Design of Tracking System Based on Mean-shift and Kalman Filter

Hao Zhang, Jingxin Hong*, Wu Lin, Lin Li
Computer Science Department of Xiamen University, Xiamen, Fujian, P.R.China, 361005

ABSTRACT

This paper presents a set of real-time tracking system based on mean-shift and Kalman filter. The proposed system is composed of two major modules. The host computer adopts tracking algorithm combining mean-shift with Kalman filter to achieve the implementation of object tracking. The improvement of the proposed algorithm is that when the moving vessels are largely occluded, the Kalman filter is updated by velocity vector which is estimated according to target locations in the prior frames, and then the filter is exploited to implement tracking alone. The algorithm achieves good tracking effect. The lower computer is a set of motion control system, consisting of high-performance camera and numerical control PTZ (Pan/Tilt/Zoom). It could manipulate PTZ to track motive object simultaneously according to the results from PC real-time tracking. Experimental results show that the system can implement marine moving object tracking with good real-time performance and robustness.

Key words: mean-shift, Kalman filter, tracking system, marine moving object, movement information

1. INTRODUCTION

Real-time object tracking is required by many vision applications such as surveillance and monitoring, perceptual user interfaces, and video compression. Real-time applications have a great economic value and a wide range of development prospects. Mean-shift is an efficient pattern-matching algorithm in the field of computer vision. Through thorough research over the past few years, it has been successfully exploited in object tracking. As a nonparametric estimator of density gradient, the algorithm has good real-time performance and can be well combinated with other algorithms; In virtue of the kernel histogram modeling, mean-shift is robust to partial occlusion, rotation and scale invariant and computationally efficient, but fast-moving objects or cluttered environment could induce failure of tracking. Kalman filter is an effective recursive filter, which can estimate linearly the minimum variance for the state sequence of dynamic system from a series of noise measurement. The filter featured with a small amount of computational cost could accurately predict the position and velocity of target. In recent years researchers have used the method combining mean-shift and Kalman filter to track non-rigid moving target [4-6], it can ensure the consistency and continuity of target motion, and achieve the effect of real-time tracking. However, it would fail to track in the presence of severe failure or adverse illumination. In this paper, the method has been improved in anti-occlusion in virtue of the target's movement information, and the accuracy of similarity matching between target model and target candidates has also been enhanced.

The main objective of the present work is to propose and describe a design to provide support for the implementation of marine moving object tracking system.

The paper is organized as follows. Section 2 presents the property of the mean shift. Section 3 introduces how to constitute the tracking model from Kalman filter. The tracking algorithm is developed and analyzed in Section 4. Section 5 describes the main parts of the proposed framework. Experiments and result analysis are given in Section 6. Section 7 presents conclusions and directions for future work.

2. MEAN-SHIFT ALGORITHM

Mean-shift is a semi-automatic tracking algorithm. After we select search window in the initial frame, it calculates the color distribution of the search window as the target model by using kernel function weighing, estimate the similarity between the target model (initial color histogram) and the target candidates(present color histogram), and search the

*hjx@xmu.edu.cn; phone 1 359 991-1519

MIPPR 2009: Automatic Target Recognition and Image Analysis, edited by Tianxu Zhang, Bruce Hirsch, Zhiguo Cao, Hanqing Lu, Proc. of SPIE Vol. 7495, 74953D · © 2009 SPIE · CCC code: 0277-786X/09/$18 · doi: 10.1117/12.832551

target candidate that is the most similar to a target model. So the procedure makes the search window move toward towards the maximum increase in the density until achieving the true position of the target. [1]

2.1 Target model and target candidates

Firstly the RGB space of target area is converted to HSV space. Secondly we extracted the component H and divided it into m shares with each corresponding to a sub-characteristic value. Lastly the whole target region can be characterized by these values. For each sub-feature, kernel-based density distribution function is exploited to calculate the probability distribution, and then the u-th characteristic probability can be denoted as

$$q_u = C\sum_{i=1}^{n} k\left(\left\|\frac{x_0 - x_i}{h}\right\|^2\right)\delta\left[b(x_i) - u\right] \tag{1}$$

where x_0 is the center of search window, x_i is the position of i-th pixels, $k\left(\|x\|^2\right)$ is kernel function, h is window radius (half of the width) , b and δ function determine whether the pixel in point x_i accord with the u-th characteristic value. The normalization constant C is derived by imposing the condition $\sum_{u=1}^{m}\hat{q}_u = 1$, from where

$$C = \frac{1}{\sum_{i=1}^{n} k\left(\left\|\frac{x_0 - x_i}{h}\right\|^2\right)} \tag{2}$$

since the summation of delta functions for u = 1…m is equal to one.

