Research on Feature Extraction Method of UAV Video Image Based on Target Tracking

Xin Zhang, Zhi-jun Liu, and Ming-fei Qu(✉)

College of Mechatronic Engineering, Beijing Polytechnic, Beijing 100176, China
Jameszlx@163.com, qmf4528@163.com

Abstract. In order to extract the key and useful features of the target in the UAV video image and strong marking ability, a feature extraction method for the UAV video image based on target tracking is proposed. The sparse beam method is used to adjust the splicing of UAV video images. Based on this, the pixel coordinates are obtained through the frame difference method to detect and locate the target. According to the target detection and positioning results, the video image of the target area is selected and preprocessed by the wavelet transform algorithm Target area video image, and extract the target area video image feature, through hierarchical particle filtering to achieve target tracking, to achieve the extraction of UAV video image feature. The experimental results show that: in the ORL database experiment, the average feature extraction percentage is 78.08%, and the average target tracking error is 1.16; in the COIL-20 database experiment, the average feature extraction percentage is 82.55%, and the average target tracking error is 1.20, which meets the needs of UAV video image feature extraction and target tracking.

Keywords: Target tracking · Drone · Video image · Feature · Extraction

1 Introduction

As a new aerial remote sensing platform, UAV has the characteristics of fast sailing, flexible operation and low cost. The digital cameras and digital cameras mounted on the aircraft can obtain high-resolution video images, and the processing of video images can satisfy the vast majority. The needs of most users in aerial photography and target monitoring [1]. However, the video image acquired by UAV has the characteristics of high altitude and small image amplitude, which can not reflect the overall situation of the camera area. In emergency rescue, it is often unable to meet the needs and applications of information. Therefore, it is necessary to use video image feature extraction method to provide the target information of camera area. Using wireless transmission technology, the video images collected by UAV can be downloaded in real time. Video image has the characteristics of small frame, low resolution, large amount of data and high redundancy. When aiming at a certain target, we can't get the complete information of the target quickly. Therefore, the feature extraction of UAV video image has important practical significance and application requirements.

S. Liu and L. Xia (Eds.): ADHIP 2020, LNICST 348, pp. 266–278, 2021.
https://doi.org/10.1007/978-3-030-67874-6_25

From the existing research results, feature extraction is one of the most fundamental problems in the field of pattern recognition, and extracting effective screening features is a prerequisite for solving target tracking. The research of feature extraction has two main purposes: one is to find the most discriminative description between targets, so as to distinguish different types of targets; the other is to compress the dimension of target data under certain circumstances [2]. According to whether it can be linearly separable, the feature extraction method can be divided into two kinds: one is linear feature extraction method, the other is non-linear feature extraction method. Among them, linear feature extraction methods include principal component analysis, independent component analysis, factor analysis, local preserving projection, linear discriminant analysis, local feature analysis and multi-dimensional scale analysis. The linear feature extraction method is easy to understand and easy to implement, and has been successfully applied to many fields such as face recognition, character recognition, speech recognition and target classification. In order to extract the key and useful features of the target in the UAV video image and strong marking ability, a feature extraction method for the UAV video image based on target tracking is proposed.

2 Research on Feature Extraction Method of UAV Video Image

2.1 Drone Video Image Stitching

The time difference of UAV video image acquisition results in gaps between video images. In order to obtain complete video image information of high quality, the UAV video image is spliced based on sparse beam adjustment [3].

Sparse beam adjustment is a method of minimizing the error between the measured value and the estimated value of the matching point based on the Levenberg-Marquardt algorithm using the irrelevance of the projection matrix. In this method, the sparse structure of the normal equation is used to reduce the computational complexity, and the optimal solution of the normal equation is obtained quickly, so that the error between the measured value and the estimated value of the matching point pair is minimized.

