Target Tracking based on improved mean shift

LI Jun-wei[1,2], LIU Feng[2], ZHU Zhen-fu[2], YU Zhenhong[2]
(1. Coll. of Information Science and Technology, Beijing Inst. of Technology, Beijing 100081,China;
2.National key Laboratory on Optical Features of Environment and Target, Beijing 100854)

Abstract：*A method based on mean shift for target track was improved, which adopted LMS filter to update target templet and Kalman filter to forecast target position. While tracking , Klaman filter was used to forecast the position in next frame according to the target move-track, then mean shift algorithm was iteratively processed around the forecast position for searching the best matching position with target histogram. The target info from iteratively processed was send to LMS filter for target template update, and the position was send to Klaman filter as the observation value of target position, which can obtain the target position through updating the Klaman filter. This method reduced the influence owing to the change of target appearance, meanwhile stably tracked target, and the tracking result was validated.*

Key words：*mean shift*；*LMS filters*；*Kalman filters*；*Target tracking*

1. Introduction

Target may be lost while tracking based on region, especially while matching with old image pixels, the target is circumrotating or the size of target has a large change or the image sequence is holistic-running for CCD's moving. Mean Shift algorithm matches with histogram of target template and affects little from target's movement., and can track target in the condition of CCD's moving or target's rigid change. An improved Mean Shift algorithm is proposed in this paper: LMS filters is to update the target template, Kalman filter with Mean Shift is to predict the target position. While tracking, Kalman filter is first to predict the target's localization in new frame with target's contrail, then the Mean Shift algorithm is iterated round the predicted location, searching for the location most matched with target histogram, and the target information iterated is send to LMS filter for updating template to reduce the effects of target's change. In order to stably track, the location searched is only send to Kalman filter as the referenced value, and the last location is get from Kalman filter's updating. The measurement vector is derived based on mean shifts, while the prediction of the next target location is computed by a Kalman filter. This algorithm not only tracks target stably but also predicts target location in next frame correctly, and the pipeline is figure 1.

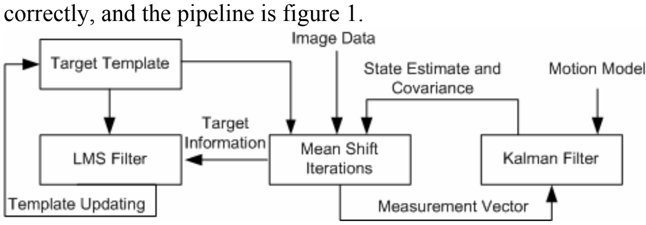


Fig.1. *Block diagram of improved Mean Shift tracking*

2 Improved Mean Shift Tracking

We introduce the Mean Shift algorithm and the coefficient—Bhattacharyya for degree of two histograms matching, and list the all steps for maximizing Bhattacharyya coefficient with Mean Shift algorithm, then update the target template with LMS filter, design the Klaman filter for predicting and estimating target's location.

2.1 Mean Shift algorithm

Mean Shift algorithm[3,8] is one matching algorithm based on estimation of kernel density without parameters. Given a set S in the n dimensional space X , $S \subset X$,the Sample Mean Shift vector around x ($x \in X$) is

$$m(y)=\frac{\sum_{i=1}^{n} x_i g\left(\left\|\frac{y-x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n} g\left(\left\|\frac{y-x_i}{h}\right\|^2\right)} \tag{1}$$

And the $g(x)=-k'(x)$,is the negative gradient of $k(x)$, $k(x)$ is the kernel function. Usually the kernel function in Mean Shift algorithm is Gaussian kernel: $k(x)=\frac{1}{2\pi}\exp(-\frac{1}{2}\|x\|^2)$, $s \in S$, h is the width of window, and $\Delta(y)=m(y)-y$ is the Mean Shift vector. The kernel of Mean Shift algorithm is to move data repeatedly along the Mean Shift vector, and calculates a new $y'=y+\Delta(y)$,until gets the extremum[1,2].And the Mean Shift vector direction is the negative gradient of the estimated density function.

