have noted that using the optical flow method for extensive corner tracking and repeated calculations adds complexity, making it challenging to maintain real-time video processing [9]. Consequently, this paper proposes a hybrid approach that combines the corner method with the optical flow technique. It employs a sparse optical flow method to streamline the operations of the optical flow technique while still ensuring effective tracking of these corners.

3 Methodology

As one of the common corner detection methods in the field of machine vision, Shi-Tomasi corner detection method determines the corner position by calculating the characteristic response value of each pixel in the image. The main idea of the detection method is to select the pixel with the smallest characteristic response value as the corner. The feature response value is evaluated by calculating the feature value within the local region of the pixel. For each pixel, the special value can be calculated by calculating the gradient matrix of the pixels around it, such as the x and y components of the gradient. Then, the eigenvalues can be composed of a characteristic vector. As shown in Figure 1, during the corner selection process, a threshold can be set, and only pixels with feature response values greater than the threshold are selected as corner points. By adjusting the threshold, the number and quality of corners can be controlled. Let image(x, y) , at the point of(x, y)Translation at(u, v)After the self-similarity, you can use the gray change function$E(u, v)$Indicates that:

Figure 1. The situation of corner judgment

$$E(u, v) = \sum_{x,y} w(x, y)[I(x + u, y + v) - I(x, y)]^2 \quad (1)$$

Among$w(x, y)$as a window function,$I(x+u, y+v)$Is the pixel gray level after translation, expressed as Equation (2)

$$I(x + u, y + v) = I(x + y) + I_x u + I_y v + O(u^2, v^2) \quad (2)$$

Substituting Equation (2) into Equation (1) gives:

$$E(u,v)=[u,v]\begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix}\begin{bmatrix} u \\ v \end{bmatrix} \tag{3}$$

When the amount of local movement$[u,v]$Very small, Equation (3) can be approximated by Equation (4) as follows

$$E(u,v)\cong[u,v]M\begin{bmatrix} u \\ v \end{bmatrix} \tag{4}$$

Where M is a derivative of the graph.The matrix M is represented by Equation (5),

$$M=\sum_{x,y} w(x,y)\begin{bmatrix} I_x^2 & I_xI_y \\ I_xI_y & I_y^2 \end{bmatrix} \tag{5}$$

At this point,$E(u,v)$Can be expressed in elliptic form as shown in Figure 2: Let the response function of characteristic points R For:

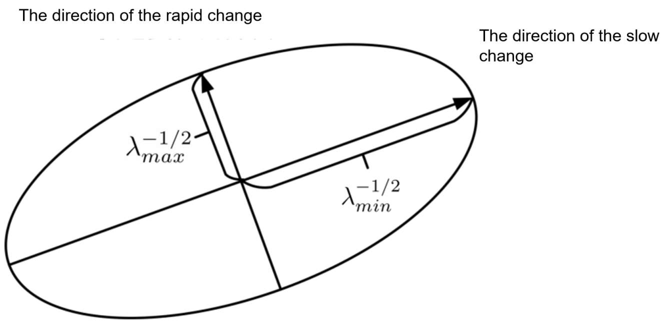

Figure 2. Representation of gray variation function

$$R=\lambda_1\lambda_2-k(\lambda_1+\lambda_2)^2 \tag{6}$$

As shown in Figure 2, the autocorrelation matrix M Can be decomposed into two eigenvalues(λ_0,λ_1).The characteristic vector corresponding to the eigenvector. And because smaller eigenvalues lead to greater uncertainty, that is,$a_0^{-1/2}$, so we can find good feature points by finding the maximum value of the smallest feature value. Based on the selected feature points, this paper introduces the method of completing square points: Kanade-Lucas-Tomasi, or KLT, KLT algorithm is a traditional sparse optical flow tracking method, which solves the offset to match the image. In the past decades, KLT algorithm has been widely used in various motion tracking examples.

KLT is an algorithm for tracking the same feature points in two consecutive images. The KLT algorithm has several assumptions: (1) the brightness of the image should be kept constant; (2) spatial consistency, that is, adjacent points have similar motions and remain adjacent; (3) The movement is continuous in time or the movement is "small movement." Usually these three points can be satisfied.