Like formula (1), assume y_0 as the center of the target candidate region in current frame, then the probability of u-th characteristic value. is

$$p_u(y_0) = C\sum_{i=1}^{n} k\left(\left\|\frac{y_0 - x_i}{h}\right\|^2\right)\delta\left[b(x_i) - u\right]\cdot \tag{3}$$

2.2 Similarity function and Target Localization [7]

Similarity function characterizes the similarity between the initial target model and target candidates, Similarity measure methods commonly used include Bhattacharyya coefficient, Fisher measure of information, and histogram intersection technique [9]. Here we use Bhattacharyya coefficient, the sample estimate is given by

$$\hat{\rho}(y) = \rho\left(\hat{p}(y), \hat{q}_u\right) = \sum_{u=1}^{m}\sqrt{\hat{p}_u(y)\hat{q}_u} \circ \tag{4}$$

Using now (4) the distance between two distributions can be defined as

$$d(y) = \sqrt{1 - \hat{\rho}(y)}\,. \tag{5}$$

The statistical measure (5) is well suited for the task of target localization. If $d(y)$ was smaller, the similarity between the two color distribution histogram would be higher.

In the search process we employ mean-shift iterations to achieve the maximization of $\hat{\rho}(y)$, the steps list below [2]:

1. Derive the weights of points in present search area according to (5)

$$w_i = \sum_{i=1}^{m}\sqrt{\frac{q_u}{p_u(y_0)}}\delta\left[b(x) - u\right] \tag{6}$$

2. calculate the next new position of target candidates

$$y_i = \frac{\sum_{i=1}^{m} x_i w_i g\left(\left\|\frac{y_0 - x_i}{h}\right\|^2\right)}{\sum_{i=1}^{m} w_i g\left(\left\|\frac{y_0 - x_i}{h}\right\|^2\right)} \tag{7}$$

3. If $\|y_i - y_0\| < \varepsilon$ or reach the maximum of iteration counts, stop. Otherwise, set $y_0 \leftarrow y_i$ and return to Step 1, continue to search the eligible target candidate position.

3. KALMAN FILTER ANALYSIS

3.1 The idea of Kalman filtering

The main idea of Kalman filtering algorithm is that predict the most probable object location in the current frame according to the results of targets tracking in the previous frame, then search target location in the neighbor area of the location. If there is a target existing in the search area, continue to process the next frame. Otherwise, expand the search area and search the target over again in the current frame, or discard the prior frame directly to process the next frame. The point rests with predicting the location of the search region.

3.2 Kalman Filter Modeling [4]

Kalman Filter includes two model:

Motion equation: $$X_k = A \cdot X_{k-1} + W_k \tag{8}$$

Observation equation: $$Z_k = H \cdot X_k + V_k \tag{9}$$

where $X_k = [x, y, dx, dy]^T$ is state of the system at the moment k, $Z_k = [x, y]^T$ is measurement value of system state at the moment k. x and dx are the horizontal position and velocity respectively, y and dy are the vertical position and velocity respectively, W_k and V_k are movement and measurement noise vectors which obey Gaussian distribution $p(w) \sim N(0, Q)$, $p(v) \sim N(0, R)$. The values of state transition matrix *A*, measurement matrix *H*, process noise covariance matrix *Q* and measurement noise covariance matrix *R* list as follow:

$$A = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad H = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}, \quad R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

4. TRACKING ALGORITHM

In [4],[5],[6] a new kind of algorithm combining mean-shift with Kalman filter is described. In [4] the application is that Kalman filter is exploited firstly to predict the initial point of object, and then use mean-shift to locate the target, in the process determining whether the object is occluded through the Kalman filter remains and accounting for the occlusion problem with the linear prediction. In [5] the final target location is gained through applying linear weighted method to the tracking results of the two algorithms respectively according to the degree of interference, and the method achieves good effect. The approach described in [6] figure out the problem of updating target model excessively, but the above algorithms haven’t exploited the target’s motion direction and speed information in process of target tracking.