2.2Histogram Similarity Measure

Given the predicted location of the target in the current frame and its uncertainty, the measurement task assumes the search of a confidence region for the target candidate that is the most similar to the target model. The similarity measure we develop is based on color information. The feature z representing the color of the target model is assumed to have a density function q_Z , while the target candidate centered at location y has the feature distributed according to $p_Z(y)$,The problem is to find the discrete location y whose associated density $p_Z(y)$ is the closest to the target density $p_Z(y)$ is the closest to the target density q_z .

Our measure of the distance between the two densities is based on the Bhattacharyya coefficient, whose general form is defined by

$$\rho(y)\equiv\rho\left[p(y),q\right]=\int\sqrt{p_z(y)q_z}dz \tag{2}$$

The derivation of Bhattacharyya coefficient from sample data involves the estimation of the densities p and q , for which we employ the histogram formulation. The discrete density $\hat{q} = \left\{\hat{q}_u\right\}_{u=1\ldots m}$ (with $\sum_{u=1}^{m} \hat{q}_u = 1$)is estimated from the m-bin histogram of the target model, while $\hat{p}(y) = \left\{\hat{p}_u(y)\right\}_{u=1\ldots m}$ (with $\sum_{u=1}^{m} \hat{p}_u = 1$)is estimated at a given location y from the m-bin histogram of the target candidate. Therefore, the sample estimate of the Bhattacharyya coefficient[3] is given by

$$\hat{\rho}(y) \equiv \rho\left[\hat{p}(y), \hat{q}\right] = \sum_{u=1}^{m} \sqrt{\hat{p}_u(y)\hat{q}_u} \tag{3}$$

The statistical measure (3) is a metric valid for arbitrary distributions, being nearly optimal and invariant to the scale of the target. It is therefore superior to other measures such as histogram intersection [4], Bhattacharyya distance, Fisher linear discriminant[5], or Kullback divergence.

2.3Candidate Target Localization

For target tracking, if all pixels of the region are gram-statistic with the same weight, the uncertainty of matching is large because the pixels around target are covered and the effect of the mixed background. This section introduces weighted histogram, the higher weight to the closer target center. The weighted histogram is adopted in(3), and maximize (3) through Mean Shift iterations.

2.3.1 Weighted Histogram Computation

Target Model We denote by $\left\{X^*_i\right\}_{i=1\ldots n}$ the pixel locations of the target model, centered at 0. Let $b: R^2 \rightarrow \{1\ldots m\}$ be function which associates to the pixel at location X^*_i the index $b\left(X^*_i\right)$ of the histogram bin corresponding to the color of that pixel. The probability of the color u in the target model is derived by employing a convex and monotonic decreasing function $k: \left[0,\infty\right) \rightarrow R$ which assigns a smaller weight to the locations that are farther from the center of the target. The weighting increases the robustness of the estimation, since the peripheral pixels are the least reliable, being often affected by occlusions (clutter) or background. By assuming that the generic coordinates x and y are normalized with $h_{(x)}$ and $h_{(y)}$, respectively, we can write

$$\hat{q}_u = C\sum_{i=1}^{n} k\left(\left\|X^*_i\right\|^2\right)\delta\left[b\left(X^*_i\right) - u\right] \tag{4}$$

Where δ is the Kronecker delta function. C is the normalization constant.

Target Candidates

Let us denote by $\left\{X_i\right\}_{i=1\ldots n_h}$ the pixel locations of the target candidate, centered at y in the current frame. Employing the same weighting function k , the probability of the color u in the target candidate is given by

$$\hat{p}_u(y) = C_h \sum_{i=1}^{n_h} k\left(\left\|\frac{y - x_i}{h}\right\|^2\right)\delta\left[b\left(X_i\right) - u\right] \tag{5}$$

The scale of the target candidate (i.e., the number of pixels)is determined by the constant h which plays the same role as the bandwidth (radius) in the case of kernel density estimation. The normalization constant C_h does not depend on y , and can be precalculated for a given weighting function k and bandwidth h .

