

Tracking Algorithm with Adaptive Bandwidth of Kernel Function for UAV Autonomous Aerial Refueling

ZhuoYa Wang¹, Yaohong Qu^{1,*}, Jiaju Wu², Bing Xiao¹ and Jianguo Yan¹

¹Northwestern Polytechnical University, Xi'an, shaanxi,China,710129

²AVIC The First Aircraft Institute, Xi'an, shaanxi, China, 710089

Abstract. The use of machine vision to track the receiver aircraft's receptacle is one of the key steps in automated aerial refueling. Traditional Meanshift tracking algorithm's kernel bandwidth is fixed, when the target size changes, it will produce with the bias even with the loss of the situation. Aiming at this problem, this paper proposes a kernel bandwidth adaptive algorithm based on corner matching. First, use backward tracking method to amend the core window center. Then according to the Harris algorithm to detect the corner, use small-scale approach for the two points between frames to match and regression analysis, namely determine the update parameters of nuclear window width. Experiments show that the algorithm can track the target accurately when the size of tracking target change, and the kernel window width can be better suited to the target size.

Keywords. Aerial refueling, Meanshift, Kernel bandwidth adaptive algorithm, Corner matching

1 Introduction

UAV (unmanned aerial vehicles) play an increasingly important role in the field of civilian and military fields because of its small size, low cost, easy to use, etc. However, weak load capacity, short life time and other issues limit the use of UAV. UAV with independent air refueling capabilities has a large flight distance and a long blank time, expanding the use of UAV.

One of the key technologies of autonomous air refueling is real-time access to the location of the receiver aircraft's receptacle. Traditional UAV navigation system is generally based on GPS, INS and integrated navigation system, which does not meet the location accuracy requirements. In recent years, the development of machine vision provides an effective support for the UAV autonomous air refueling. Introduce the machine vision into the autonomous air refueling process can get more accurate location information. In many machine vision tracking methods, many scholars have studied Meanshift because it is an efficient pattern matching algorithm with the advantages of high robustness, real-time and easy realization. Document [3] use RGB linear fusion and add Kalman Filter algorithm to estimate the location of moving conditions, which can track the target accurately when the size of target is constant. When the tanker closes to receiver aircraft's receptacle, target will gradually become larger in the tracking video. Then the tracking is not accurate because the method kernel

function bandwidth is fixed. Document [4] uses the method of comparing Bhattacharyya coefficients to change the bandwidth of kernel function. However, this method needs to do multiple times of Meanshift algorithm with different kernel bandwidth. The calculation is large, and the search box is proportional change, which cannot adapt to the target changes. Document [5] uses positive and negative 10% increments to change the kernel function bandwidth. It needs three times of Meanshift tracking based on three kernel bandwidth in the tracking of each frame. It will get a good tracking effect when the target is smaller, not suitable for autonomous air refueling process by the receiver aircraft's receptacle tracking. Document [6] changes the kernel bandwidth by the backtracking method based on the corner feature, but the tracking is not accurate enough only when the target feature is described.

Aiming at this problem, this paper proposes a kernel bandwidth adaptive algorithm based on corner matching. Use backward tracking method to amend the core window center and make the tracking window aim at the target heart. According to the Harris algorithm to detect the corner, use small-scale approach for the two points between frames to match, which reduce the complexity of corner matching. Obtain the affine parameters based on the result of corner matching and modify the kernel function. In addition, when using the Meanshift tracking algorithm, add spatial information and color information

* Corresponding author: qyh0809@126.com

to describe the target model comprehensively, and ensure the tracking effect is more accurate.

2 Meanshift Algorithm Theory

Figures and tables, as originals of good quality and well contrasted, are to be in their final form, ready for reproduction, pasted in the appropriate place in the text. Try to ensure that the size of the text in your figures is approximately the same size as the main text (10 point). Try to ensure that lines are no thinner than 0.25 point.