The sparse beam method adjustment can be used to globally optimize the spliced UAV video image, and minimize the conversion error of each video image to the reference plane. The UAV video stream splicing belongs to sequence video image splicing. If the video image is globally optimized, it can only be done at the end of aerial photography. In order to obtain high-quality splicing video image in real time, this study uses the center constraint to dynamically select the reference plane, and ensures that the spliced video image has the local optimal characteristics each time the reference plane is changed. Finally, the middle frame of the flight belt is taken as the final reference plane to complete the splicing of UAV video stream [4]. The specific splicing steps are as follows:

Step 1: Using UAV trajectory planning data to obtain the approximate range of the shooting area, determine the longitude and latitude of the first and last video images of

each flight area, and convert them into geodetic coordinates to determine the geodetic coordinates of the middle position of the flight area.

Step 2: When the drone reaches the preset altitude and is flying at a constant speed, calculate the ground area of the video image during vertical shooting, and then roughly determine the video image to be extracted for a single flight strip according to the first and last geodetic coordinates of the flight strip and the given overlap Quantity n, recursively calculate $n/2$, and store the result of each calculation in vector v_1, then recursively calculate $(n/2+n)/2$, store the result of each calculation in vector v_r, and transform the reference plane before and after the intermediate frame as P_{li} and P_{ri};

Step 3: according to the camera parameters, altitude and flight speed, calculate the sampling interval of the key frame, extract the key frame and correct the video image, and cut the corrected video image into the corrected image with the shortest width as the image width and the width height ratio of 4:3;

Step 4: Perform sequence stitching on the cropped corrected images. Before the middle frame, when stitching to the reference plane P_{li}, optimize the absolute homography matrix of each video image to the $i-1$th reference video image and re-splice; when splicing to the middle frame video image $P_{n/2}$, the projection surface is fixed to the middle frame to continue splicing. When splicing to P_{ri}, the video image that has been spliced after the middle frame is optimized using the middle frame as the projection surface, so that each transformation is guaranteed The stitched video image before the reference plane is locally optimal and avoids the transfer of errors at both ends of the aircraft belt [5].

Through the above process, the UAV video image splicing is completed, which is ready for the following target detection and positioning.

2.2 Target Detection and Positioning

On the basis of the above video images, the pixel coordinates are obtained based on the frame difference method, and the target is detected and located.

Among the many target detection methods, considering the high speed of the frame difference method, this study selected this method as the basis of the pixel extraction algorithm. According to the result of the frame difference method, the update of the UAV video image is divided into pixel level and frame level. The former is used to detect small and slow changes; the latter is used to detect global and sudden changes. Different learning strategies and update speeds [6].

A dynamic feature matrix $D_{i,j}(t)$ is constructed to reflect the state of video image at time t, which is expressed as:

$$D_{i,j}(t)=\begin{cases} D_{i,j}(t-1)-1 & S_{i,j}(t)=0, D_{i,j}(t-1)\neq 0 \\ \lambda & S_{i,j}(t)\neq 0 \end{cases} \tag{1}$$

In formula (1), $S_{i,j}(t)$ represents a logical matrix; λ represents the pixel of frame λ.

The calculation formula of logical matrix $S_{i,j}(t)$ is:

$$S_{i,j}(t) = \begin{cases} 0 & \left|f_{i,j}(t) - f_{i,j}(t-\tau)\right| \le T_s \\ 1 & otherwise \end{cases} \tag{2}$$

In formula (2), $f_{i,j}(t)$ represents the gray value of pixel $p(i,j)$ at time t; τ represents the interval; T_s represents the threshold.

If the values of logic matrix $S_{i,j}(t)$ and continuous λ frame are all 0, it means that the gray value of corresponding pixel has little change in continuous λ frame. It is considered that there is no target or noise at this point in this period of time. Therefore, it can be considered that the gray value is the background gray value with great possibility. The gray value can be used to update the corresponding background point. The updating formula is:

$$B_{i,j}(t) = \alpha \cdot f_{i,j}(t) + (1-\alpha) \cdot B_{i,j}(t-1) \tag{3}$$

In Eq. (3), $B_{i,j}(t)$ represents the background point update value; α represents the update coefficient.