2.3.2Comparability Maximization

The search for the new target location in the current frame starts at the predicted location $\hat{y}_0$ of the target computed by the Kalman filter. Thus, the color probabilities $\left\{\hat{p}_u\left(\hat{y}_0\right)\right\}_{u=1\ldots m}$ of the target candidate at location $\hat{y}_0$ in the current frame have to be computed first. The minimization of the distance (3) being equivalent to the maximization of the Bhattacharyya coefficient, we start with the Taylor expansion of $\rho\left[\hat{p}(y), \hat{q}\right]$ around the values $\hat{p}_u\left(\hat{y}_0\right)$, which yields

$$\rho\left[\hat{p}(y), \hat{q}\right] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u\left(\hat{y}_0\right)\hat{q}_u} + \frac{1}{2}\sum_{u=1}^{m}\hat{p}_u(y)\sqrt{\frac{\hat{q}_u}{\hat{p}_u\left(\hat{y}_0\right)}} \tag{6}$$

Introducing now (5) in (6) we obtain

$$\rho\left[\hat{p}(y), \hat{q}\right] \approx \frac{1}{2}\sum_{u=1}^{m}\sqrt{\hat{p}_u\left(\hat{y}_0\right)\hat{q}_u} + \frac{C_h}{2}\sum_{i=1}^{n_h} w_i k\left(\left\|\frac{y - x_i}{h}\right\|\right) \tag{7}$$

where

$$w_i = \sum_{u=1}^{m}\delta\left[b\left(X_i\right) - u\right]\sqrt{\frac{\hat{q}_u}{\hat{p}_u\left(\hat{y}_0\right)}} \tag{8}$$

For the first term in equation (7) is independent of y, to maximize equation(7), the second term has to be maximized. The maximization can be efficiently achieved based on the mean shift iterations [6], using the following algorithm.

Given the distribution $\left\{\hat{q}_u\right\}_{u=1\ldots m}$ of the target model and the predicted location $\hat{y}_0$ of the target, the maximization of Bhattacharyya Coefficient $\rho\left[\hat{p}(y), \hat{q}\right]$ is

1). Compute the distribution $\left\{\hat{p}_u\left(\hat{y}_0\right)\right\}_{u=1\ldots m}$ of the predicted location $\hat{y}_0$, and evaluate the similarity

$$\rho\left[\hat{p}\left(\hat{y}_0\right), \hat{q}\right] = \sum_{u=1}^{m}\sqrt{\hat{p}_u\left(y_0\right)\hat{q}_z}$$

2). Derive the weights $\left\{w_i\right\}_{i=1\ldots m}$ according to (8).

3). Derive the new location of the target $\hat{y}_1$

$$\hat{y}_1 = \frac{\sum_{i=1}^{n_h} x_i w_i g\left(\left\|\frac{\hat{y}_0 - x_i}{h}\right\|^2\right)}{\sum_{i=1}^{n_h} w_i g\left(\left\|\frac{\hat{y}_0 - x_i}{h}\right\|^2\right)}$$

4). If $\left\|\hat{y}_1 - \hat{y}_0\right\| < \varepsilon$ Stop. Otherwise Set $\hat{y}_1 = \hat{y}_0$ and go to Step 1.

The threshold ε in step 4 should be little enough to make the vectors $\hat{y}_0$ and $\hat{y}_1$ within the same pixel in image coordinates.

2.4LMS Filter for Target Template Updating

For noise infection in the image sequence, The histogram ingredients of the target template kernel function can't correctly reflect the information of the current target color, and the change and shelter for acute illumination in video under some condition are considered as noise. How to duly and truly update the target template kernel histogram function and wipe off the infection of target change and shelter is the urgent question in moving target tracking .