Given n number sample points x_i in a d -dimensional space R^d . Meanshift vector at the point of x is defined as:

$$M_h(x) = \frac{1}{k} \sum_{x_i \in S_h} (x_i - x) \quad (1)$$

Where, k represents k number points in n number sample points fall in the selection area. x_i is the pixel, x is the initial position, S_h is a high-dimensional ball with the radius h , which is the set point of y satisfying the relationship.

$$S_h = \{y : (y-x)^T (y-x) \leq h^2\} \quad (2)$$

It can be seen $(x_i - x)$ is the shift vector of sample x respect to the initial position. Figure 1 is Meanshift iteration process. The average offset vector $M_h(x)$ will point to most sample distribution area, that is, the gradient direction of the probability density function. Meanshift algorithm will eventually converge to the actual location of the target through constant iteration.

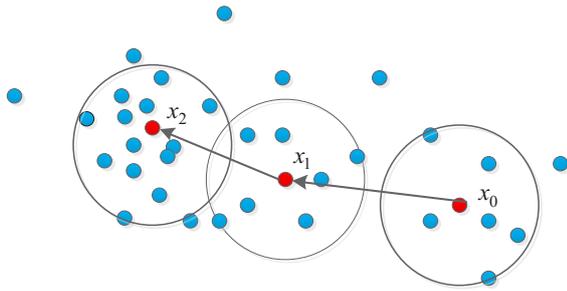


Fig. 1. Meanshift iteration process

3 Meanshift Tracking Algorithm with Adaptive Bandwidth

3.1. Target model and similarity measure

Add the spatial location information of the pixel, and synthesize the color feature and the spatial feature to describe the target template. According to the distance between each pixel and the reference point, the kernel function is used to assign different weights to different pixels, and the weighted color histogram is constructed.

The weight of each pixel is calculated by the following kernel function:

$$K(x) = 1 - \frac{(x_i - x_{mid})^2 + (y_i - y_{mid})^2}{(2 * x_{mid})^2 + (2 * y_{mid})^2} \quad (3)$$

Where, x_{mid} 、 y_{mid} are the x-axis and y-axis center coordinate of the selected graphic template. x_i, y_i , respectively, are horizontal and vertical coordinates of point x . $(2 * x_{mid})^2 + (2 * y_{mid})^2$ is the square of the radius of the kernel function. When point x is at the center of the rectangular template, $K(x)$ is 1. When point x is at the border of the rectangular template, $K(x)$ is 3/4. The more the distance from center pixel template is, the smaller the value is.

The probability density of the target template is estimated as:

$$q_u = C \sum_{i=1}^M K \left(\left\| \frac{x - x_i}{h} \right\| \right) \delta [b(x_i) - u] \quad (4)$$

Where, q_u is the probability density of the target template histogram, h is kernel function bandwidth, $\delta [b(x_i) - u]$ is histogram index function, M is the number of pixels in a given space, x_i is any pixel in the template, C is normalization coefficient.

Similarly, the probability density of the sample template with center point x^j is estimated as:

$$p_u(x^j) = C \sum_{i=1}^M K \left(\left\| \frac{x^j - x_i}{h} \right\|^2 \right) \delta [b(x_i) - u] \quad (5)$$

Introduce BH coefficients to describe the similarity between the target template and the sample template:

$$BH [p^j, q] = \sum_{u=1}^m \sqrt{p_u^j(x^j) q_u} \quad (6)$$

Where q_u is the probability density histogram of the target template, x^j is the j th particle of the particle set, $p_u^j(x^j)$ is the probability density histogram of the sample template centered on x^j , and m is the number of color histograms u . The higher the value of BH , the higher the similarity between the target template and the sample template is, and vice versa.