As the research object of this paper aimed at marine moving ships, we can make use of the relative stableness of marine target speed. In the experiment, assume a threshold α. If Bhattacharyya distance $d(y) > \alpha$, it is inferred that the object has been occluded, and mean-shift could not be utilized to search the target location. So the position of the center of target in previous continuous frames should be exploited to estimate the velocity vector $\hat{V}$ of the moving object, and it is used to update measurement noise vector V_k. Afterward apply Kalman filter to tracking, and compute $d(y)$ simultaneously until $d(y) < \alpha$, then mean-shift would be used to implement the tracking again. The flow char of tracking algorithm combining mean-shift with Kalman filter is shown in figure 1.

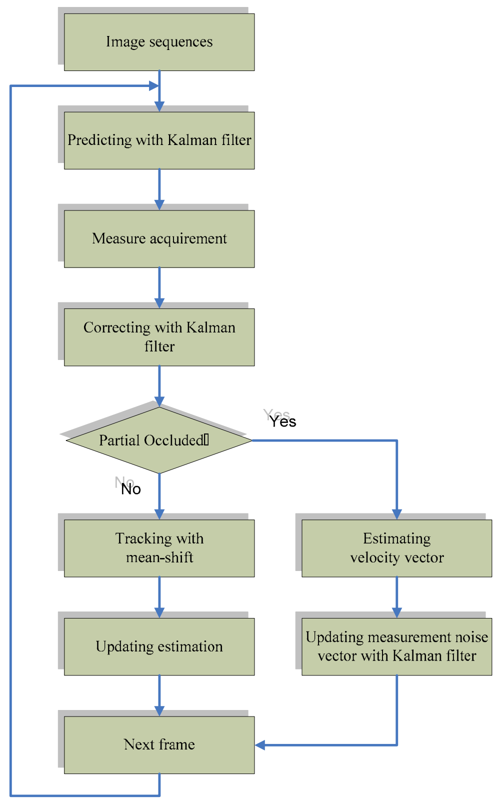


Fig. 1. Flow char of tracking algorithm

The detailed steps of the algorithm are listed as follow:

1. Predict the position of the target at moment k by Kalman filter, and compute the priori error estimate covariance.

$$\hat{x}'_k = A\hat{x}_{k-1} + W_k \;,\; P'_k = AP_{k-1}A^T + Q \tag{10}$$

2. Centered with predicted position $\hat{x}'_k$, acquire the observation value z_k according to (9).

3. Correct the measurement with Kalman filter, compute the revision matrix and renew poster state estimation as well as posteriori error estimate covariance.

$$K_k = P'_k H^T (HP'_k H^T + R)^{-1},$$

$$\hat{x}_k = \hat{x}'_k + K_k(z_k - H\hat{x}'_k),$$

$$P_k = (1 - K_k H)P'_k \tag{11}$$

4. If $d(y) < \alpha$, it is regarded that the target hasn't been partially occluded. Otherwise, go to Step 7.

5. Refer to estimated filtering value $\hat{x}_k$ as the starting point, and exploit the mean-shift algorithm in the neighborhood to search optimal value. Convergence point locates target point in current frame.
6. Replace a posteriori state estimation for convergence point and go to Step 9.
7. Estimate the velocity vector $\hat{V}$ of the moving object through the target location of previous continuous frames.
8. Update measurement noise vector V_k in Kalman filter.
9. Get the next frame at moment k+1 and go to Step 1.

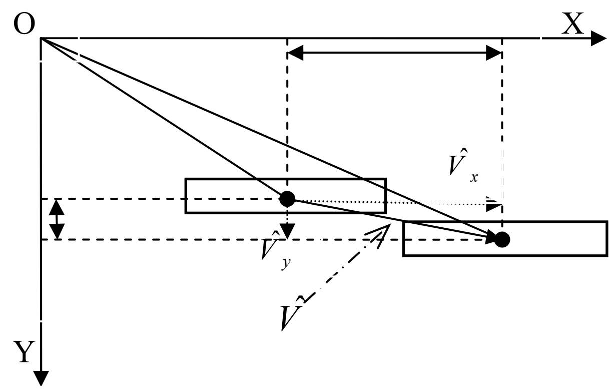


Fig. 2. Estimate velocity vector $\hat{V}$ through previous continuous frames

In figure 2, the rectangle denotes the target model area. The center of the rectangle is estimation of the actual target location. In Step 7, the horizontal component $\hat{V}_x$ and vertical component $\hat{V}_y$ could be estimated respectively through the two formulas below.

$$\hat{V}_x = \alpha_x \sum_{i=n+1}^{n+f} (x_i - x_{i-1}) / f \,, \tag{12}$$

$$\hat{V}_y = \alpha_y \sum_{i=n+1}^{n+f} (y_i - y_{i-1}) / f \,\cdot \tag{13}$$

In both formulas, x_i and y_i are the horizontal and vertical coordinate of target center respectively, α_x and α_y are proportional coefficients, f is the number of continuous frames.