Set up I an dJ for two consecutive images in the image sequence,The grayscale value of a point (x,y) in the image is denoted as $I(x,y)$ and $J(x,y)$ Based on the assumptions of the KLT algorithm, set the coordinates of a feature point in the image I as u,the feature point correctly matched with u in image j is v,point v is obtained by moving point u a short distance d,so there are $v=u+d$,the point v at time $t+1$ can be obtained by moving d from u at time t.The purpose of the KLT algorithm is to solve for the change in displacement d.When solving KLT, it is equivalent to searching within a certain neighborhood around the feature points. Let u expand the window range by ω_x and ω_y,the window size is $(2\omega_x*2\omega_y)$.So finding d can be transformed into finding the minimum value of function $\varepsilon(d)$, as shown in Equation (7):

$$\varepsilon(d)=\sum_{x=u_x-\omega_x}^{x=u_x+\omega_x}\sum_{y=u_y-\omega_y}^{y=u_y+\omega_y}(I(x,y)-J(x+d_x,y+d_y))^2 \tag{7}$$

Equation (7), expressed in integral form, can be equivalent to:

$$\varepsilon(d)=\int\int(J(x+\frac{d}{2})-I(x-\frac{d}{2}))^2w(x)dx \tag{8}$$

The minimum value of $\varepsilon(d)$ can be obtained by solving the partial derivative of vector d to make it equal to 0, resulting in:

$$\frac{\partial_\varepsilon}{\partial_d}=2\int\int(J(x+\frac{d}{2})-I(x-\frac{d}{2}))(\frac{\partial J(x+\frac{d}{2})}{\partial d}-\frac{\partial J(x-\frac{d}{2})}{\partial d})w(x)dx \tag{9}$$

Equation (9) is expanded using the Taylor series:

$$J(\xi)\approx J(a)+(\xi_x-a)\frac{\partial J}{\partial x}(a)+(\xi_y-a_y)\frac{\partial J}{\partial y}(a) \tag{10}$$

According to Equation (10), expanding the Taylor formula on $J(x+\frac{d}{2})$ and $I(x-\frac{d}{2})$ yields:

$$\begin{aligned} J(x+\frac{d}{2}) &\approx J(x)+\frac{d_x}{2}\frac{\partial J}{\partial x}(x)+\frac{d_y}{2}\frac{\partial J}{\partial y}(x) \\ I(x+\frac{d}{2}) &\approx I(x)+\frac{d_x}{2}\frac{\partial I}{\partial x}(x)+\frac{d_y}{2}\frac{\partial I}{\partial y}(x) \end{aligned} \tag{11}$$

Substituting Equation (11) into Equation (9) gives:

$$\frac{\partial_\varepsilon}{\partial_d}\approx\int\int(J(x)-I(x)+g^Td)g(x)w(x)dx \tag{12}$$

Among them, the expression for g is:

$$g = [\frac{\partial}{\partial x}(\frac{I+J}{2})\frac{\partial}{\partial y}(\frac{I+J}{2})]^T \tag{13}$$

What ultimately needs to be solved is:

$$\frac{\partial_\varepsilon}{\partial_d} = \int\int (J(x) - I(x) + g^T d) g(x) w(x) dx = 0 \tag{14}$$

After opening it:

$$\int\int [J(x) - I(x)] g(x) w(x) dx = -\int\int g^T(x) d g(x) w(x) dx \tag{15}$$

After separately proposing d, it is concluded that:

$$-\int\int g^T(x) d g(x) w(x) dx = -[\int\int g(x) g^T(x) w(x) dx] d \tag{16}$$

Equation (16) can be simplified as:

$$Zd = e \tag{17}$$

where z and e are:

$$Z = \int\int g(x) g^T(x) w(x) dx$$

$$e = \int\int [I(x) - J(x)] g(x) w(x) dx \tag{18}$$

If we want d to have a solution, we must ensure that z is reversible. Equation (17) calculates the offset d iteratively, which can obtain a more accurate offset. The iterative method is:If the displacement obtained from the $(k-1st)$ iteration is $d^{k-1} = [d_x^{k-1} d_y^{k-1}]$, then $J(x,y)$, which is the $k-th$ iteration, can be represented by the $(k-1st)$ iteration as follows:

$$J(x,y) = J(x + d_x^{k-1}, y + d_y^{k-1}) \tag{19}$$

After multiple iterations of Equation (20), the $k-th$ displacement d^k can be obtained as follows

$$\varepsilon(d) = \sum_{x=u_x-w_x}^{u_x+w_x} \sum_{y=u_y-w_y}^{u_y+w_y} (I(x,y) - J(x + d_x^{k-1}, y + d_y^{k-1}))^2 \tag{20}$$

$$d^k = Z^{-1} e_k \tag{21}$$

We assume that after k iterations, Equation (20) converges, resulting in d:

$$d = \sum_{k=1}^{k} d^k \tag{22}$$

By solving for d, we have determined the matching relationship between the feature points on image I and image J, as well as the motion relationship between the two frames of images.