If the value of logical matrix $S_{i,j}(t)$ is 1 or it can't guarantee that the continuous frame λ is 0, the gray value of the corresponding point is considered to be unstable, and the gray value of the point should not be collected. According to the above analysis, matrix $D_{i,j}(t)$ stores the changes of the corresponding pixels in the video image space at time t. if the value is 0, it belongs to the pixels that have not changed the gray value of the continuous λ frame; if the value is not 0, it belongs to the pixels that have changed in the continuous λ frame. The larger the value, the closer the change is to the current frame [7].

The difference between the real-time gray-scale matrix $f_{i,j}(t)$ and the background matrix $B_{i,j}(t)$ and the thresholding will obtain the binarized video image data, which is expressed as:

$$F_{i,j}(t) = \begin{cases} 1 & \left|f_{i,j}(t) - B_{i,j}(t-\tau)\right| \le T_F \\ 0 & otherwise \end{cases} \tag{4}$$

In Eq. (4), T_F represents the threshold parameter.

Binary video image data a usually contains $F_{i,j}(t)$ lot of noise - unreal target area, which is called false detection target, needs to be filtered. Because the specific location of the target can not be determined, the convolution of filtering is carried out in the whole video image space. Excessive filtering may cause the target area to be destroyed by the convolution operation and the serious blurring of the edges. Even if a target splits into multiple small targets or multiple targets merge into a target phenomenon, these results will be It causes difficulties for follow-up [8]. Therefore, the goal of filtering should not be to eliminate all noises, but to protect the target from being eroded or blurred as much as possible while filtering. Therefore, this study adopts a relatively conservative filtering process. The main steps are as follows: in order to reduce the amount of calculation, $F_{i,j}(t)$ is sub sampled twice; after the sub sampled video image is processed by morphological filtering once, the video image matrix $I(t)$

is obtained, and the pixel coordinates with the value of 1 are extracted to form a data set composed of n pixel coordinates: $X(t) = \{x_1(t), x_2(t), \cdots, x_n(t)\}$.

Calculate the initial clustering of $X(t)$, and merge the clusters repeatedly under the guidance of the criterion function J_{nn} until the criterion function is the smallest. The target is calibrated according to the final stable clusters. The calibration position used is the cluster centroid coordinate, which completes the detection and positioning of the target.

2.3 Feature Extraction of Video Image in Target Area

Based on the above target detection and positioning results, the target region video image is selected, the target region video image is preprocessed by the wavelet transform algorithm, and the target region video image feature is extracted.

Wavelet transform is widely used in video image processing because of its good properties of multi-scale and multi-resolution. In video image processing, wavelet transform has perfect reconstruction power, that is, it will not lose the original information, nor generate redundant information. Wavelet transform can easily obtain the frame information and detailed information of the original video image after wavelet decomposition. After decomposing the video image, the two-dimensional wavelet transform will produce four sub bands, namely *LL*, *LH*, *HL* and *HH*. Among them, *LL* represents horizontal and vertical low frequency information; *LH* represents horizontal low frequency and vertical high frequency information; *HL* represents horizontal high frequency and vertical low frequency information; *HH* represents horizontal high frequency and vertical high frequency information. The low frequency part corresponds to the average gray level in a video image, and reflects the smooth part of the video image. The high level part is the gray level which changes faster and faster in the video image, corresponding to the edge, detail and noise in the video image. Therefore, the processing of the low frequency part of the video image will not lose or change the details and edge information of the video image, nor will it change the noise in the video image [9, 10].