3.2 Kernel function bandwidth correction

3.2.1 Affine model

Affine transformation is a basic transformation in affine geometry, and the affine transformation of two-dimensional continental space can be expressed as:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} a_x & a_{xy} \\ a_{yx} & a_y \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} l_x \\ l_y \end{bmatrix} \quad (7)$$

Where $(x_i, y_i)^T$ and $(x_{i+1}, y_{i+1})^T$ are the position coordinates of any feature point in the i frame and the $i+1$ frame, respectively. $(l_x, l_y)^T$ is the translation

vector, $\begin{bmatrix} a_x & a_{xy} \\ a_{yx} & a_y \end{bmatrix}$ is a synthetic matrix for rotation.

Vector $a = [a_x, a_{xy}, a_{yx}, a_y, l_x, l_y]$ determines the coordinate transformation between two feature points.

Here only consider the most common transformations in the air refueling: translation and expansion. Make $a_{xy} = a_{yx} = 0$, the affine model is defined as:

$$\begin{bmatrix} x_{i+1} \\ y_{i+1} \end{bmatrix} = \begin{bmatrix} a_x & 0 \\ 0 & a_y \end{bmatrix} \begin{bmatrix} x_i \\ y_i \end{bmatrix} + \begin{bmatrix} l_x \\ l_y \end{bmatrix} \quad (8)$$

Then the kernel function bandwidth can be modified according to Eq. (9):

$$h_{i+1} = h_i \cdot \sqrt{a_x^2 + a_y^2} \quad (9)$$

Where, h_i and h_{i+1} are the kernel function bandwidth of the i frame and $i+1$ frame respectively.

3.2.2 Kernel window centre registration

Set the frame i of the tracking window (namely kernel window) as T_i , the center of the window is C_i , the target heart is O_i . It is known from the literature [6] that when the target becomes smaller, can obtain the best match window through the Meanshift iteration, as long as the expansion and contraction of target in the frame $i+1$ are in the T_i range. That is, the center C_{i+1} of the tracking window will always lock the target's centroid. In most cases, Meanshift tracking algorithm can lock the target centroid as long as most of the region of the target is within the range of T_i . When the target gradually increases, it will produce positioning bias, and then need to register the center of the window registration.

When the target is gradually increased, if the $i+1$ frame target is larger than the tracking window range, the tracking window T_{i+1} contains only part of the target, the current total target centroid O_{i+1} and the tracking window center C'_{i+1} that has not been registered at this time will have a certain offset.

$$d = C'_{i+1} - O_{i+1} \quad (10)$$

At this point the tracking window T_{i+1} contains only part of the target; the corresponding tracking window center C'_{i+1} is actually the center O'_{i+1} of the target. In order to register the center of the window to find the

current target heart, need to compensate for the offset d . As mentioned above, when the target is gradually smaller, there is no deviation between the tracking window and the centroid. Therefore, use the backward tracking, and the part of the tracking window of the $i+1$ frame is used as the new target for Meanshift tracking. Get the tracking window T'_i , center C'_i of T'_i is the part of the target in the i frame of the centroid, and then the centroid deviation of the target and the total target in the i frame part is:

$$d' = C_i - C'_i \quad (11)$$

Since the interval between two adjacent frames is very short, so use d' to make up the offset d , the $i+1$ frame track window after the registration is estimated as follows:

$$C_{i+1} = d' + C'_{i+1} \quad (12)$$

3.2.3 Corner detection and matching

Corner detection of the target template, match the corner points between the two frames by a small range of fringes, and the affine model parameters are estimated according to the corner matching results.

Use Harris algorithm to detect corner [8-9], judge whether a point is a corner point by analyzing the autocorrelation function characteristic matrix of the surrounding area of the pixel, the autocorrelation function is:

$$E(u, v) = \sum_{x, y} w(x, y) [I(x, y) - I(x+u, y+v)]^2 \quad (13)$$

Where u, v are small offset in the x, y direction, $I(x, y)$ is the gray value at (x, y) pixel, $w(x, y)$ is filter, the autocorrelation function for Taylor expansion can be obtained from the correlation matrix:

$$M = \sum_{x, y} w(x, y) \otimes \begin{bmatrix} I_x^2 & I_x I_y \\ I_y I_x & I_y^2 \end{bmatrix} \quad (14)$$

Corner response function is:

$$R = |M| - k \cdot \text{tr}^2(M) \quad (15)$$

Among them, the value of scale factor k is generally 0.04-0.06, and then can use the value of R to determine whether the pixel point is corner. When absolute value of R is smaller, it represents a flat area. R is negative when it represents boundary and positive when it is corner. Roughly suppress R to retain more than the threshold T and the local maximum point as a corner point.