5. FRAMEWORK FOR TRACKING SYSTEM

The tracking system based on the algorithm combining mean-shift with Kalman filter is composed of 2 modules which are depicted in figure 2. The host computer adopts the above tracking algorithm to implement object tracking. The method not only ensures consistency and coherence of motive object, but also gains the effect of real-time tracking. The inferior machine is a set of motion control system, consisting of high-performance camera and numerical control PTZ(Pan/Tilt/Zoom). It could manipulate PTZ to track motive object simultaneously according to the results from PC real-time tracking.

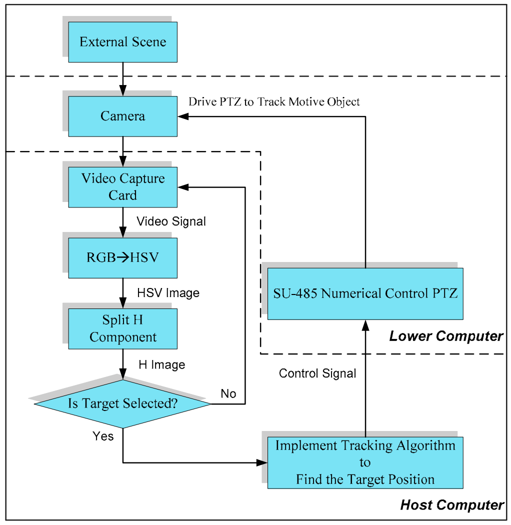


Fig. 3. Framework for the set of tracking system

5.1 Host computer

The host is an Intel PentiumⅣ2.83 GHz Dual Core PC with 2GB memory, and the 100moons video capture card is set inside it; The software of the tracking system is developed on the basis of DirectShow SDK by Visual C++6.0 and OpenCV. The system utilizes the above algorithm based on mean-shift and Kalman filter to implement tracking.

5.2 Lower computer

The lower computer consists of SU-450 numerical control PTZ and a monocular high-performance camera fixed on PTZ. The pixel resolution of camera is 320×240 and the capture frame rate is 25f/s. The role of the PTZ is to actuate the camera's rotation, and the superior machine may send instruction to PTZ and camera through the serial port to control camera's rotation and zoom. This method can change camera's field of vision to achieve the effect of real-time tracing.

5.3 Communication system

This system uses serial communication mode and RS485 is supported by the communication interface of numeric control PTZ. Through the RS232/RS485 conversion interface host computer could communicate with lower computer under the condition that the baudrate of serial port communication is 9600 b/s. The operations of sending and receiving data are carried out between superior machine and serial port through the API functions supplied by Windows CE, and actuate the SU-450 numerical control PTZ.

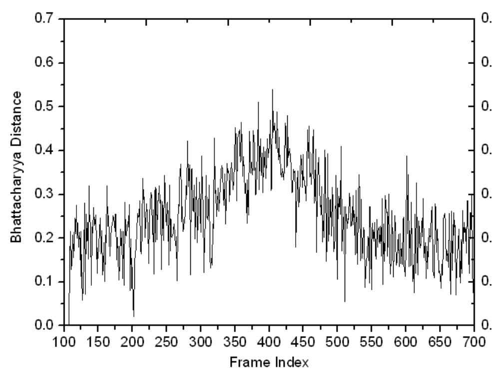
6. EXPERIMENTS & ANALYSIS

In the IDE VC6.0, We finish the video capture program based on DirectShow and use C++ and OpenCV to develop the tracking algorithm. In the experiments we use the system to track moving ships on sea surface. The tracking system could execute real-time detection for the selected motive objects in the camera's field of vision and calculate centroid of the selected object. When the moving target is partly occluded, it can predict the direction and velocity of the movement, and implement reliable tracking according to the above information. However, if the target is about to move out of range, it will control PTZ to rotate or zoom to ensure target in the camera's field of vision. In the experiments, the Bhattacharyya distance α= 0.4, speed of PTZ is 0.047rad/s. The results are shown in figure 5, 6 where the red rectangle is tracking window and the green cross denote the centroid of target.