The steps of video image preprocessing based on wavelet transform algorithm are as follows:

Step 1: Acquire the video image of the target area, record it as $f(x, y)$, and use db8 wavelet to decompose $f(x, y)$;

Step 2: In the four sub bands *LL*, *LH*, *HL* and *HH*, *LL* sub bands are selected for histogram equalization;

Step 3: Set an appropriate threshold and use wavelet threshold denoising technology to suppress noise in high-frequency video images;

Step 4: The processed *LL* low-frequency video image and other high-frequency video image are reconstructed by wavelet, and the video image $f_1(x, y)$ is obtained;

Step 5: Then perform histogram equalization processing on the video image $f_1(x, y)$ to obtain the final video image $g(x, y)$.

Based on the video image $g(x, y)$ of the target area obtained in the above steps, Vectorize it to form a vector χ of $mn \times 1$ connected end to end, as shown in Fig. 1.

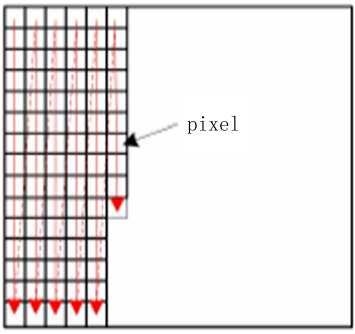


Fig. 1. Vectorization of video image in target area

Then the average vector of M target area video image is:

$$\mu = \frac{1}{M}\sum_{i=1}^{M}\chi_i \tag{5}$$

Then the covariance matrix of the video image in the target area is expressed as:

$$C = \frac{1}{M}\sum_{i=1}^{M}(\chi_i - \mu)(\chi_i - \mu)^T = \frac{1}{M}XX^T \tag{6}$$

In formula (6), XX^T is the simplified formulation of $\sum_{i=1}^{M}(\chi_i - \mu)(\chi_i - \mu)^T$. by solving formula (6), the eigenvalues and eigenvectors of video images in the target area can be obtained.

The feature vectors are sorted according to the size of the feature value. The larger the feature value is, the better the feature vector can reflect the video image features of the target area, and the size of the feature value decreases exponentially [11, 12]. The video image of the target area corresponding to the feature vector is called a feature sub-video image, that is, a feature face. The more blurred the feature face, the less information it contains. The feature sub video image is used to reconstruct the video image in the target area. It can be seen that a large amount of information of the original video image in the target area can be recovered with fewer features. Then the feature vector obtained is representative and recorded as $X\eta$.

2.4 Feature-Based Target Tracking

Based on the video image feature vector of the target area obtained above, the target tracking is realized by hierarchical particle filter.

Target tracking can be understood as adaptively "dragging" the window as the target moves, so that the tracked target is always in the window, and the size and angle of the window are adjusted in real time according to the size and posture of the target. In the video image coordinate of UAV, the target tracking window is shown in Fig. 2.

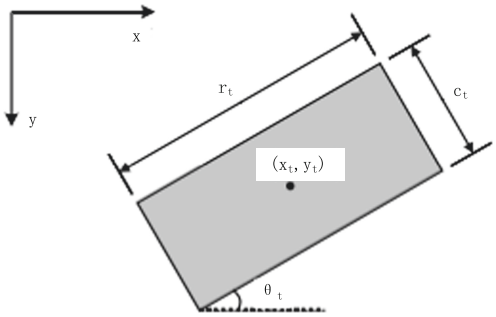


Fig. 2. Schematic diagram of target tracking window

As shown in Fig. 2, (x_t, y_t) represents the target coordinate; r_t represents the length of the window; c_t represents the width of the window; θ_t represents the angle between the tracking window and the coordinate axis.