Use corner match after corner point detection. Set the tracking window T_i of i frame has N corners; the tracking window T_{i+1} in $i+1$ frame after center matching has N' corners. T_i is same as T_{i+1} in size. Center of the tracking window is consistent with the target centroid, with using small-scale approach for

corner matching. For any corner P_i^j in T_i , find the corresponding corner in the $i+1$ frame. First give a $5 * 5$ small window in the $i+1$ frame with the center of corner P_i^j , find the brightness corner close to P_i^j . Then the point is the corresponding corner in $i+1$ frame, regression analysis the corner coordinates to obtain a_x and a_y , complete the kernel function of the bandwidth correction.

4 Tracking Algorithm

Use backward tracking method to amend the core window center. According to the Harris algorithm to detect the corner, use small-scale approach for the two points between frames to match. The affine parameters are obtained based on the result of corner matching and the kernel function is modified. The algorithm is as follows:

Step 1: Initialization: manually select the target area and determine the target template histogram.

Step 2: Based on the backward tracking method, according to the formula (11) to calculate the error compensation value d' , according to formula (12) to calculate the window centre C_{i+1} of modified $i+1$ frame.

Step 3: According to formula (15) to calculate the corner points in the tracking window of the i frame and the $i+1$ frame respectively, and match the corresponding corner points by the small range adjacent method.

Step 4: regression analysis of the matching corners to determine the affine parameters a_x , a_y , amend the kernel bandwidth.

Step 5: track and correct the next frame, go to step (2).

5 Experiment Results

The effectiveness of the algorithm is verified by the semi-physical simulation platform. We use F-16 model as tracking target. In the x, y-axis direction, it is headed by a stepper motor driving a ball screw. The movement holder is shown in Fig 2.



Fig. 2 Target motion trestle

Tracking results of receiver aircraft's receptacle which use traditional Meanshift algorithm are shown in Fig. 3:

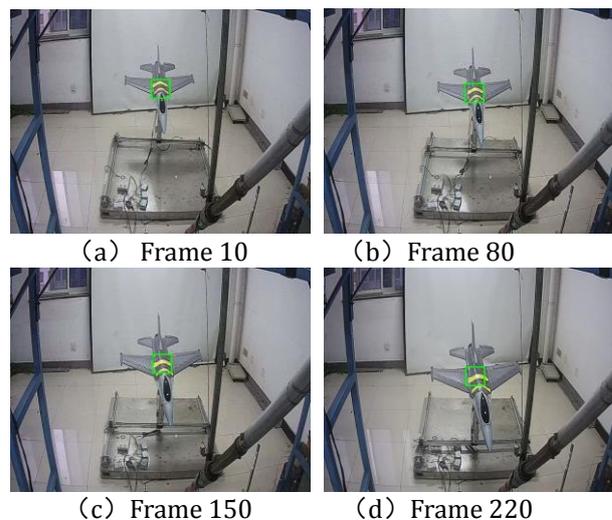


Fig. 3 Tracking results of traditional MS algorithm

As shown in Fig. 3 (a) (b), when the target is slightly larger, the tracking window (kernel window) has a small deviation, the tracking result is still possible, in Fig. 3 (c) target continues to become larger, tracking error becomes larger, and the tracking window is slightly smaller with the target. As shown in Fig. 3 (d), as the target becomes larger, the tracking window is significantly smaller than the target and has a significant deviation.