In the experiment, we achieved visualized optical flow tracking results as shown in Figure 3. This experiment is a still shot of a video, in front of the bus and car passing by, because the background is still, so there is no optical flow tracking trajectory on the static object, passing by the bus and car, red for feature points, green for each feature point tracking trajectory.

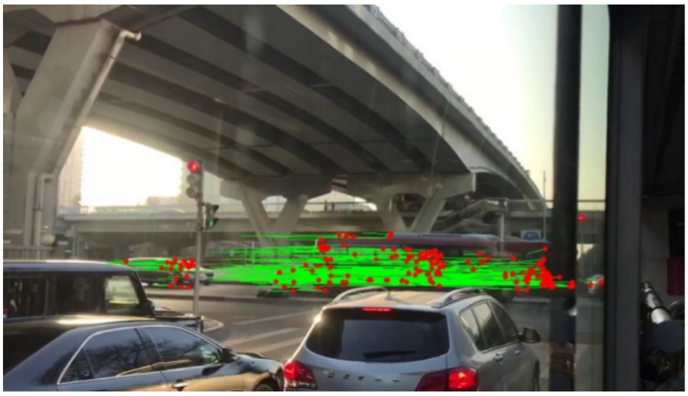

Figure 3. Visualization results of optical flow tracking

When the system is running, read in the video image processing process as follows:

1.Firstly, the first frame image is detected, and 200 Shi-Tomasi corners are extracted from the first frame image using the goodFeaturesToTrack function in Opencv. When extracting corners, the interval between corners is more than 30 pixels to ensure the uniform distribution of feature points. Then, each feature point is tagged.

2.The image frames after the first frame are tracked by KLT sparse optical flow, and the untracked feature points are eliminated.

3.The tracking between two frames also produces mismatched points. The outlier points are removed by 2 point-RANSAC method.

4.Then calculate the number of the remaining feature points k, and add one to these feature points records, that is, the number of successful tracing, add one, set the mask within the range of 30 pixels around the remaining feature points, no more feature points, so as to ensure the uniform distribution of the extracted features;

5.200-k Shi-Tomasi corners are extracted in the current frame, so that the total number of feature points is still 200;

6.Go to the next frame and repeat the cycle as above.

In the front-end feature tracking algorithm flow, when we extract feature points in the first frame, To ensure that the feature points interval is more than 30 pixels, in the process of feature tracking set mask, in the tracking of the feature points 30 pixels to add feature points, so as to ensure the uniform distribution of feature points in the image, for the subsequent image pose calculation. If this operation is not performed, the feature points will be unevenly distributed, as shown in Figure 4. After setting the mask, the image feature points are extracted as shown in Figure 5.

Figure 4. Image of extracting feature points without mask

Figure 5. Image of extracting feature points after mask is set

4 Experiments

To verify the effectiveness of the proposed method, we compared it with the method of completing the square [10]. The comparison involved conducting experiments on the same dataset to assess both methods in terms of processing time and translation error. The experiments were conducted using a Lenovo ThinkPad E470 computer equipped with an i5-7200U CPU at 2.50 GHz, 8 GB RAM, and running the Ubuntu 16.04 operating system, along with Kinetic for ROS. The experiment is carried out on a set of vehicle data collected from Kwun Tong Community, Chaoyang District, Beijing. The vehicle speed is from 0 km / h to 30 km / h.

The experimental results of video-based feature matching are shown in Figure 6, where red indicates the experimental results of the proposed method, and blue indicates the experimental results for traditional feature matching. Traditional feature matching [10] Compared with the method proposed in this paper, the experimental time consumption and translation error are obtained as shown in Figure 7, where (a) is the experimental time consumption comparison of the two methods, and the horizontal axis is the number of extracted feature points. The vertical axis is the time needed. It can be seen that with the increase of the number of extracted feature points, the calculation time of the two methods for feature extraction and data association is also increased, but the optical flow tracking method designed in this topic is less time-consuming than the feature matching method. The horizontal axis[11] represents the number of extracted feature points, and the vertical axis represents the translation error. It can be seen from Figure 8 that the error of the proposed method is less than that of the feature point matching method within 800 feature points. Experimental results show that compared with the feature point matching method[12], the proposed method is less time-consuming and has lower translation error.