The steps of target tracking algorithm based on hierarchical particle filtering are as follows:

Step 1: set the tracking window manually, select the target to be tracked, take the center of the window as the initial position of the target, the angle between the long axis of the window and the axis as the initial direction of the target, and the size of the window as the initial size of the target. Based on this, the initial value of the target state can be determined as $s_0 = [x_0, y_0, r_0, c_0, \theta_0]^T$. And extract the video image in the window, and use the process of Sect. 2.3 to find the feature vector;

Step 2: initialization of particle filter. The initial state of the particle filter is obtained by initializing N position particle with the position information in the initial value vector of the target state: $\{x_0^i, 1/N_s\}_{i=1}^{N_s}$;

Step 3: Importance sampling. At time $t > 0$, N_s particles are re-sampled according to the particle group of the previous two frames and the target state transition model to obtain the predicted value of the filter: $\{x_t^i, 1/N_s\}_{i=1}^{N_s}$;

Step 4: Similarity measurement. Calculate freely to the left according to the derivation process of the hierarchical particle structure, and finally get the target state $s_t^k = [x_t^k, y_t^k, r_t^k, c_t^k, \theta_t^k]^T$ corresponding to the k th first-layer particle.

Step 5: target state estimation and target vector update. Firstly, the state value of the target at the moment is estimated according to the state value of N_s target: $E(s_t) = \frac{1}{N_s}\sum_{i=1}^{N_s} s_t^i \cdot \omega_t^i$, where $s_t^i \cdot \omega_t^i$ represents the state value of the first layer particle; then, the corresponding video image is grabbed according to the target area corresponding to $E(s_t)$ and the feature vector $p_{E(s_t)}$ is calculated, and the similarity between it and the original target is compared, and the target vector is updated or not according to this. If it is updated, it will be used as a new target prototype feature;

Step 6: Determine whether the target disappears. If it does not disappear, return to step three; otherwise, exit.

Through the above process, the feature extraction of UAV video image based on target tracking is realized, which helps the application of UAV.

3 Performance Analysis of Feature Extraction Method for UAV Video Image

This research will verify the effectiveness of the UAV video image feature extraction method based on target tracking on ORL database and COIL-20 database. In the experiment, two random partition methods are selected for each database video image set to verify the feasibility of the proposed method. In the experiment, the nearest neighbor classifier is used. The experimental software environment is matlab 7.0a. All the experimental results are executed ten times Then take the average.

3.1 Simulation Experiment on ORL Database

The information of ORL database is not described in detail due to space limitation. In this experiment, all UAV video images are cropped to the size of 64 × 64 dimensions, and then each video image is divided into 4 blocks, row and column are divided into 16 blocks in total, then the size of each sub block video image is 16 × 16.

In this experiment, the ORL database is used as the input UAV video image set, and 10 video images belonging to the same target are used as a class. For the first time, 5 video images in the database are randomly selected as training video images, and the remaining 5 video images as a test video image, the training video image set has 200 images, and the test video image set has 200 images, referred to as 5train for short. In the second random extraction of 3 video images in the database as training video images and the remaining 7 video images as test video images, the training video image set has 120 images and the test video image set has 280 images, referred to as 3train for short.

Through experiments, the percentage of feature extraction of the proposed method is shown in Table 1.

Table 1. Feature extraction percentage analysis

Number of goals	5train	3train
1	85.10%	85.46%
2	86.44%	80.12%
3	82.13%	82.25%
4	85.01%	80.00%
5	86.66%	79.45%
6	85.03%	78.11%
7	85.55%	76.23%
8	85.00%	75.00%
9	85.49%	75.00%
10	85.11%	74.22%
Average value	78.08%	

It can be seen from Table 1 that at 5train, the proposed method is more robust, and the feature extraction percentage remains almost unchanged, while at 3train, as the number of targets increases, the feature extraction percentage decreases rapidly. Through calculation, the average feature extraction percentage is 78.08%, which satisfies the requirements for feature extraction of UAV video images.

The experimental results of target tracking error are shown in Table 2.

Table 2. Target tracking error analysis

Number of experiments	5train	3train
10	1.2	0.7
20	1.1	1.0
30	1.0	1.1
40	0.9	1.1
50	1.0	1.2
60	1.1	1.3
70	1.2	1.5
80	1.0	1.6
90	1.0	1.6
100	0.8	1.8
Average value	1.16	

As shown in the data in Table 2, at 5train, the proposed method error is small and almost unchanged; at 3train, the proposed method error gradually increases. The average value of target tracking error is 1.16, which meets the requirements of target tracking.