Kernel bandwidth adaptive algorithm based on corner matching tracking results is shown in Figure 4:

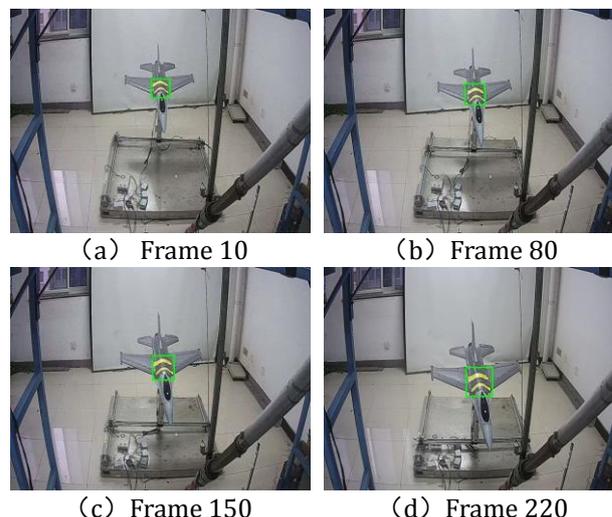


Fig. 4 Tracking results of kernel bandwidth adaptive algorithm

As shown in Fig.4, with the change of the target size, the algorithm mentioned in this paper can accurately track the target, the tracking window can also adapt to the target size changes.

6 Conclusions

This paper proposes a kernel bandwidth adaptive algorithm based on corner matching. Use backward tracking method to amend the core window centre and make the tracking window aim at the target heart. According to the Harris algorithm to detect the corner, use small-scale approach for the two points between frames to match. The affine parameters are obtained based on the result of corner matching and the kernel function is modified. In addition, when using the Meanshift tracking algorithm, add spatial information and colour information to describe the target model comprehensively, and ensure the tracking effect is more accurate. Experiments show that the algorithm proposed in this paper can adapt the core window centre to the target centroid according to the change of the target size, and correct the kernel bandwidth according to the size of the target.

Acknowledgements

This work has been supported by Project of the Program of National Natural Science Foundation of China under the Grant Numbers 61473229.

References

1. SUN ZH S, SUN J X, SONG J ZH, et al.. Antioclusion arithmetic for moving object tracking[J].*Opt. Precision Eng.*,2007,15(2):267-271.
2. ZHOU Shang bo, LI Kun. Algorithm of vedio object tracking based on mean shift with adaptive bandwidth of kernel function [J]. *Computer Engineering and Design*, 2012, 33(07):2776-2781.
3. Wu J, Yan J, Wang Z, et al. Object tracking algorithm for UAV autonomous Aerial Refueling[C]// *Advanced Information Management, Communicates, Electronic and Automation Control Conference*. IEEE, 2017:1665-1669.
4. Bai X, Li A, Li X, et al. Mean-Shift tracking algorithm based on adaptive band width [J]. *Journal of Computer Applications*, 2011, 31(01):254-257.
5. Comaniciu D, Ramesh V, Meer P. Kernel-Based object tracking. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 2003, 25(5):564–575.
6. Peng NS, Yang J, Liu Z, Zhang FC. Automatic selection of kernel-bandwidth for Mean-Shift object tracking [J]. *Journal of Software*, 2005,16(9):1542–1550.
7. Yang Xiaofeng, Zhang Guilin. A Target Tracking Algorithm Based On Affine Transform Model [J]. *Computer and Digital Engineering*,2005,(12):30-34.
8. Liu Chuang, Gong Shengrong. Tracking Multiple Objects Method on Harris corner Sampling[J]. *Journal of Image and Graphics*,2008,13(10):1873-1877.
9. LIU Yu-ran, ZHANG Yu-dong, RAO Xue-jun, HOU Ming-lian. Method of Corner Matching Used in Image Registration [J]. *Opto-Electronic Engineering*,2008,(01):100-104.
10. XU Wei, WANG Wei. Corner Matching-based Approach of View Synthesis [J]. *Science Technology and Engineering*, 2007, (14):3263-3265+3280.