This section Filter disposal is applied to non-zero portion of histogram of target template kernel function in Mean Shift algorithm. For the finity of target color distribution, the non-zero portion is relatively few, and the filter can have few and high speed. While updating the non-zero portion, eliminating disappeared and filling in new portion, timely reflects target change, settles the inflection of target change and shelter.

LMS filter can primly dispose non-linearity non-smooth signal, and its prime weight estimation V_t can give the kernel function histogram portion $\hat{q}_u(t)$ for reducing noise. For the kernel function histogram of target template, the output of t frame in LMS filter is: $\hat{q}_u(t) = V_t^T Z(t)$ and

$$V_t^T = [v_0(t), v_1(t) \cdots v_{K-1}(t)]^T \tag{9}$$

$\{v_i(t)\}_{i=0 \cdots K-1}$ is the weight gene, $Z(t)$ is composed of $q_u(t)$ time-lapse sequence.

$$Z(t) = [q_u(t), q_u(t-1) \cdots q_u(t-K+1)]^T \tag{10}$$

and $q_u(t)$ is the u portion of the target kernel function histogram of the t frame. K is the count of filter weight vector, commonly is 5 to 10.The recursive formula of weight vector $V(t)$:

$$V(t) = V(t-1) + 2\mu[q_u(t) - V^T(t-1)Z(t)]Z(t) \tag{11}$$

and μ is parameter for adjusting the convergence speed.

The process of LMS filter is: first calculate the error $e(t) = q_u(t) - V^T(t-1)Z(t)$, then the weight vector $\{v_i(t)\}_{i=0 \cdots K-1}$ for (12)

$$\begin{cases} v_0(t) = v_0(t-1) + 2\mu e(t) q_u(t) \\ v_1(t) = v_1(t-1) + 2\mu e(t) q_u(t-1) \\ \vdots \\ v_{K-1}(t) = v_{K-1}(t-1) + 2\mu e(t) q_u(t-K+1) \end{cases} \tag{12}$$

get the output $\hat{q}_u(t) = \sum_{i=0}^{K-1} v_i(t) q_u(t-i)$ (13)

$\hat{q}_u(t)$ is the result of the u portion in the target kernel template histogram of the t frame, as the value for updating template.

2.5Target Tracking with Kalman Filter and Mean Shift

The tracker estimates the target moving state portion x 、y with two independent Kalman Filters. In this section, the filters have the same parameters, so we can design filter with x portion. The target vector is : $X = (x, \dot{x}, \ddot{x})$,which means the target position(coordinate), speed and acceleration. The movement function and observation function are:

$$\begin{aligned} X_i &\sim N(D_i X_{i-1}, \Sigma_{d_i}) \\ Y_i &\sim N(M_i X_i, \Sigma_{m_i}) \end{aligned} \tag{14}$$

and $N(\mathrm{M}, \Sigma)$ is the multidimensional normal school with mean M ,covariance matrix Σ , X_i is the state of i frame, Y_i is the observation value of target state in i frame.

The measured vector of Kalman is the candidating target location in Mean Shift, so the M_i is $[1,0,0]^T$,the noise Σ_{m_i} is a quantity. For the invariable acceleration, the acceleration between two frames keeps little surge, so the transfer matrix D_i is:

$$D_i = \begin{bmatrix} 1 & 1 & 0 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}$$

the covariance matrix Σ_{d_i} means the uncertainty of state transfer, supposed the uncertainty of three moving state portions is independent, Σ_{d_i} is one diagonal matrix, with the value on diagonal denoting the error square.