3.2 Simulation Experiment on COIL-20 Database

Due to space limitation, the information of COIL-20 database is not described in detail [13]. In this experiment, in COIL-20 database, we select 24 video images under 15O rotation of UAV, and there are 480 video images in total. If the image size is 64 × 64, the video image is divided into 16 blocks, and the row and column are all 4 blocks, then the video image size of each sub block is 16 × 16, so as to prepare for the experiment.

The COIL-20 database is used as the input video image set, and 24 video images belonging to the same target are used as a category. For the first time, 12 video images in the database are randomly selected as training video images, and the remaining 12 video images are used as test videos. For video, the training video image set has 240 images, and the test video image has 240 images, referred to as 12train. In the second random extraction of 9 images as training video images, and the remaining 15 video images as test video images, the training video image set has 180 images, and the test video image set has 300 images, referred to as 9train.

In the case that the dependent variable is the number of targets, the feature extraction percentage of the proposed method is obtained through the simulation comparison experiment, as shown in Table 3.

Table 3. Feature extraction percentage analysis

Target quantity	12train	9train
1	84.12%	85.02%
2	84.00%	83.12%
3	84.02%	81.00%
4	84.13%	80.94%
5	84.44%	80.50%
6	85.00%	80.44%
7	85.01%	80.00%
8	84.49%	79.45%
9	83.39%	78.51%
10	85.46%	78.00%
Average value	82.55%	

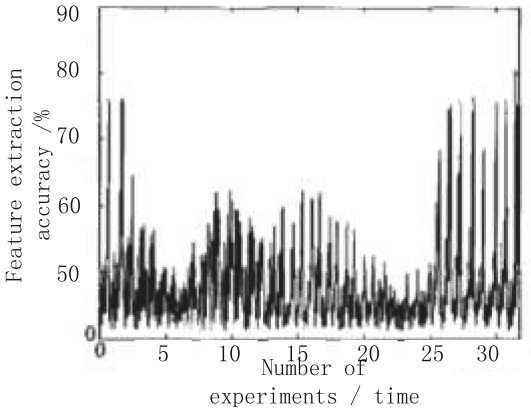
It can be seen from Table 3 that the proposed method is relatively robust at 12 train, and the feature extraction percentage almost remains unchanged, while at 9 train, with the increase of the number of targets, the feature extraction percentage drops rapidly. The average percentage of feature extraction is 82.55%, which meets the requirements of UAV video image feature extraction.

The experimental results of target tracking error are shown in Table 2.

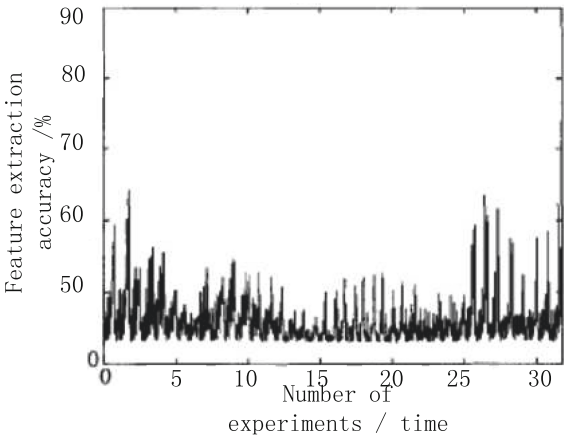
As shown in Table 4, at 12train, the proposed method error is small and almost unchanged; at 9train, the proposed method error gradually increases. Through calculation, the average value of target tracking error is 1.20, which meets the requirements of target tracking.