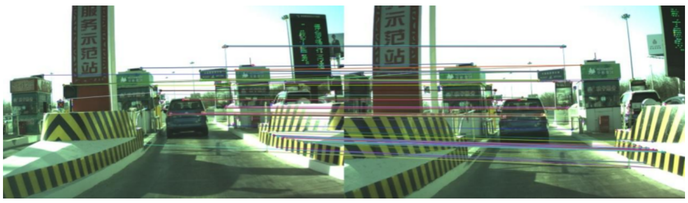

Figure 6. Result of feature matching

5 Conclusion

This paper aims to solve the problem of computational complexity and instability of feature point matching in visual tracking methods. Therefore, we propose a Shi-Tomasi method of completing square corners based on optical flow tracing.

Firstly, we use Shi-Tomasi corner detection algorithm to select the most representative corner as feature points.

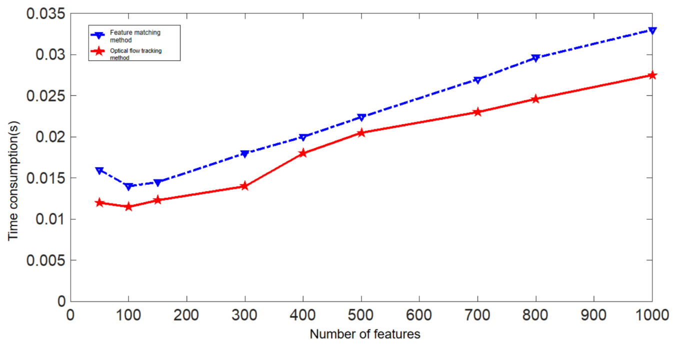

Figure 7. Performance comparison between the proposed optical flow tracking method and the traditional feature matching method(a)

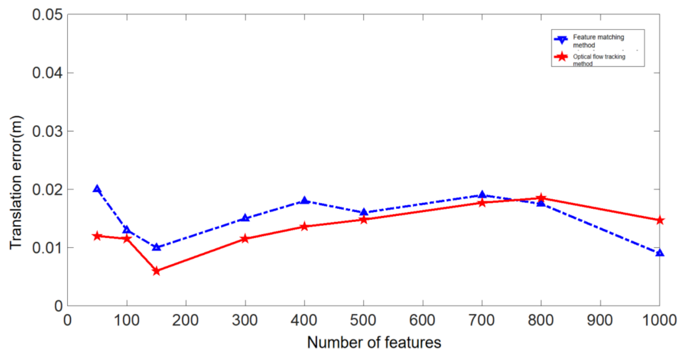

Figure 8. Performance comparison between the proposed optical flow tracking method and the traditional feature matching method(b)

This method can reduce the extraction of redundant information, reduce the computational complexity, and improve the accuracy and stability of feature points. Secondly, we optimize and improve the optical flow tracking process. In order to solve the problem of mismatching in optical flow tracking, we introduce a feature selection method based on optical flow. By filtering the motion of feature points, we eliminate the low-quality feature points, improve the accuracy of matching, and further enhance the stability of the tracking algorithm. Finally, we test the method and compare it with the traditional method of completing square. The results show that the Shi-Tomasi method of completing square corners based on optical flow tracing is accurate and stable.

To sum up, a novel Shi-Tomasi corner matching method based on optical flow tracing is proposed by improving the method of completing square points. Experimental results show that the proposed method is more stable in the case of large changes of illumination and texture, and can estimate the pose of the robot in real time.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgement

This work was supported in part by the National Natural Science Foundation of China No. 62173007, 62006008, 62203020, Project of Humanities and Social Sciences No. 22YJCZH006(Ministry of Education in China, MOC).

References

[1] Ke, X., Yu, Y., Li, K., Wang, T., Zhong, B., Wang, Z., & Wang, C. (2023). Review on robot-assisted polishing: Status and future trends. *Robotics and Computer-integrated manufacturing*, 80, 102482. [CrossRef]

[2] Guo, H., Lou, J., Yang, Z., & Xu, Y. (2023). Research on Multi Autonomous Vehicle Decentralization Strategy Based on Auction Multi Agent Deep Deterministic Strategy Gradient.*Journal of Electronics and Information Technology*, 45(7), 1-12.