With these Kalman filter parameters, the after-probability distributing can be educed through iterative forecasting function and updating function. "-" forecasting value, and "+" updating value:

The forecasting function for Kalman Filter:

$$\begin{aligned} X_i^- &= D_i X_{i-1}^+ \\ \Sigma_i^- &= \Sigma_{d_i} + D_i \Sigma_{i-1}^+ D_i^T \end{aligned} \quad \mathrm{e} \tag{15}$$

The updating function for Kalman Filter:

$$\begin{aligned} K_i &= \Sigma_i^- M_i^T [M_i \Sigma_i^- M_i^T + \Sigma_{m_i}]^{-1} \\ X_i^+ &= X_i^- + K_i [y_i - M_i X_i^-] \\ \Sigma_i^+ &= [I - K_i M_i] \Sigma_i^- \end{aligned} \tag{16}$$

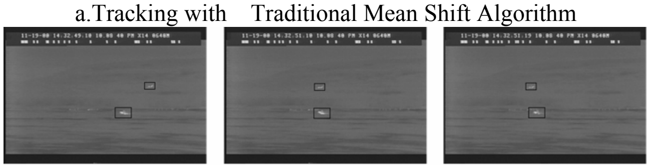
with Gain matrix K_i , unit matrix I .

With the initialization of target moving-state, the observation noise of target state and the transferring error covariance of conjoint target state are chose based on fact, the target tracking can run based on Kalman filter and Mean Shift Filter. The initialization location of target is from the detection module, and the speed and acceleration is 0.

3. EXPERIMENTS

One sequence image with high-speed target was tested with the traditional Mean Shift and improved Mean Shift. The 54th, 76th, 77th frames are displayed with large plane with low speed on ground and small plane with high speed in sky. The Figure 2-a shows the result with traditional Mean Shift, and Figure 2-b with improved Mean Shift. For the large plane on ground, the shift pixels relative to the size of target is small, the Mean Shift algorithm without Klaman filter can also search and match the right location in the current frame. But for the small plane in sky, the speed is not fast before 76th frame, and the shift pixel relative to the size of target is not large, so the old algorithm can also track target. But between the 76th and 76th frame the shift pixel relative to the size is larger, so the matching location with the traditional algorithm has large difference with the fact, and the improved algorithm with Kalman filter matching, the location error is small, can convergent to the right location.

a.Tracking with Traditional Mean Shift Algorithm



b.Tracking with Improved Mean Shift Algorithm

Fig.2. *Plane sequence Tracking with two algorithm (The frames 54, 76, and 77 are shown left-right)*

4conclusion

We proposed an improved Mean Shift algorithm for target tracking, which adopted LMS filter to update target template and Kalman filter to forecast target position. This method reduced the influence owing to the change of target appearance and shelter, meanwhile stably tracked target, and the examination shown its validated tracking capability and high Robust.

5. REFERENCES

[1]Cheng, Y. Mean shift, mode seeking and clustering, IEEE Trans. Pattern Analysis and Machine Intelligence,1995,17(8),790-799.

[2]Collins, R.T. Mean shift blob tracking through scale space, IEEE Computer Vision and Pattern Recognition, 2003,2,234-240.

[3] Comaniciu,D,Ramesh, V.,Kernel-based object tracking , IEEE Trans. Pattern Analysis and Machine Intelligence, 2003,25(5),564-575

[4] T.Kailath. The Divergence and Bhattacharyya Distance Measures in Signal Selection, *IEEE Trans. Commun. Tech.*, COM-15:52–60, 1967.

[5] M.J. Swain, D.H. Ballard, "Color Indexing," *Intern. J. Comp. Vis.*, 7(1):11–32, 1991.

[6] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Second Ed., Academic Press, Boston, 1990.

[7] D. Comaniciu, P. Meer, "Mean Shift Analysis and Applications," *IEEE Int'l Conf. Comp. Vis.*, Kerkyra, Greece, 1197– 1203, 1999.

[8] Comaniciu, D.,Ramesh, V., Real-time tracking of non-rigid objects using mean shift, IEEE Internat.Proc. Computer Vision Pattern Recognition,2000,2,142-149

Li Junwei received his BS in optical from the Institute No.207 of the second Academy.CASIC(2002) and PhD in optical from Beijing Inst. of Technology (2007). Since then he has worked in the National key Laboratory on Optical Features of Environment and Target, his work has focused on the image processing and. target tracking.