[a] Frame #97 [b] Frame #206 [c] Frame #279 [d] Frame #308

Fig. 4. Tracking the ship with mean-shift algorithm

[a] Frame #60 [b] Frame #259 [c] Frame #411 [d] Frame #699



[e] Bhattacharyya distance during the tracking procedure

Fig.5. Tracking the ship in occlusion situation with the proposed algorithm

Figure 4 shows the effect of tracking the ship with mean-shift, it can be observed that it failed to track object in the frame 308 because of the large occlusion. Figure 5 shows the effect of tracking the ship in occlusion situation with the proposed algorithm. In Fig. 5[b] the target position could be detected when the ship was partly occluded. It can be observed in Fig. 5[c] that tracking system still traced the ship under the condition of severe occlusion. Fig. 5[d] shows the ship was tracked stably after it drove out of the occlusion. From fig. 5[e] it could be manifested that Bhattacharyya distance mainly concentrates between 0.2 and 0.3. The accuracy of model similarity matching is enhanced to a certain extent.

[a] Frame #937 [b] Frame #942 [c] Frame #947 [d] Frame #954

Fig.6. Tracking the ship under the condition of camera rotating

From fig. 6 we can see that PTZ actuated camera's movement to trace the target when it was about to move out of range. At that moment, the tracking window deviated from the actual target location just as the fig.6 [a, c] shows. This is because camera's movement produced local disturbance to the tracking region. However, the object was tracked stably over again at once, which is shown in fig. 6[d]. It is manifested that the proposed algorithm has good robustness.

Experimental results show that the system can detect the specified object and implement real-time object tracking under the simple circumstance such as moving ships tracking on sea surface when both of distance and the rate of target's movement are in a certain range.

7. CONCLUSIONS & FUTURE WORK

In this paper, we proposed a tracking system framework based on mean-shift and Kalman filter to perform real-time tracking in live video. The mean-shift combined with Kalman filter, which is more effective than mean-shift in target location, has also been successfully applied to object tracking. Through exploiting the estimated value for target's velocity to update Kalman filter's observational equation, the resistance of the resulting filter to the partial occlusion enables the accurate detection and handling of more severe occlusion. The method also improves accuracy of the similarity matching to a certain extent. It has been verified that this tracking system can track object stably in a long time, even in an occlusion situation. Future works include robustness improvement of target location under the lighting condition of abrupt changes and relocation in camera zooming. Fine-tuning and enhancements to the system integration to facilitate an integrated system functioning in a seamless manner in real-time will be further explored in future work.

REFERENCES

1. Dorin Comaniciu, Visvanathan Ramesh, Peter Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intellligence*, 25(5):564-577 (2003).
2. Dorin Comaniciu, Peter Meer, "Mean Shift: A Robust Approach Toward Feature Space Analysis," *IEEE Transaction. on Pattern Analysis and Machine Intelligence*, Vol.24, No.5, pp:603-619 (2002).
3. FilterHieu T. Nguyen, Arnold W.M. Smeulders, "Fast Occluded Object Tracking by a Robust Appearance filter," *IEEE Transactions on Pattern analysis and Machine Intelligence*, Vol. 26, No.8 (2004).
4. Shengli Zhu, Shanan Zhu, Xuchao Li, "Algorithm for tracking of fast motion objects with Mean shift," *Opto-Electronic Engineering*, Vol. 33(5), pp: 66-70 (2006).
5. Faliang Chang, Xue Liu, Huajie Wang, "Target tracking algorithm based on Mean-shift and Kalman filter," *Computer Engineering and Applications*, Vol.43 (12), pp: 50-52 (2007).
6. Zhengzhou Li, Guojing Liu, "Target tracking based on Mean-shift and Kalman filter," *Journal of Projectiles, Rockets, Missiles and Guidance*, Vol.28 (1), pp: 71-74 (2008).
7. Ningsong Peng, Jie Yang, Zhi Liu, Feng-Chao Zhang, "Automatic Selection of Kernel-Bandwidth for Mean-Shift Object Tracking," *Journal of Software*, Vol.16(9), pp:1542-1549 (2005).
8. Jing Liang, Cheng Zhi, Jun Zhou, "Anti-occlusion Motive Target Tracking Arithmetic Based on Mean Shift," *Video Application & Project*, Vol.32 (12), pp: 82-85 (2008).
9. M.J. Swain, D.H. Ballard, "Color Indexing," *Intern. J. Comp*. Vis., 7(1):11-32 (1991).