[3] Gao, J., Yang, X., Zhang, T., & Xu, C. (2016). A Robust Visual Tracking Method Based on Deep Learning. *Journal of Computer Science*, 39(7), 16.

[4] Li, H., Bi ,D., Yang, Y., Cha, Y., Qin, B., & Zhang, L. (2015).Research on Visual Tracking Algorithm Based on Deep Feature Expression and Learning. *Journal of Electronics and Information Technology*, 37(9), 7.

[5] Li, H., Hu, Z., & Chen, X. (2017). Plp-slam:Visual SLAM method based on point, line, and surface feature fusion. *robot*(2), 8.

[6] Stephens, M., & Harris, C. (1989). 3D wire-frame integration from image sequences. *Image and Vision Computing*, 7(1), 24-30.

[7] Su, Z., He, Q., & Xie, Z. (2018). Detection of molten steel level based on optical flow analysis method. *Journal of Northeastern University: Natural Science Edition*, 39(2), 158-161.

[8] Jiang, L., & Cheng, G. (2015). Effective Target Point Enhancement Tracking Based on LK Optical Flow Tracking Method. *Microcomputers and Applications*, 34(6), 5.

[9] Huang, Z. (2014). Research on Human Behavior Recognition Based on Dense Optical Flow Trajectory. (Doctoral dissertation, Northeastern University).

[10] Hua, Y., Lin, J., & Lin, C. (2010, July). An improved SIFT feature matching algorithm. In *2010 8th World Congress on Intelligent Control and Automation* (pp. 6109-6113). IEEE. [CrossRef]

[11] Kumar, D., Pandey, R. C., & Mishra, A. K. (2024). A review of image features extraction techniques and their applications in image forensic. *Multimedia Tools and Applications*, 1-102. [CrossRef]

[12] Wang, J., Huang, W., & Biljecki, F. (2024). Learning visual features from figure-ground maps for urban morphology discovery. *Computers, Environment and Urban Systems*, 109, 102076.

[13] Cheng, L., Wang, Y., Liu, Q., Epema, D. H., Liu, C., Mao, Y., & Murphy, J. (2021). Network-aware locality scheduling for distributed data operators in data centers. *IEEE Transactions on Parallel and Distributed Systems*, 32(6), 1494-1510. [CrossRef]

[14] Liu, Q., Xia, T., Cheng, L., Van Eijk, M., Ozcelebi, T., & Mao, Y. (2021). Deep reinforcement learning for load-balancing aware network control in IoT edge systems. *IEEE Transactions on Parallel and Distributed Systems*, 33(6), 1491-1502. [CrossRef]

Jiaxi Li received the B.S. Automation from Beijing Technology and Business University, Beijing, China, in 2022. Currently pursuing a Master's degree in Control Engineering at Beijing Technology and Business University, main research interests focus on navigation and control of multi-modal object detection technology and applications.(Email: lijiaxi@st.btbu.edu.cn)

Binbin Wang received the M.S. Control engineering from Beijing University of Technology and Business, Beijing, China, in 2019.Main research interests focus on pattern recognition and information fusion, unmanned vehicles, machine learning, and other related fields. (Email: wangbinbin@st.btbu.edu.cn)

Huijun Ma received the M.S. Atomic and Molecular Physics from Changchun Institute of Optics and Mechanics , Changchun, China, in 2010. She is currently pursuing a Ph.D. in Systems Science from Beijing University of Technology and Business, with a research focus on complexity System modeling, pattern recognition and information fusion, machine learning, etc.(Email: huijunma@btbu.edu.cn)

Longfei Gao received the B.S. electrical engineering and automation from University of Jinan, Shandong, China, in 2022. Currently pursuing a Master's degree in Control Engineering at Beijing Technology and Business University, main research interests focus on image detection and pattern recognition. (Email: gaolongfei@st.btbu.edu.cn)

Heran Fu received the B.S. Automation from Beijing Technology and Business University, Beijing, China, in 2022. Currently pursuing a Master's degree in Control Engineering at Beijing Technology and Business University, main research interests focus on image classification and detection, pattern recognition, deep learning, among others. (Email: fuheran@st.btbu.edu.cn)