Table 4. Target tracking error analysis

Experiments	12train	9train
10	1.0	1.2
20	1.1	1.1
30	0.8	1.1
40	0.9	1.6
50	1.0	1.2
60	0.9	1.4
70	0.9	1.5
80	1.0	1.9
90	1.0	1.6
100	0.8	2.0
Average value	1.20	



(a) The accuracy of image feature extraction based on this method



(b) Accuracy of image feature extraction based on traditional methods

Fig. 3. Comparison of video image feature extraction accuracy of UAV

In order to further verify the effectiveness of this method, the proposed method and the traditional method of UAV video image feature extraction accuracy are compared and analyzed, the comparison results are shown in Fig. 3.

According to Fig. 3, the accuracy of UAV video image feature extraction in this paper is up to 80%, while that of traditional method is only 65%. The accuracy of UAV video image feature extraction in this paper is higher than that of traditional method.

4 Empirical Conclusion

This paper proposes a feature extraction method for UAV video images based on target tracking, and the experimental results show that: under the ORL database experiment, the average feature extraction percentage is 78.08%, and the average target tracking error is 1.16; under the COIL-20 database experiment, the average feature extraction percentage is 82.55%, and the average target tracking error is 1.20, which can meet the needs of UAV video image feature extraction and target tracking.

Acknowledgements. The application of UAV spray in the city pest control service (2019H033-KQ)

References

1. Chen, Y., Liu, J., Pei, J., et al.: The risk factors that can increase possibility of mandibular canal wall damage in adult: a cone-beam computed tomography (CBCT) study in a Chinese population. Med. Sci. Monitor **24**(2), 26–36 (2018)
2. Azimi, S.M., Britz, D., Engstler, M., et al.: Advanced steel microstructural classification by deep learning methods. Sci. Rep. **8**(1), 2128 (2018)
3. Ballerini, L., Lovreglio, R., Valdés Hernández, M.C., et al.: Perivascular spaces segmentation in brain MRI using optimal 3D filtering. Sci. Rep. **8**(1), 2132–2132 (2018)
4. Lewis, A.G., Schriefers, H., Bastiaansen, M., et al.: Assessing the utility of frequency tagging for tracking memory-based reactivation of word representations. Sci. Rep. **8**(1), 7897 (2018)
5. Liu, S., Liu, D., Srivastava, G., et al.: Overview and methods of correlation filter algorithms in object tracking. Complex Intell. Syst. (2020). https://doi.org/10.1007/s40747-020-00161-4
6. Hilchey, M.D., Rajsic, J., Huffman, G., et al.: Dissociating orienting biases from integration effects with eye movements. Psychol. Sci. **29**(3), 328–339 (2018)
7. Jiang, H., Qu, P., Wang, J.W., et al.: Effect of NF-κB inhibitor on Toll-like receptor 4 expression in left ventricular myocardium in two-kidney-one-clip hypertensive rats. Eur. Rev. Med. Pharmacol. Sci. **22**(10), 3224–3233 (2018)
8. Rina, L.D.N., Walburg, K.V., Martin, V.H.P., et al.: Retinal pigment epithelial cells control early mycobacterium tuberculosis infection via interferon signaling. Investigative Opthalmol. Vis. Sci. **59**(3), 1384–1395 (2018)

9. Harris, H., Sagi, D.: Visual learning with reduced adaptation is eccentricity-specific. Sci. Rep. **8**(1), 608 (2018)
10. Liu, S., Liu, G., Zhou, H.: A robust parallel object tracking method for illumination variations. Mob. Netw. Appl. **24**(1), 5–17 (2019)
11. Liu, S., Lu, M., Li, H., et al.: Prediction of gene expression patterns with generalized linear regression model. Front. Genet. **10**, 120 (2019)
12. Barrea, A., Delhaye, B.P., Lefèvre, P., et al.: Perception of partial slips under tangential loading of the fingertip. Sci. Rep. **8**(1), 7032–7032 (2018)
13. Lu, M., Liu, S.: Nucleosome positioning based on generalized relative entropy. Soft. Comput. **23**(19), 9175–9188 (2018). https://doi.org/10.1007/s00500-018-3602-2